A New Algorithm with Square Conservative Exponential Integral for Transport Equation

SHIFENG HAO,[a,b] XIAOPENG CUI,[c,d,e] AND JIANPING HUANG[f]

[a] *Zhejiang Meteorological Observatory, Hangzhou, Zhejiang, China*
[b] *Heavy Rain and Drought-Flood Disasters in Plateau and Basin Key Laboratory of Sichuan Province, Chengdu, SiChuan, China*
[c] *Key Laboratory of Cloud-Precipitation Physics and Severe Storms (LACS), Institute of Atmospheric Physics, Chinese Academy of Sciences, Beijing, China*
[d] *Collaborative Innovation Center on Forecast and Evaluation of Meteorological Disasters, Nanjing University of Information Science and Technology, Nanjing, China*
[e] *University of Chinese Academy of Sciences, Beijing, China*
[f] *NOAA/NCEP/Environmental Modeling Center, I. M. System Group, College Park, Maryland*

(Manuscript received 13 February 2020, in final form 25 October 2020)

ABSTRACT: The square conservative exponential integral method (SCEIM) is proposed for transport problems on the sphere. The method is a combination of the square conservation algorithm and the exponential integral method. The main emphasis in the development of SCEIM is on conservation, positive-definite, and reversibility as well as achieving comparable accuracy to other published schemes. The most significant advantage of SCEIM is to change the forward model to the backward model by setting a negative time step, and the backward model can be used to solve the inverse problem. Moreover, the polar problem is significantly improved by using a simple effective central skip-point difference scheme without major penalty on the overall effectiveness of SCEIM. To demonstrate the effectiveness and generality of the SCEIM, this method is evaluated by standard cosine bell tests and deformational flow tests. The numerical results show that SCEIM is a time-convergence method as well as a grid-convergence method, and has a strong shape-preserving ability. In the tests of the inverse problem, the sharp fronts are successfully regressed back into their initial weak fronts and the cosine bells move against the wind direction and return to the initial position with high accuracy. The numerical results of forward simulations are compared with those of published schemes, the total mass conservation, and error norms are competitive in term of accuracy.

KEYWORDS: Transport; Conservation of mass; Inverse methods

1. Introduction

The time integration scheme is an important consideration in the design of an atmospheric model. Stability, accuracy and efficiency are obvious requirement for any numerical method, particularly for numerical weather prediction in an operational context. Exponential integration methods (EIM) have received increased attention as an alternative to multistep and multistage time integration methods commonly used in conjunction with finite difference method (FDM) and finite element methods. EIMs are exact for linear systems of ODEs with constant coefficients and are thus unconditionally stable. EIMs have the advantage of being accurate and, similarly to traditional known implicit methods, allowing integration with large time steps and possessing good stability properties. Beylkin et al. (1998) extended EIMs to solving nonlinear partial differential equations (PDEs) by splitting nonlinear PDEs into linear and nonlinear parts, this is known as exact treatment of the linear part (ELP) scheme, whose distinctive feature is the exact evaluation of the contribution of the linear term of nonlinear PDEs. An alternative to ELP is the exponentially fitted Euler method, which require the evaluation of a particular family of exponential function of matrix, namely, the φ functions, and this method also commonly appears in the literature, for example, Hochbruck et al. (1998) and Tokman (2006). Although EIMs offer the potential for high accuracy, the evaluation can be very computationally expensive due to requiring computation of the matrix exponential involved in EIMs. Approximation of such matrix functions or their products with vectors constitutes the main computational cost of an exponential integrator. This is of particular concern in the solution of PDEs, where the spatial discretization results in very large matrices. A number of methods have been proposed to calculate the exponential functions of a matrix (Moler and Van Loan 1978, 2003). Most of them, however, are of little practical use for large-scale stiff matrices due to high computational cost. It is common, however, that only the product of exponential functions of matrix and vectors is required for an implementation of exponential integral. This is the case, for instance, when the matrix results from a spatial discretization of the atmosphere or ocean dynamical system. To deal with this problem, Krylov subspace techniques are among the most promising methods. The basic idea of the Krylov subspace techniques is to project a large matrix onto a relatively small Krylov subspace where calculating the matrix is significantly less computationally expensive. By taking the advantages of this method, a variety of efficient Krylov subspace algorithms have been proposed and developed, such as restarted Krylov methods (Botchev 2013), block Krylov subspaces (Frommer et al. 2017; Li et al. 2016), the adaptive Krylov subspaces methods (Niesen and Wright 2012; Gaudreault et al. 2018) and the exponential propagation iterative methods(Tokman 2006).

