A Multimoment Finite-Volume Shallow-Water Model on the Yin–Yang Overset Spherical Grid

XINGLIANG LI
Tokyo Institute of Technology, Yokohama, Japan, and State Key Laboratory of Severe Weather, Chinese Academy of Meteorological Sciences, Beijing, China

DEHUI CHEN
State Key Laboratory of Severe Weather, Chinese Academy of Meteorological Sciences, Beijing, China

XINDONG PENG* and KEIKO TAKAHASHI
Earth Simulator Center, JAMSTEC, Yokohama, Japan

FENG XIAO
Institute of Mechanics, Chinese Academy of Sciences, Beijing, China, and Tokyo Institute of Technology, Yokohama, Japan

(Manuscript received 26 March 2007, in final form 7 November 2007)

ABSTRACT

A numerical model for shallow-water equations has been built and tested on the Yin–Yang overset spherical grid. A high-order multimoment finite-volume method is used for the spatial discretization in which two kinds of so-called moments of the physical field [i.e., the volume integrated average (VIA) and the point value (PV)] are treated as the model variables and updated separately in time. In the present model, the PV is computed by the semi-implicit semi-Lagrangian formulation, whereas the VIA is predicted in time via a flux-based finite-volume method and is numerically conserved on each component grid. The concept of including an extra moment (i.e., the volume-integrated value) to enforce the numerical conservativeness provides a general methodology and applies to the existing semi-implicit semi-Lagrangian formulations. Based on both VIA and PV, the high-order interpolation reconstruction can only be done over a single grid cell, which then minimizes the overlapping zone between the Yin and Yang components and effectively reduces the numerical errors introduced in the interpolation required to communicate the data between the two components. The present model completely gets around the singularity and grid convergence in the polar regions of the conventional longitude–latitude grid. Being an issue demanding further investigation, the high-order interpolation across the overlapping region of the Yin–Yang grid in the current model does not rigorously guarantee the numerical conservativeness. Nevertheless, these numerical tests show that the global conservation error in the present model is negligibly small. The model has competitive accuracy and efficiency.

1. Introduction

In the past decade, some sophisticated numerical algorithms that were originally developed and used in the computational fluid dynamics (CFD) community have been introduced and getting an increasing popularity in global atmospheric and oceanic modeling. Among the representative ones are the spectral-element (SE) method (Thomas and Loft 2002; Giraldo and Rosmond 2004) and the discontinuous Galerkin (DG) method (Nair et al. 2005a). Although these methods usually require more computational effort than the conventional finite-difference or finite-volume methods, their superiority in numerical accuracy and convergence motivates the further exploration to implement them in geophysical fluid simulations.

* Current affiliation: State Key Laboratory of Severe Weather, Chinese Academy of Meteorological Sciences, Beijing, China.

Corresponding author address: Feng Xiao, Department of Energy Sciences, Tokyo Institute of Technology, 4259 Nagatsuta, Midori-ku, Yokohama, 226-8502, Japan.
E-mail: xiao@es.titech.ac.jp

DOI: 10.1175/2007MWR2206.1

© 2008 American Meteorological Society
We have recently developed a general numerical framework, the so-called Constrained Interpolation Profile/Multimoment Finite-Volume Method (CIP/MM FVM; Yabe et al. 2001; Xiao and Yabe 2001; Xiao 2004; Xiao et al. 2006a; Li and Xiao 2007) for CFD applications. The basic idea of the CIP/MM method is to make use of more than one moments, which are in fact the volume-integrated average (VIA) and the point value (PV) in the present model for a physical field when constructing the spatial numerical approximations, and to treat all the moments as the model variables that are integrated forward in time separately. Based on multimoments, the high-order interpolation can be constructed over single grid cell. The CIP/MM FVM has a great flexibility in updating different moments (i.e., different moments can be computed by different numerical methods). In the present work, the PV, which is not necessarily conserved, is updated by a semi-Lagrangian and semi-implicit numerical procedure that has been widely used in meteorological modeling. The conservative moment VIA on the other hand is advanced through a flux-based finite-volume formulation (or equivalently a volume remapping). The resultant numerical algorithm can be interpreted as a combination of the semi-implicit semi-Lagrangian method and a finite-volume method, where one of the moments (VIA) is numerically conserved, and is quite simple and easy to implement. Different numerical dispersion can also be obtained from the multimoment formulation. Xiao et al. (2006b), for example, discussed the numerical dispersion of the simplest multimoment finite-volume method for the geostrophic adjustment.1

Another problem associating the construction of the global circulation model comes from the numerical representation of the spherical geometry. Because of the singularity and the convergence of meridians in the polar regions of the longitude–latitude grid system, other alternatives that share more uniform grid spacing are recently explored, for instance, the icosaheblack geodesic grid (Heikes and Randall 1995; Stuhne and Peliter 1999; Tomita et al. 2001; Majewski et al. 2002) and the gnomonic-cubic grid (Sadourny 1972; Rančić et al. 1996; Ronchi et al. 1996; McGregor 1997). Being a Chimera grid, the Yin–Yang grid (see Fig. 1) was suggested by Kageyama and Sato (2004) as a quasi-uniform overset grid free from the polar singularity. A Yin–Yang grid is constructed by overlapping two perpendicularly oriented longitude–latitude grid components of the low-latitude part. With each of its component being part of the conventional longitude–latitude grid, the Yin–Yang grid provides a convenient platform readily to accommodate the numerical works originally developed for the latitude–longitude grid. Li et al. (2006) developed an accurate semi-Lagrangian scheme on the Yin–Yang grid by using a high-order interpolation for the overlapping region. However, the interpolation for communicating data across the Yin–Yang border usually does not guarantee the numerical conservativeness. Peng et al. (2006) proposed an exactly conservative advection transport scheme on the Yin–Yang grid based on a piecewise constant reconstruction. For higher-order approximations or the general system of conservative equations for fluid dynamics, the complete conservativeness across the Yin–Yang overlapping region requires more complicated numerics. Nevertheless, as shown in this paper, using a conservative numerical scheme for each individual grid component and minimizing the overlapping zone, one can still obtain adequate, even if not rigorous, conservativeness for medium-range weather prediction or short-range climate simulation.