Corresponding author: Dr. ShifengHao, shifenghao@aliyun.com

DOI: 10.1175/MWR-D-20-0049.1

© 2021 American Meteorological Society. For information regarding reuse of this content and general copyright information, consult the AMS Copyright Policy (www.ametsoc.org/PUBSReuseLicenses).
Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

Krylov subspace methods have been successfully used for the simulation of the shallow-water equations on the sphere (Clancy and Pudykiewicz 2013). Compared with the rational functions arising from the implicit approach, EIMs have an advantage over the traditional implicit approach due to faster Krylov convergence of exponential functions (Loffeld and Tokman 2013). This has provided strong motivation for the further development of these methods.

In the previous investigations, how to improve the computational efficiency of exponential functions of matrix has always been the focus of attention. In recent years, more and more attention has been paid to the positive-definite and conservation associated with EIMs (Bhatt and Moore 2017, 2019; Clancy and Pudykiewicz 2013). However, these two problems associated with finite difference method have been well solved and implemented by many methods. Among them, the conservation semi-Lagrangian with rational function (CSLR) scheme constitutes a typical example. With this technique, the piecewise rational method (PRM) was proposed to solve problems of mass conservation and positive-definiteness of semi-Lagrangian advection (Xiao and Peng 2004). For dynamic cores of numerical model with explicit schemes based on Euler equations, The nonlinear renormalization scheme was proposed to solve the positive-definiteness problem (Bott 1989), which was then greatly improved and applied successfully to improve the overpredicted cumulative precipitations of WRF model (Skamarock and Weisman 2009). In the present article, we investigate the potential of EIMs for solving transport equation, and emphasis on how to obtain conserving numerical solution that is also everywhere positive. Therefore, we have to mention an effective algorithm, namely, square conservation algorithm (SCA), which was proposed by Zeng and Ji in 1981. In general, if $\int_\Omega (\partial y/\partial t)\, d\Omega = 0$, where y is a time-dependent variable in space Ω, then y is a global conservative variable. Similarly, the concept of square conservation is that if $\int_\Omega y(\partial y/\partial t)\, d\Omega = 0$, then y is a global square conservative variable due to $y(\partial y/\partial t) = (1/2)\partial y^2/\partial t$. the atmospheric or oceanic dynamical equations may be transformed into square conservation systems under some conditions, then symplectic schemes can be used to solve them, and SCA can also be adopted to solve Hamiltonian system (Wang et al. 1995). Computational stability and energy conservation are very important for a long-time numerical integration. For this reason, some implicit and explicit schemes with square conservation have been developed successively, and applied successfully in numerical simulations of atmospheric and oceanic circulations (Zeng and Ji 1981; Ji and Zeng 1982; Ji and Wang 1991; Wang and Ji 1993; Wang et al. 1993, 2004). In recent years, a new algorithm with generalized square conservation algorithm is proposed to construct finite difference schemes for evolution equations, this algorithm can incorporate historical observation data, and perform very well in the linear advection tests (Gong et al. 2013a, 2013b). The SCA can be achieved by implicit or explicit schemes. The implicit schemes are nonlinear iterative methods, which require a large amount of computational cost, and the accuracy is not ensured. In contrast, the explicit scheme proposed by Wang et al. (1993, 1995) possesses notable advantages, but it is difficult to construct a positive-definite dissipative operator to achieve square conservation. Because the SCA is a nonlinear conservative algorithm, the finite difference method with discrete temporal coordinates is the major reason leading to difficulty in designing an explicit SCA. However, this is exactly the problem that can be solved by EIMs, so the aim of the present article is to develop a square conservative exponential integral method (SCEIM), which is explicit, conservative and positive-definite algorithm for transport equation, by combining SCA, and EIMs without constructing dissipative operator.

2. Square conservative exponential integral method (SCEIM)

a. *Square conservative property of transport equation*

The two-dimensional transport equation in conservation form on a sphere with radius r is given by

$$\frac{\partial \rho \cos\theta}{\partial t} = -\frac{\partial u\rho \cos\theta}{r\partial\lambda \cos\theta} - \frac{\partial v\rho \cos\theta}{r\partial\theta}, \tag{1}$$

where ρ is density of a transported scalar, which is everywhere nonnegative; t is time; θ and λ are latitude and longitude, respectively; and u and v are components of wind velocity vector along east and north directions, respectively. The values of ρ, u, and v are dependent of t, λ, and θ. Then the boundary conditions in θ and λ directions, respectively, are defined by $(u\rho\cos\theta)|_{\lambda=0} \equiv (u\rho\cos\theta)|_{\lambda=2\pi}$ and $(v\rho\cos\theta)|_{\theta=-\pi/2} \equiv (v\rho\cos\theta)|_{\theta=\pi/2} \equiv 0$. Since ρ is everywhere nonnegative, we can now set $(\rho^*)^2 = \rho\cos\theta$, then Eq.(1) can be expressed alternatively as

$$\frac{\partial \rho^*}{\partial t} = L_{\text{lat}}(\lambda,\theta,t) + L_{\text{lon}}(\lambda,\theta,t), \tag{2}$$

where

$$L_{\text{lat}}(\lambda,\theta,t) = \frac{-1}{2r\cos\theta}\left(\frac{\partial u\rho^*}{\partial\lambda} + u\frac{\partial \rho^*}{\partial\lambda}\right), \tag{3}$$

$$L_{\text{lon}}(\lambda,\theta,t) = \frac{-1}{2r}\left(\frac{\partial v\rho^*}{\partial\theta} + v\frac{\partial \rho^*}{\partial\theta}\right). \tag{4}$$

Here, note that ρ^* is allowed to go negative, but this negativity is irrelevant because ρ will always be positive. Multiplying Eq. (2) by $2r^2\rho^*$, we can obtain

$$r^2\frac{\partial(\rho^*)^2}{\partial t} = 2r^2\rho^*[L_{\text{lat}}(\lambda,\theta,t) + L_{\text{lon}}(\lambda,\theta,t)] \tag{5}$$

and then integrating the first term of the right-hand side of Eq. (5) in λ direction with the periodic boundary conditions, it yields

$$2r^2\int_0^{2\pi}\rho^* L_{\text{lat}}(\lambda,\theta,t)\, d\lambda = \frac{-r}{\cos\theta}\int_0^{2\pi}\frac{\partial u(\rho^*)^2}{\partial\lambda}d\lambda = 0. \tag{6}$$

By integrating the second term of the right-hand side of Eq. (5) in θ direction with the homogeneous boundary condition, it yields

$$2r^2\int_{-\pi/2}^{\pi/2}\rho^* L_{\text{lon}}(\lambda,\theta,t)\, d\theta = -r\int_{-\pi/2}^{\pi/2}\frac{\partial v(\rho^*)^2}{\partial\theta}d\theta = 0. \tag{7}$$

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

From Eqs. (6) and (7), we can obtain

$$r^2\int_0^{2\pi}\int_{-\pi/2}^{\pi/2}\frac{\partial(\rho^*)^2}{\partial t}d\theta\,d\lambda=\int_0^{2\pi}\int_{-\pi/2}^{\pi/2}\frac{\partial\rho}{\partial t}r^2\cos\theta\,d\theta\,d\lambda=0, \quad (8)$$

which means that $(\rho^*)^2$ satisfies total mass conservation. Since $(\rho^*)^2$ is the square of ρ^*, we say "ρ^* satisfies square conservation." With the definition of ρ^*, ρ is thus everywhere nonnegative.