In this paper, we will implement the CIP/MM FVM on the Yin–Yang grid. All the computations on each individual grid component are exactly conserved and the halo layer in the overlapping region is minimized by the multimoment reconstruction. This paper is organized as follows. The Yin–Yang grid is briefly described in section 2. Section 3 presents the spatial discretization based on two kinds of moments (i.e., PV and VIA) in two-dimensional longitude–latitude grid. The time integration scheme for the shallow-water equation is discussed in section 4. The semi-implicit/semi-Lagrangian procedure (McDonald and Bates 1989) is adopted to compute the PVs. With the PVs updated, the VIAs are simply predicted by evaluating the numerical fluxes in terms of the PVs averaged over two time steps. The scheme for advection is briefly presented in section 5. The model is verified in section 6 by a series of widely used benchmark tests. We present our conclusions in section 7.

2. Yin–Yang grid

The Yin–Yang grid (Kageyama and Sato 2004) is an overset grid for spherical geometry, which consists of

1 We should notify another solution of the dispersion equation of the M grid in Xiao et al. (2006b) as follows [see Xiao et al. (2006b) for notation description]:

$$
\left( \frac{a}{f} \right)^2 = \frac{1}{2} + 2\gamma - \sqrt{\left( \frac{1}{2} - 2\gamma \right)^2 + \epsilon}.
$$

It is observed that this mode has a frequency significantly lower than that of the physical one discussed in Xiao et al. (2006b). From the numerical experiments carried out so far, we have not found a significant effect of this computational mode on the numerical solutions.
two notched longitude–latitude grids normal to each other (Fig. 1). The component (Yin or Yang) grid is selected to be the low-latitude part of the longitude–latitude grid. The composition of the two component grids, with one of the components being perpendicular to the other, covers the globe with an overlapping zone where the data need to be communicated between the two components during the computation.

The relationship between the Yin coordinate and the Yang coordinate is easily found by considering any position vector in the Cartesian coordinates \((X, Y, Z)\) as:

\[
(X'', Y'', Z'') = (-X', Z', Y'),
\]

where the superscript \(n\) denotes the Yang coordinate and \(e\) is the Yin coordinate.

From Eq. (1), we have the relationship in the spherical coordinate \((\lambda, \varphi)\):

\[
\cos \varphi'' \cos \lambda'' = -\cos \varphi' \cos \lambda', \\
\cos \varphi' \sin \lambda'' = \sin \varphi', \quad \text{and} \\
\sin \varphi'' = \cos \varphi' \sin \lambda'.
\]

In this paper, the computational domain (Yin or Yang grid) is defined to be from \(45^\circ\)S to \(45^\circ\)N in the latitudinal direction and from \(45^\circ\) to \(315^\circ\)E in the longitudinal direction. As we can see from the next section, the spatial reconstruction based on multimoments can be done over a single computational cell, thus the overlapping zone of the Yin–Yang grid used here is minimized.

The grid partitioning for both Yin and Yang components are

\[
\varphi^j_i = \varphi^\text{min}_i + j\Delta \varphi, \quad (j = 0, N^\varphi - 1) \quad \text{and} \\
\lambda^i_j = \lambda^\text{min}_j + i\Delta \lambda, \quad (i = 0, N^\lambda - 1),
\]
with
\[
\Delta \varphi = (\varphi_{\text{max}} - \varphi_{\text{min}})/(N_\varphi - 1) \quad \text{and} \quad \\
\Delta \lambda = (\lambda_{\text{max}} - \lambda_{\text{min}})/(N_\lambda - 1),
\]
where \( l \) represents \( n \) (Yang grid) or \( e \) (Yin grid). As mentioned before, the domain bounds of the two components are specified, respectively, as \( \varphi_{\text{min}} = -\pi/4, \varphi_{\text{max}} = \pi/4, \lambda_{\text{min}} = \pi/4, \) and \( \lambda_{\text{max}} = 7\pi/4. \) Here \( N_\varphi \) and \( N_\lambda \) denote the total grid points in the latitude and longitude directions, respectively.

As an overset grid, the interpolation is needed between the Yin and Yang grid to communicate data. For scalar variables, the data transfer between the Yin and Yang grids is straightforward as long as the interpolation in the overlapping region is properly constructed. However, extra attention must be paid when one communicates a vector field since the expressions of a vector in the Yin and Yang spherical coordinates are different.