b. *Square conservation property of discrete transport equation*

To develop a square conservation algorithm for a transported scalar in discretion form, we consider the spherical mesh with radius r defined by

$$\begin{cases}(\Delta\lambda,\Delta\theta)=\left(\dfrac{\pi}{I},\dfrac{\pi}{2J+1}\right)\\ i\in[0,2I],\quad j\in[-J,J]\\ (\lambda_i,\theta_j)=(i\Delta\lambda,j\Delta\theta)\\ \lambda_{2I}\equiv\lambda_0\equiv0,\quad \lambda_0-\Delta\lambda\equiv\lambda_{2I-1}\\ (\lambda_i,\theta_{\pm J}\pm\Delta\theta)\equiv(\lambda_{I+i},\theta_{\pm J}),\quad i\in[0,I)\\ (\lambda_i,\theta_{\pm J}\pm\Delta\theta)\equiv(\lambda_{i-I},\theta_{\pm J}),\quad i\in[I,2I]\end{cases}. \quad (9)$$

Here, I and J are integers, the indices of the grid points in λ and θ directions, respectively, are denoted by i and j. For any $i\in[0, I-1]$, because of $\lambda_{i+I}-\lambda_i=\pi$, the longitude λ_i line and the longitude λ_{i+I} line compose a closed longitude circle. Figure 1 shows the spherical mesh given by (9). Because of $\Delta\theta=\pi/(2J+1)$, we can have $\pi/2-|\theta_{\pm J}|=0.5\Delta\theta$, and then $\theta_{i,\pm J}\pm\Delta\theta=\theta_{I+i,\pm J}$, $0\leqq i<I$ and $\theta_{I+i,\pm J}\pm\Delta\theta=\theta_{i,\pm J}$, $I<i\leqq 2I$, which means that this spherical mesh system is a closed domain with periodic boundary conditions in both λ and θ directions. It is important to note that both the North and South Poles, which are singularities in the latitude–longitude coordinate system, are eliminated in this spherical mesh.

Here we use "Arakawa unstaggered grid A" (Arakawa 1972) and take second-order spatial central difference scheme as an example to demonstrate the square conservation of the discrete ρ^* in the grid system given by Eq. (9). For any $2P$-order spatial central

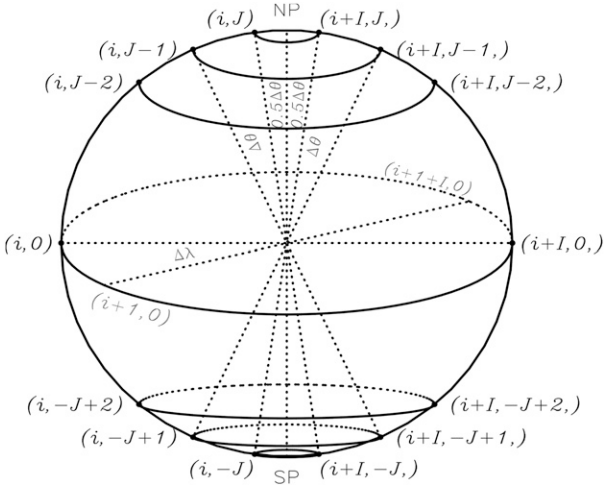

FIG. 1. Diagram of the latitude–longitude grid system. The NP and SP denote North Pole and South Pole, respectively.