The transformation of a horizontal vector \( \mathbf{v} = (u, v) \) between Yin and Yang components is given as
\[
\begin{pmatrix}
u^e \\ u^e
\end{pmatrix} = \mathbf{P} \begin{pmatrix}
u^g \\ u^g
\end{pmatrix},
\]
where the projection matrix \( \mathbf{P} \) is
\[
\mathbf{P} = \begin{pmatrix}
-\sin \lambda^e \sin \lambda^g & -\cos \lambda^g / \cos \varphi^g \\
\cos \lambda^g / \cos \varphi^g & -\sin \lambda^e \sin \lambda^g
\end{pmatrix}.
\]

Because of the symmetry in the two components, the inverse transformation from Yin into Yang is
\[
\begin{pmatrix}
u^g \\ u^g
\end{pmatrix} = \mathbf{P}^{-1} \begin{pmatrix}
u^e \\ u^e
\end{pmatrix},
\]
where
\[
\mathbf{P}^{-1} = \begin{pmatrix}
-\sin \lambda^g \sin \lambda^e & -\cos \lambda^e / \cos \varphi^e \\
\cos \lambda^e / \cos \varphi^e & -\sin \lambda^g \sin \lambda^e
\end{pmatrix}.
\]

It is worth noting that each component grid is nothing but part of the conventional longitude–latitude grid, so numerical techniques developed for the latter can be transplanted to the Yin–Yang grid very conveniently. Moreover, since only the lower-latitude part of the longitude–latitude grid is used for both the Yin and Yang components, the grid spacing of the Yin–Yang grid is quite uniform. In fact, the ratio of the minimum and the maximum grid spacing is approximately 0.707.

Bearing in mind that the Yin and Yang components are identical to a longitude–latitude coordinate, we limit our discussions in the following sections only to the Yang grid.

The basic idea of the CIP/MM method is that one can make use of more than one kind of moment to describe the spatial variation of a given physical field and predict all the moments in time as model variables. In the present paper, we use two kinds of moments (i.e., the VIA and the PV).

Considering a general two-dimensional control volume (mesh cell) \( \Omega_{ij} = [\xi_{i-1/2}, \xi_{i+1/2}] \times [\eta_{j-1/2}, \eta_{j+1/2}] \) shown in Fig. 2, two kinds of moments are defined, respectively, for the field variable \( \Phi(\xi, \eta) \). When applying this to the longitude–latitude grid, one just needs to switch \( \xi \) to \( \lambda \) and \( \eta \) to \( \varphi \).

The VIA is defined over the control volume \( \Omega_{ij} \) as
\[
\nabla \Phi_{ij} = \frac{1}{\Delta \xi_i \Delta \eta_j} \int_{\eta_j-1/2}^{\eta_j+1/2} \int_{\xi_i-1/2}^{\xi_i+1/2} \Phi(\xi, \eta) \, d\xi \, d\eta, \quad \text{where} \quad \Delta \xi_i = \xi_{i+1/2} - \xi_{i-1/2} \quad \text{and} \quad \Delta \eta_j = \eta_{j+1/2} - \eta_{j-1/2}.
\]

Eight PVs are defined at the four vertices and the middle points of the four boundary edges of \( \Omega_{ij} \) as
\[
\nabla \Phi_{i,j}^e = \Phi(\xi_i, \eta_j),
\]
with \((\xi, \eta)\) being \((i - \frac{1}{2}, j - \frac{1}{2}), (i + \frac{1}{2}, j - \frac{1}{2}), (i - \frac{1}{2}, j), (i + \frac{1}{2}, j), (i - \frac{1}{2}, j + \frac{1}{2}), (i, j + \frac{1}{2})\), and \((i + \frac{1}{2}, j + \frac{1}{2})\), respectively.

Given one VIA and eight PVs as in Eqs. (11) and (12), a 2D quadratic polynomial for interpolation reconstruction.
\[ F_{i,j}(\xi, \eta) = C_{00} + C_{10}(\xi - \xi_{i-1/2}) + C_{20}(\xi - \xi_{i-1/2})^2 + C_{01}(\eta - \eta_{j-1/2}) + C_{02}(\eta - \eta_{j-1/2})^2 + C_{11}(\xi - \xi_{i-1/2})(\eta - \eta_{j-1/2}) + C_{12}(\xi - \xi_{i-1/2})(\eta - \eta_{j-1/2})^2 + C_{21}(\xi - \xi_{i-1/2})^2(\eta - \eta_{j-1/2}) + C_{22}(\xi - \xi_{i-1/2})^2(\eta - \eta_{j-1/2})^2, \] (13)

can be built over the single cell \( \Omega_{ij} \). The coefficients of interpolation function can be determined immediately by using the constraint conditions. Further details may be found in the appendix.

The piecewise interpolation reconstruction function \( F_{i,j}(\xi, \eta) \) will be then used in finding the departure point values in the semi-Lagrangian updating and other spatial discretizations. As will be seen later, defining two kinds moments on a single cell does not only allow us to construct high-order reconstruction interpolation with compact mesh stencil, but more importantly, it provides us with a framework to compute the PVs efficiently through the semi-Lagrangian method and to update the VIA with numerical conservativeness.