difference scheme, it can be written as a linear combination like $\partial f_k/\partial x=(1/\Delta x)\sum_{p=1}^{P}C_p(f_{k+p}-f_{k-p})$, where f_k represents a discrete variable (u,v,ρ) on the kth grid point in $x\in(\lambda,\theta)$ direction and C_p is coefficient, which can be obtained by solving a polynomial of order $2P$. If $P=1$ and $C_1=1/2$ then $\partial f_k/\partial x$ is a second-order spatial central difference scheme. Therefore, the demonstrations for square conservation with higher-order central difference schemes are similar to that of the second-order scheme. According to the convention for discrete systems, we use u_{ij}, v_{ij}, and ρ_{ij} represent $u(\lambda_i,\theta_j,t)$, $v(\lambda_i,\theta_j,t)$, and $\rho(\lambda_i,\theta_j,t)$, respectively. Then Eqs. (2)–(4) can be written using the second-order spatial central difference scheme as follows:

$$\begin{aligned}L_{\rm lat}(\lambda_i,\theta_j,t)&=-\frac{1}{2}\left(\frac{u_{i+1,j}\rho^*_{i+1,j}-u_{i-1,j}\rho^*_{i-1,j}}{2r\Delta\lambda\cos\theta_j}+u_{ij}\frac{\rho^*_{i+1,j}-\rho^*_{i-1,j}}{2r\Delta\lambda\cos\theta_j}\right)\\&=\frac{1}{2r\Delta\lambda\cos\theta_j}(u_{i-1/2,j},-u_{i+1/2,j})\begin{pmatrix}\rho^*_{i-1,j}\\ \rho^*_{i+1,j}\end{pmatrix},\end{aligned} \quad (10)$$

$$\begin{aligned}L_{\rm lon}(\lambda_i,\theta_j,t)&=-\frac{1}{2}\left(\frac{\varepsilon_{+1}v_{i,j+1}\rho^*_{i,j+1}-\varepsilon_{-1}v_{i,j-1}\rho^*_{i,j-1}}{2r\Delta\theta}+v_{i,j}\frac{\rho^*_{i,j+1}-\rho^*_{i,j-1}}{2r\Delta\theta}\right)\\&=\frac{1}{2r\Delta\theta}(v_{i,j-1/2},-v_{i,j+1/2})\begin{pmatrix}\rho^*_{i,j-1}\\ \rho^*_{i,j+1}\end{pmatrix},\end{aligned} \quad (11)$$

$$\frac{\partial\rho^*_{ij}}{\partial t}=\frac{(u_{i-1/2,j},-u_{i+1/2,j})}{2r\Delta\lambda\cos\theta_j}\begin{pmatrix}\rho^*_{i-1,j}\\ \rho^*_{i+1,j}\end{pmatrix}+\frac{(v_{i,j-1/2},-v_{i,j+1/2})}{2r\Delta\theta}\begin{pmatrix}\rho^*_{i,j-1}\\ \rho^*_{i,j+1}\end{pmatrix}, \quad (12)$$

where $u_{i\pm(1/2),j}=(u_{i\pm1,j}\pm u_{i,j})/2$, $v_{i,j\pm(1/2)}=(\varepsilon_{\pm1}v_{i,j\pm1}\pm v_{i,j})/2$, and the ε_{+1} and ε_{-1} are two parameters that control the relative directions of $v_{i,j+1}$ and $v_{i,j-1}$, respectively. Because south wind is defined positive, when the variables on grids crossing the north(south)pole are used in Eq. (11), that is to say, if $j+1>J$ in Eq. (11), then the ε_{+1} *is* set to be $\varepsilon_{+1}=-1$, otherwise $\varepsilon_{+1}=1$. Similarly, when $j-1<-J$ then the ε_{-1} is set to be $\varepsilon_{-1}=-1$ otherwise $\varepsilon_{-1}=1$. From Eq. (12), the transport equation in discrete form on the sphere can now be written as a differential equation set:

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

$$\frac{\partial \boldsymbol{\rho}_G^*}{\partial t} = [U(t) + V(t)]\boldsymbol{\rho}_G^*, \tag{13}$$

where $\boldsymbol{\rho}_G^*$ defines a vector, which is set to be $\boldsymbol{\rho}_G^* = \{\rho_{ij}^* | 0 \le i \le 2I-1 \text{ and } -J \le j \le J\}$; $\mathbf{U}(t)$ and $\mathbf{V}(t)$ represent the transport-coefficient matrices, which are associated with u and v, respectively. With the definition of (9), we have the periodic boundary conditions $(u\rho^*)_{2I,j} = (u\rho^*)_{0,j}$, $j \in [-J, J]$ in λ direction, $(v\rho^*)_{i,\pm J\pm 1} = (v\rho^*)_{I+i,J}$, $i \in [0, I-1)$ and $(v\rho^*)_{i,\pm J\pm 1} = (v\rho^*)_{i-I,J}$, $i \in [\mathrm{I}, 2I]$ in θ direction. Multiplying Eq. (13) by ρ_G^{*T}, we can obtain

$$(\boldsymbol{\rho}_j^*)^{\mathrm{T}} U_j \boldsymbol{\rho}_j^* = \sum_{i=0}^{2I-1} \rho_{i,j}^* L_{\mathrm{lat}}(\lambda_i, \theta_j, t) = 0, \tag{14}$$

$$(\boldsymbol{\rho}_i^*)^{\mathrm{T}} V_i \boldsymbol{\rho}_i^* = \sum_{j=-J}^{J} \rho_{i,j}^* L_{\mathrm{lon}}(\lambda_i, \theta_j, t) + \sum_{j=-J}^{J} \rho_{I+i,j}^* L_{\mathrm{lon}}(\lambda_{I+i}, \theta_j, t) = 0, \tag{15}$$

where

$$\boldsymbol{\rho}_j^* = (\rho_{0,j}^*, \rho_{1,j}^* \rho_{2,j}^*, \ldots, \rho_{2I-2,j}^* \rho_{2I-1,j}^*)^{\mathrm{T}}, \quad j \in [-J, J] \tag{16}$$

is a vector composed by the discrete ρ^* on the latitude θ_j;

$$\boldsymbol{\rho}_i^* = (\rho_{i,-J}^*, \rho_{i,-J+1}^*, \ldots, \rho_{i,J}^*, \rho_{i+I,J}^*, \rho_{i+I,J-1}^*, \ldots, \rho_{i+I,-J}^*)^{\mathrm{T}}, \quad i \in [0, I) \tag{17}$$

is a vector composed by the discrete ρ^*on longitude λ_i and λ_{i+I};

$$\mathbf{U}_j = \frac{-1}{2r\Delta\lambda \cos\theta_j} \begin{bmatrix} 0 & u_{1/2,j} & 0 & 0 & \ldots & 0 & -u_{2I-1/2,j} \\ -u_{1/2,j} & 0 & u_{1+1/2,j} & 0 & \ldots & \ldots & 0 \\ 0 & -u_{1+1/2,j} & 0 & u_{2+1/2,j} & \ldots & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & 0 & -u_{2I-2+1/2,j} & 0 & u_{2I-3/2,j} \\ u_{2I-1/2,j} & 0 & 0 & \ldots & 0 & -u_{2I-3/2,j} & 0 \end{bmatrix} \tag{18}$$

and

$$\mathbf{V}_i = \frac{-1}{2r\Delta\theta} \begin{bmatrix} 0 & -v_{i,-J-1/2} & 0 & 0 & \ldots & 0 & v_{i,-J+1/2} \\ v_{i,-J-1/2} & 0 & -v_{i,-J-3/2} & 0 & \ldots & \ldots & 0 \\ 0 & v_{i,-J-3/2} & 0 & -v_{i,J-5/2} & \ldots & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & \ldots & 0 & v_{i+I,-J-3/2} & 0 & -v_{i+I,-J-1/2} \\ -v_{i,-J+1/2} & 0 & 0 & \ldots & 0 & v_{i+I,-J-1/2} & 0 \end{bmatrix} \tag{19}$$

are transport-coefficient matrices of $\boldsymbol{\rho}_j^*$ and $\boldsymbol{\rho}_i^*$, respectively, which compose the $\mathbf{U}(t)$ and $\mathbf{V}(t)$. From Eqs.(13)–(15), the corresponding relation can then be derived as

$$r^2 \Delta\theta\Delta\lambda \sum_{i=0}^{2I-1} \sum_{j=-J}^{J} \left(\frac{\partial (\rho_{i,j}^*)^2}{\partial t} \right) = \sum_{i=0}^{2I-1} \sum_{j=-J}^{J} \left(\frac{\partial \rho_{i,j}}{\partial t} \Delta s \right) = 0, \tag{20}$$

where $\Delta s = r^2 \cos\theta_j \Delta\theta\Delta\lambda$ is an area of grid cell on latitude θ_j. Therefore, we can draw a conclusion that Eq. (13) is global square conservative. From another point of view, it can be seen that those $\mathbf{U}_j$ and $\mathbf{V}_i$ are antisymmetrical matrices, according to the property of antisymmetric matrix, $\mathbf{U}(t_n) + \mathbf{V}(t_n)$ is also antisymmetric, that is to say we can have $(\boldsymbol{\rho}_G^*)^{\mathrm{T}}[\mathbf{U}(t_n) + \mathbf{V}(t_n)]\boldsymbol{\rho}_G^* = 0$, by which the square conservation property of the transport equation can be preserved in discretization form.

c. *Semianalytical solution of transport equation*

Let Δt be the temporal increment, $t_n = n\Delta t$, and $\boldsymbol{\rho}_G^*(t_n)$ is a vector at time t_n, where n is an integer. Then, the analytical solution of Eq. (13) is given by

$$\boldsymbol{\rho}_G^*(t_{n+1}) = \mathbf{e}^{\int_{t_n}^{t_{n+1}} [\mathbf{U}(t) + \mathbf{V}(t)]dt} \boldsymbol{\rho}_G^*(t_n). \tag{21}$$

Because $\mathbf{U}(t)$ and $\mathbf{V}(t)$ are dependent of time, the right-hand side of Eq. (21) is usually split into linear part and nonlinear part, the known methods are ELP method and exponentially fitted Euler method (see the introduction). In this work, we considered a second-order leapfrog approximation to the time integration of Eq. (21), as follows:

$$\int_{t_{n-1}}^{t_{n+1}} [\mathbf{U}(t) + \mathbf{V}(t)]\, dt \approx 2\Delta t[\mathbf{U}(t_n) + \mathbf{V}(t_n)], \tag{22}$$

where t_n is the midtime between t_{n-1} and t_{n+1}. Then the approximate solution to the Eq. (21) can now be written using Eq. (22) as follows:

$$\boldsymbol{\rho}_G^*(t_{n+1}) = \mathbf{e}^{2\Delta t[\mathbf{U}(t_n) + \mathbf{V}(t_n)]} \boldsymbol{\rho}_G^*(t_{n-1}). \tag{23}$$

The exponential matrix in Eq. (23) is referred as transport matrix. Comparing with the three time-levels leapfrog scheme (ELP) proposed by Beylkin et al. (1988), the nonlinear part is

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

omitted in Eq. (23) and the error is proportional to Δt^2, but in this way, the properties of square conservation and reversibility can be successfully preserved by Eq. (23) because Eq. (22) is an antisymmetric matrix. By multiplying the inverse matrix of the exponential matrix of Eq. (23), we can obtain

$$\boldsymbol{\rho}_G^*(t_{n-1}) = \mathbf{e}^{-2\Delta t[\mathbf{U}(t_n)+\mathbf{V}(t_n)]}\boldsymbol{\rho}_G^*(t_{n+1}), \tag{24}$$

which is the inverse function to Eq. (23), and is able to solve the inverse problem of $\boldsymbol{\rho}_G^*(t_{n+1})$. The difference of the two transport matrices between Eqs. (23) and (24) is the sign of time step. If the negative time step ($-\Delta t < 0$) is adopted, the transport governed by Eq. (24) is in the opposite time direction to that by Eq. (23).