4. Time integration for the shallow-water equations

As discussed above, the CIP/MM FVM keeps both VIA and PV as the model variables that need to be predicted in time. We use different forms of the shallow-water equations as the governing equations for each of them (i.e., the advective form for the PV moment and the conservative form for the VIA moment).

a. The governing equations for PV and VIA moments

The advective form (or primitive form) of the shallow-water equations in longitude–latitude grid are written as

\[
\frac{du}{dt} = - \frac{1}{a \cos \phi} \frac{\partial}{\partial \lambda} (gh - cD) + fu + N_u, \tag{14}
\]

\[
\frac{dv}{dt} = - \frac{1}{a \cos \phi} (gh - cD) - fu + N_v, \quad \text{and} \quad \tag{15}
\]

\[
\frac{dh_w}{dt} = -h_w D, \tag{16}
\]

where \( a \) is the Earth radius. The horizontal wind components along the latitude and longitude are defined as \( u = (d/dt)(a \cos \phi) \) and \( v = (d/dt)(a \phi) \), respectively. The divergence \( D \) is expressed by

\[
D = \frac{1}{a \cos \phi} \left[ \frac{\partial u}{\partial \lambda} + \frac{\partial}{\partial \phi} (v \cos \phi) \right].
\]

The substantial derivative is given by

\[
\frac{d}{dt}() = \frac{\partial}{\partial t}() + \frac{u}{a \cos \phi} \frac{\partial}{\partial \lambda}(()) + \frac{v}{a} \frac{\partial}{\partial \phi}().
\]

Following McDonald and Bates (1989), we include the divergence damping in the equations for numerical stability, and the damping coefficient is generally set as \( 6.8 \times 10^6 \text{ m}^2 \text{ s}^{-1} \) unless it is specified otherwise. Here \( N_u \) and \( N_v \) are the spherical metric terms (i.e., the curvature term) given by

\[
N_u = \frac{uv \tan \phi}{a} \quad \text{and} \quad N_v = -\frac{u^2 \tan \phi}{a}.
\]

Here we denote \( h_w \) as the depth of fluid and \( h \) is the total height of the free surface. In case the effect of the topography \( h_s(\lambda, \phi) \) is included, the total height of the water surface is \( h = h_w + h_s \).

The conservative form of the shallow-water equations in longitude–latitude grid is represented (Rossmanith et al. 2004; Rossmanith 2006) as

\[
\frac{\partial}{\partial t} \left( \sqrt{G h_w} \right) + \frac{\partial (ah_w u)}{\partial \lambda} + \frac{\partial (a \cos \phi h_w v)}{\partial \phi} = 0, \tag{17}
\]

\[
\frac{\partial}{\partial t} \left( \sqrt{G h_w u^2} \right) + \frac{\partial}{\partial \lambda} \left[ \left( h_w u^2 + \frac{1}{2} gh_w^2 \right) \cos \phi \right] + \frac{\partial (h_w u v)}{\partial \phi} = -(fa + 2u \tan \phi) h_w v - gh_w \frac{1}{\cos \phi} \frac{\partial h_p}{\partial \lambda}, \tag{18}
\]

\[
\frac{\partial (\sqrt{G h_w u^2})}{\partial t} + \frac{\partial (h_w u v)}{\partial \lambda} + \frac{\partial}{\partial \phi} \left[ \cos \phi \left( h_w v^2 + \frac{1}{2} gh_w^2 \right) \right] = -(fa \cos \phi + u \sin \phi) h_w u - \frac{1}{2} gh_w^2 \sin \phi - gh_w \cos \phi \frac{\partial h_p}{\partial \phi}, \tag{19}
\]
where $\sqrt{G}h$, $\sqrt{G}h_u u^i$, and $\sqrt{G}h_u u^j$ are the conservative variables whose VIA values are updated by flux-based finite-volume formulations given later. The contravariant velocity components are $u^i = (u / a \cos \varphi)$ and $u^j = (v / a)$ with the metric factor being $\sqrt{G} = a^2 \cos \varphi$.

b. The semi-implicit semi-Lagrangian procedure for PVs

Since the pioneering work of Wiin-Nielsen (1959), the semi-Lagrangian scheme has received much attention for its superiority in computational stability and efficiency for large time steps. Robert (1981, 1982) has obtained stable integrations for the shallow-water equations of different forms with large time steps, using a semi-Lagrangian semi-implicit method. Since then, many researchers such as Ritchie (1985) and McDonald and Bates (1987, 1989) have investigated the numerical accuracy, computational stability, and efficiency of the semi-implicit semi-Lagrangian scheme. Staniforth and Côté (1991) gave a detailed review of this method and its applications to numerical models for atmospheric dynamics. In the present paper, we use the semi-implicit semi-Lagrangian solution to update the PVs.

Following the procedure used by McDonald and Bates (1989), we integrate the momentum equations [Eqs. (14) and (15)] in two half time steps of $\Delta t / 2$. In the first half-step (i.e., the semi-Lagrangian step) the Coriolis terms are treated implicitly while the pressure gradient and the divergence damping terms are treated explicitly. The reason is that we can easily solve the half-time-level momentum directly by way of semi-Lagrangian method, so it is called the semi-Lagrangian step. The second half-step, which is called the semi-implicit step, starts with those terms obtained from the first half-step. The curvature terms are integrated for a single time step $\Delta t$ in the first step. The continuity equation [Eq. (16)] is integrated in a full step of $\Delta t$.

Let $\bar{h}$ be a constant reference height such that $h = \bar{h} + h'$ and $|h'| \ll \bar{h}$. In the all test cases presented in this paper, we set $\bar{h}$ as $h_0$ in the initial condition for height. Sorting all the $(n + 1)$th time-level quantities to the left-hand side in the resulting equations we have

$$\left[ u + \frac{\Delta t}{2} \frac{1}{a \cos \varphi} \frac{\partial}{\partial \lambda} (gh - cD) \right]^{n+1} = B_u,$$  \hspace{1cm} (20)

$$\left[ v + \frac{\Delta t}{2} \frac{1}{a \sin \varphi} (gh - cD) \right]^{n+1} = B_v,$$  \hspace{1cm} (21)

and

$$\left[ h_w + \frac{\Delta t}{2} \bar{h} D \right]^{n+1} = B_h,$$  \hspace{1cm} (22)

where

$$B_u = \left[ (W_u)^n + \mathcal{F}(W_u)^n \right]_t,$$

$$B_v = \left[ (W_v)^n + \mathcal{F}(W_v)^n \right]_t,$$

and

$$B_h = \left[ (h_w)^n - \left( \frac{\Delta t}{2} \bar{h} + \Delta t \dot{h} \right) \right]_t,$$

with

$$W_u = \frac{Y_u + \mathcal{F} Y_v}{1 + f^2},$$

$$W_v = \frac{Y_v + \mathcal{F} Y_u}{1 + f^2},$$

$$Y_u = u - \frac{\Delta t}{2} \frac{1}{a \cos \varphi} \frac{\partial}{\partial \lambda} (gh - cD) + \Delta t N_u,$$

$$Y_v = v - \frac{\Delta t}{2} \frac{1}{a \sin \varphi} (gh - cD) + \Delta t N_v,$$

and

$$\mathcal{F} = f \frac{\Delta t}{2}.$$

In the above expressions, the subscript “$d$” denotes the position at the departure point where the corresponding value of the physical field should be obtained from the multimoment reconstruction described in section 3.

From Eqs. (20) and (21) we find

$$D^{n+1} = \nabla \cdot \mathbf{B} = \frac{\Delta t}{2} \nabla^2 (gh - cD)^n + 1,$$  \hspace{1cm} (23)

where $\mathbf{B} = (B_u, B_v)$. Eliminating $D^{n+1}$ between Eqs. (22) and (23) leads to the Helmholtz equation

$$\nabla^2 h_w^{n+1} - \mu^2 h_w^{n+1} = H,$$  \hspace{1cm} (24)

where

$$\mu^2 = \left[ \left( \frac{\Delta t}{2} \right)^2 g \bar{h} + c \left( \frac{\Delta t}{2} \right) \right]^{-1}$$

and

$$H = \mu^2 \left[ \left( \frac{\Delta t}{2} \bar{h} \right) \nabla \cdot \mathbf{B} + \left( \frac{\Delta t}{2} \nabla^2 - 1 \right) B_h - \left( \frac{\Delta t}{2} \bar{h} \nabla^2 (gh) \right) \right].$$

The spatial discretization for above semi-implicit semi-Lagrangian formulation is carried out completely in terms of the PVs. The great circle algorithm of McDonald and Bates (1989) is used to determine the departure location and the detailed procedures can be found in Li et al. (2006). For the eight PV moments (black circle) shown in Fig. 2, we can directly utilize the interpolation function $F_{ij} (\lambda, \varphi)$ constructed in the previous section to obtain the velocity components $u$, $v$, $w$. 
and the height $h$ field at the departure point. For the value at the triangle point $r^\theta \Phi_{i,j}^n$ in Fig. 2, which is not a model variable, we should evaluate its value at the departure point by using the same interpolation function $F_{i,j} (\lambda, \varphi)$ since we will need it in the semi-implicit semi-Lagrangian solution. For the pressure gradient term, we first utilize central differencing to evaluate the gradient in the arrival point at the current time level, then use a third-order Lagrangian interpolation function (Li et al. 2006) based on 16 grid points to obtain the gradient value at the departure point. At this step, the point value at the cell center is involved as other PVs and should be computed from the multimoment reconstruction. We have then obtained the right-hand side terms in the prognostic equations [Eqs. (20)–(22)].

With all the values at the departure points known, we can obtain the right-hand side of the Helmholtz equation $H$ in Eq. (24).

The Laplacian operator in Eq. (24) in a longitude–latitude grid is

$$\nabla^2 = \frac{1}{a^2 \cos^2 \varphi} \frac{\partial^2 (\cdot)}{\partial \lambda^2} + \frac{1}{a^2 \cos \varphi} \frac{\partial}{\partial \varphi} \left[ \cos \varphi \frac{\partial (\cdot)}{\partial \varphi} \right].$$

Shown in Fig. 3, a five-point central differencing in terms of $r^\theta \Phi_{i-1,j}^n$, $r^\theta \Phi_{i,j}^n$, $r^\theta \Phi_{i+1,j}^n$, $r^\theta \Phi_{i,j-1}^n$, and $r^\theta \Phi_{i,j+1}^n$ is used to approximate the second derivative at the $(i,j)$ point. Because of the uniform spacing in respect to $\lambda$ and $\varphi$, we have

$$\begin{align*}
\left( \frac{\partial^2 \Phi}{\partial \lambda^2} \right)_{i,j} &= \frac{r^\theta \Phi_{i+1,j}^n + r^\theta \Phi_{i-1,j}^n - 2r^\theta \Phi_{i,j}^n}{\Delta \lambda^2} \\
\left[ \frac{\partial}{\partial \varphi} \left( \cos \varphi \frac{\partial \Phi}{\partial \varphi} \right) \right]_{i,j} &= \frac{\cos \varphi_{i,j+1/2} - \cos \varphi_{i,j-1/2}}{\Delta \varphi}.
\end{align*}$$

Note that the trigonometric functions $\cos \varphi_{i,j+1/2}$ and $\cos \varphi_{i,j-1/2}$ are computed at the middle points shown as the triangle marks in the $\varphi$ direction in Fig. 3.

The discretized Helmholtz equation has to be solved over the whole sphere for a globally converged numerical solution. Extra attention must be paid in the overlapping region of the Yin–Yang grid. As shown in Fig. 4, when we discretize the Helmholtz equation at a boundary point $N$ in the Yang component, for example, the value of $\Phi$ at point $M$ that falls in the Yin component (indicated by the dashed lines) is needed. The value of $\Phi$ out of the boundary (the solid line) is interpolated by a third-order Lagrangian interpolation function (Li et al. 2006) based on the 16 points from another component grid (Yin grid). The same applies when Helmholtz equation is solved on the Yin grid and requires the value at a point falling in the Yang grid.

The Helmholtz equation is solved iteratively by the classical Schwarz method on the Yin–Yang grid with the boundary data exchanged at every step through the interpolation. Thus, the equivalent overlap subdomain in context of the Schwarz method is “one”. In our computations, we require the residual for both the Yin and Yang grids to be less than $10^{-8}$. It is noted that Qadouri et al. (2007) recently introduced an optimized Schwarz methods in the Yin–Yang grid by using improved transmission conditions across the Yin–Yang boundary.

Successive overrelaxation (SOR; Press et al. 1992) is used to solve the Helmholtz equation on each component of the Yin–Yang grid, respectively, in the present model. As expected, the iteration number needed for convergence depends on the problem size. Shown in Table 1, more iteration number is required for simulation with higher resolution. Other more scalable solver, such as the multigrid methods (Wesseling 1991) should...
be more demanding in large-scale simulation. We summarize the procedure for getting numerical solution to the Helmholtz equation [Eq. (24)] on the Yin–Yang grid as follows:

1) Calculate the values at the boundary points (e.g., point M in Fig. 4) of the Yang grid from the Yin grid by interpolation.

2) Use SOR iterative method to solve the Helmholtz equation in the Yang component to update the numerical solution of the Helmholtz equation for one step.

3) Interpolate the values at the boundary points for the Yin grid from the Yang grid by interpolation in a similar manner.

4) Similar to step 2, iterate the Helmholtz equation in the Yin component grid to update the numerical solution for one step.

5) Repeat steps 1–4 until the numerical solution converges on both the Yin and Yang grids.

Once we get the water height field \( h_i^{n+1} \) as the solution of the Helmholtz equation at the next time level, we can evaluate the velocity components \( u_i^{n+1} \) and \( v_i^{n+1} \) at the next time level by Eqs. (20) and (21).

c. The flux-based finite-volume method for VIA

As mentioned before, the VIA moment is updated in terms of the flux form shallow-water equation system in Eqs. (17)–(19). We recast them into a vector form as

\[
\frac{\partial \mathbf{U}}{\partial t} = - \frac{\partial \mathbf{F}(\mathbf{U})}{\partial \lambda} - \frac{\partial \mathbf{G}(\mathbf{U})}{\partial \varphi} + \mathbf{S}(\mathbf{U}),
\]  

where

\[
\mathbf{U} = \begin{bmatrix} a^2 \cos\varphi h_v \\ ah_u u \\ a \cos\varphi h_v v \end{bmatrix}, \quad \mathbf{F}(\mathbf{U}) = \begin{bmatrix} ah_u u \\ h_u u^2 + \frac{1}{2} g h_w^2 / \cos\varphi \\ h_u u v \end{bmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{bmatrix} a \cos\varphi h_v v \\ h_u u v \\ \cos\varphi \left( h_u v^2 + \frac{1}{2} g h_w^2 \right) \end{bmatrix}, \text{ and} \\
\mathbf{S}(\mathbf{U}) = \begin{bmatrix} 0 \\ (fa + 2u \tan\varphi) h_v v - gh_w \ \frac{1}{\cos\varphi} \frac{\partial h_v}{\partial \lambda} \\ -(fa \cos\varphi + u \sin\varphi) h_v u - \frac{1}{2} g h_w^2 \sin\varphi - gh_v \cos\varphi \ \frac{\partial h_v}{\partial \varphi} \end{bmatrix}.
\]

We have the VIA within the domain \( \Omega_{i,j} \) as

\[
\nabla \mathbf{U}_{i,j} = \frac{1}{\Delta \lambda_i \Delta \varphi_j} \int_{\lambda_{i-1/2}}^{\lambda_{i+1/2}} \int_{\varphi_{j-1/2}}^{\varphi_{j+1/2}} \mathbf{U}(\lambda, \varphi, t) \, d\lambda \, d\varphi.
\]  

The governing equations for VIA are obtained by integrating Eq. (28) over domain \( \Omega_{i,j} \):

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{Resolution} & \text{Case 2} & \text{Case 5} & \text{Case 6} & \text{Case 7} \\
\hline
16 \times 48 \times 2 & 10/11 & 13/13 & 14/15 & 15/15 \\
32 \times 96 \times 2 & 17/19 & 26/26 & 30/31 & 31/32 \\
64 \times 192 \times 2 & 38/43 & 72/69 & 84/87 & 88/92 \\
\hline
\end{array}
\]
VIA as above, a second-order time integration can be used for the semi-implicit semi-Lagrangian procedure discussed in Sect. 3. Lauritzen et al. (2006) presented a mass-conservative method for the semi-Lagrangian semi-implicit model by using a vol-

ume-conserving advection in a single-moment model. Relative to the conventional semi-Lagrangian/semi-implicit scheme, extra computa-
tional cost is required for the numerical fluxes in Eq. (30).

\[
\begin{align*}
\nabla \mathbf{U}^{n+1} &= \nabla \mathbf{U}^{n} + \frac{\Delta t}{2} [\mathcal{L}(\mathbf{U})^{n+1}_{i,j} + \mathcal{L}(\mathbf{U})^{n}_{i,j}],
\end{align*}
\]

where \( \mathcal{L}(\mathbf{U})^{n}_{i,j} \) and \( \mathcal{L}(\mathbf{U})^{n+1}_{i,j} \) means the right-hand sides of Eq. (30) computed with the PVs at the \( n \)th and the \( (n + 1) \)th step, respectively.

Because we have to keep both PV and VIA independently as the model variables, the present model requires twice as much as the memory storage needed in a single-moment model. Relative to the conventional semi-Lagrangian/semi-implicit scheme, extra computation cost is required for the numerical fluxes in Eq. (30). Given the PV moments that are updated at an expense, which is almost the same as in a conventional semi-Lagrangian scheme, the numerical fluxes can be explicitly computed. In Table 2, we show the CPU time consumed by different parts of the model for different resolutions. Updating VIA takes only a small portion of the total CPU time.

As discussed above, adding a conserved moment (i.e., the volume integrated value) provides us a convenient way to enforce the numerical conservativeness in a semi-Lagrangian semi-implicit model. Being a general methodology, it should be able to apply to other semi-Lagrangian semi-implicit formulations as well. As another particular way to introduce the conservation to a semi-Lagrangian semi-implicit model, we note that Lauritzen et al. (2006) presented a mass-conservative semi-Lagrangian semi-implicit model by using a volume-mapping advection transport scheme.

\[
\begin{align*}
\frac{\partial \mathbf{U}^{n}_{i,j}}{\partial t} &= -\frac{1}{\Delta \lambda_i \Delta \varphi_j} \left\{ \int_{\varphi_{j-1/2}}^{\varphi_{j+1/2}} [\mathbf{F}(\mathbf{U})_{i,j-1/2} - \mathbf{F}(\mathbf{U})_{i,j-1/2}] \, d\varphi + \int_{\lambda_{i-1/2}}^{\lambda_{i+1/2}} [\mathbf{G}(\mathbf{U})_{i-1/2} - \mathbf{G}(\mathbf{U})_{i-1/2}] \, d\lambda \right\} \\
&+ \frac{1}{\Delta \lambda_i \Delta \varphi_j} \int_{\lambda_{i-1/2}}^{\lambda_{i+1/2}} \mathbf{S}(\mathbf{U}) \, d\lambda \, d\varphi = \mathcal{L}(\mathbf{U})_{i,j},
\end{align*}
\]

Given all the PVs on the control volume boundary, the integrals over the boundary segments in Eq. (30) are computed by the three-point quadrature formula:

\[
\int_{\varphi_{j-1/2}}^{\varphi_{j+1/2}} \mathbf{F}(\mathbf{U})_{i,j-1/2} \, d\varphi = \frac{\Delta \varphi_j}{6} [\mathbf{F}(\nabla \mathbf{U})_{i-1/2,j-1/2} + 4\mathbf{F}(\nabla \mathbf{U})_{i-1/2,j} + \mathbf{F}(\nabla \mathbf{U})_{i-1/2,j+1/2}].
\]

The area integral of source term is computed by the following nine-point quadrature formula:

\[
\int_{\lambda_{i-1/2}}^{\lambda_{i+1/2}} \mathbf{S}(\mathbf{U}) \, d\lambda \, d\varphi = \Delta \lambda_i \Delta \varphi_j \left\{ \frac{1}{96} [\mathbf{S}(\nabla \mathbf{U})_{i-1/2,j-1/2} + \mathbf{S}(\nabla \mathbf{U})_{i+1/2,j-1/2} + \mathbf{S}(\nabla \mathbf{U})_{i-1/2,j+1/2} + \mathbf{S}(\nabla \mathbf{U})_{i+1/2,j+1/2} + \mathbf{S}(\nabla \mathbf{U})_{i,j-1/2} + \mathbf{S}(\nabla \mathbf{U})_{i,j+1/2}] + \frac{4}{9} \mathbf{S}(\nabla \mathbf{U})_{i,j} \right\}.
\]

5. The advection scheme

For completeness, we briefly describe the advection scheme used in the numerical tests in this paper.

The two-dimensional flux-form advection equation for any scalar \( h \) can be expressed as

\[
\frac{\partial h}{\partial t} + \frac{\partial \left( \mathbf{u} h \right)}{\partial \lambda} + \frac{\partial \left( \mathbf{v} h \right)}{\partial \mu} = 0
\]

on the sphere, where

\[
\mathbf{u} = \frac{u \cos \varphi}{a(1 - \mu^2)}, \quad \mathbf{v} = \frac{v \cos \varphi}{a}, \quad \text{and} \quad \mu = \sin \varphi.
\]

As discussed above, both the VIA and PV as the model variables need to be predicted in time.

We update the PV moment by a semi-Lagrangian scheme, considering the advection form of Eq. (32):

\[
\frac{dh}{dt} = -D,
\]

where \( D \) denotes the divergence as given before.

With the interpolation function in Eq. (13) constructed in terms of both PV and VIA moments at time step \( n \), one can predict the PVs on the cell boundary by

\[
\mathbf{PV}^{n+1}(\lambda, \varphi) = \mathbf{PV}(\lambda - \delta \lambda_c, \varphi - \delta \varphi_c) - \int_C D \, dt,
\]

where subscript “up” denotes the piecewise interpolation function for the cell in which the departure point \((\lambda_c - \delta \lambda_c, \varphi - \delta \varphi_c)\) falls, and \( C \) is the trajectory connecting the departure and arrival points. The displacement \((\delta \lambda_c, \delta \varphi_c)\) is computed according to the velocity field. In the present model the great circle algorithm of
Table 2. Elapsed time in seconds per step. The hardware platform is a single node (single CPU of 1.9 GHz, 8 GB memory) of an IBM cluster1600 (AIX, version 5). “Helmholtz Eq.,” “PV,” “VIA,” “Other,” and “Tot” denote the time for solving Helmholtz equation, updating PV moments (including the computation of Helmholtz equation), updating the VIA moment, the other part, and the total time per step, respectively.

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Helmholtz Eq.</th>
<th>PV</th>
<th>VIA</th>
<th>Other</th>
<th>Tot</th>
</tr>
</thead>
<tbody>
<tr>
<td>16 × 48 × 2</td>
<td>0.11</td>
<td>0.18</td>
<td>0.01</td>
<td>0.05</td>
<td>0.24</td>
</tr>
<tr>
<td>32 × 96 × 2</td>
<td>0.34</td>
<td>0.59</td>
<td>0.01</td>
<td>0.15</td>
<td>0.75</td>
</tr>
<tr>
<td>64 × 192 × 2</td>
<td>1.17</td>
<td>2.14</td>
<td>0.04</td>
<td>0.50</td>
<td>2.68</td>
</tr>
</tbody>
</table>

McDonald and Bates (1989) is used to determine the departure location.

Given all the PVs for both the nth and (n + 1)th time steps on the control volume boundary, the VIA moment of the advected field \( ^{TV}_{n+1} \) can be computed by a finite-volume formulation in the same exact way as the first equation of Eq. (30).

The interpolation for communicating data across the Yin–Yang boundary in the advection computations is conducted in the same manner as previously discussed.

\[
\begin{align*}
\ell_1(h) &= \frac{I[[h(\lambda, \phi) - h_T(\lambda, \phi)]]}{I[[h_T(\lambda, \phi)]]}, \\
\ell_2(h) &= \frac{\{I[[h(\lambda, \phi) - h_T(\lambda, \phi)]^2]\}^{1/2}}{I[[h_T(\lambda, \phi)]^{2}]^{1/2}}, \quad \text{and} \\
\ell_\infty(h) &= \frac{\max[h_{\text{yang},T}(\lambda, \phi) - h_{\text{yang}}(\lambda, \phi), |h_{\text{yin}}(\lambda, \phi) - h_{\text{yin},T}(\lambda, \phi)|]}{\max[h_{\text{yang},T}(\lambda, \phi), |h_{\text{yin},T}(\lambda, \phi)|]},
\end{align*}
\]

where \( h_{\text{yang},T}(\lambda, \phi) \) and \( h_{\text{yin},T}(\lambda, \phi) \) are the true solutions on the Yin–Yang grids, respectively. Note that Eq. (34) indicates that the integral over the overset region of the Yin–Yang grid is computed twice.

6. Numerical tests

According to Williamson et al. (1992), we define the global integral \( I \) as

\[
I(h) = \frac{1}{4\pi} \int_{\varphi_1}^{\varphi_2} \int_{\varphi_3}^{\varphi_4} h_{\text{yin}}(\lambda, \varphi) \cos \varphi \, d\varphi \, d\lambda \\
+ \frac{1}{4\pi} \int_{\varphi_1}^{\varphi_2} \int_{\varphi_3}^{\varphi_4} h_{\text{yang}}(\lambda, \varphi) \cos \varphi \, d\varphi \, d\lambda,
\]

and the normalized global errors as

\[
\ell_1 = \frac{I[[h(\lambda, \varphi) - h_T(\lambda, \varphi)]]}{I[[h_T(\lambda, \varphi)]]}, \quad \ell_2 = \frac{\{I[[h(\lambda, \varphi) - h_T(\lambda, \varphi)]^2]\}^{1/2}}{I[[h_T(\lambda, \varphi)]^{2}]^{1/2}}, \quad \text{and} \quad \ell_\infty = \frac{\max[h_{\text{yang},T}(\lambda, \varphi) - h_{\text{yang}}(\lambda, \varphi), |h_{\text{yin}}(\lambda, \varphi) - h_{\text{yin},T}(\lambda, \varphi)|]}{\max[h_{\text{yang},T}(\lambda, \varphi), |h_{\text{yin},T}(\lambda, \varphi)|]},
\]

where the parameter \( \alpha \) is the angle between the axis of the solid-body rotation and the polar axis of the spherical coordinate system. Tests in this paper were run with \( \alpha = 0, \alpha = \pi/4, \alpha = \pi/2, \) and \( u_0 = 2\pi a/(12 \text{ days}) \).

We carried out global advection tests with the resolution of 2.8125° × 2.8125° (equivalent to a 32 × 96 × 2 grid). The Courant–Friedrichs–Lewy (CFL) number is specified as 0.5. The normalized errors are shown in Tables 3 for different angles. It reveals that the Yin–Yang grid can get accurate advections along different paths on the sphere. The transported concentration and difference between the numerical solution and exact solution are shown in Fig. 5 for \( \alpha = \pi/4 \). The time evolution of the corresponding \( \ell_1, \ell_2, \) and \( \ell_\infty \) errors for \( \alpha = \pi/4 \) and \( \alpha = \pi/2 \) are also given in Fig. 5.

We compared our numerical results with other existing global advection schemes. For the convenience of intercomparison, we referenced the numerical results from other schemes, such as the semi-Lagrangian tran-