3. Exponential integral methods

a. Adaptive Krylov subspace method

A powerful class of methods to solve exponential integration are the Krylov subspace methods, by which we can seek an approximation solution to Eqs. (23) and (24). Because Eq. (23) or (24) is one of the problems of $\mathbf{b}(t) = \mathbf{e}^{\mathbf{A}t}\mathbf{b}_0$ when $\mathbf{A}$ is a $m_\mathbf{A}$-by-$m_\mathbf{A}$ antisymmetric matrix, where $\mathbf{b}_0 \equiv \mathbf{b}(0)$ is an initial vector, we can take $\mathbf{b}(t) = \mathbf{e}^{\mathbf{A}t}\mathbf{b}_0$ as an example to illustrate Krylov subspace method. Krylov subspace methods are based on the idea of projecting the matrix $\mathbf{A}$ and vector $\mathbf{b}_0$ onto a lower-dimension subspace $\mathbf{K}_m$ defined by

$$\mathbf{K}_m = \text{span}\{\mathbf{b}_0, \mathbf{A}\mathbf{b}_0, \mathbf{A}^2\mathbf{b}_0, \cdots, \mathbf{A}^{m-1}\mathbf{b}_0\}, \tag{25}$$

where the parameter m is the dimension of $\mathbf{K}_m$, the criterion of $m \ll m_\mathbf{A}$ is required because calculating an exponential matrix in $\mathbf{K}_m$ subspace with small size is significantly less computationally expensive. However, the basis vectors of $\mathbf{K}_m$ defined as (25) are almost linearly dependent, so Arnoldi iteration is often used to transform (25) into an orthonormal Krylov subspace whose m orthonormal basis vectors compose an $m_\mathbf{A}$-by-m matrix $\mathbf{B}_m$, in which the projection of the action of $\mathbf{A}$ to this $\mathbf{K}_m$ is $\mathbf{H}_m = \mathbf{B}_m^{\mathrm{T}}\mathbf{A}\mathbf{B}_m$, and then the approximation to $\mathbf{e}^{\mathbf{A}t}\mathbf{b}_0$ is given by

$$\mathbf{e}^{\mathbf{A}t}\mathbf{b}_0 \approx \|\mathbf{b}_0\|\mathbf{B}_m\mathbf{e}^{\mathbf{H}_m t}\mathbf{e}_1, \tag{26}$$

where $\mathbf{H}_m$ is an upper m-by-m Heisenberg matrix, and $\mathrm{e}_1 = (1, 0, 0, \ldots, 0)^{\mathrm{T}}$ is the first vector of $\mathbf{B}_m^{\mathrm{T}}\mathbf{B}_m$.

As we all know, when matrix $\mathbf{A}$ is symmetric, Lanczos iteration is more effective then Arnoldi iteration for building $\mathbf{B}_m$ and $\mathbf{H}_m$ of $\mathbf{K}_m$ subspace (Niesen and Wright 2012). By some modification, the Lanczos iteration can be used to deal with antisymmetric matrices and build a $\mathbf{K}_m$ subspace. It related to the matrix $\mathbf{A}$ by the relation $\mathbf{H}_m = \mathbf{B}_m^{\mathrm{T}}\mathbf{A}\mathbf{B}_m$, if A is antisymmetric then $\mathbf{H}_m$ is both antisymmetric and Heisenberg, which means that it is an antisymmetric tridiagonal matrix, whose diagonal elements are zero. Then Lanczos iteration can now be modify to deal with the antisymmetric $\mathbf{A}$. The "Algorithm 1" in the appendix shows the modification of Lanczos iteration for antisymmetric matrix, where $\mathbf{B}_m(k)$, $k = 0, 1, ..., m$ denote the kth vector of $\mathbf{B}_m$.

Although the modified Lanczos iteration can build $\mathbf{B}_m$ and $\mathbf{H}_m$ when the $\mathbf{A}$ is antisymmetric, the key to this method is to be able to use an appropriate m, because it controls the accuracy of Eq. (26), moreover, the criterion of $m \ll m_\mathbf{A}$ is needed. Therefore, the best choice is searching an appropriate m adaptively under certain condition of error tolerance. The error estimate of Eq. (26) can be bounded by the inequality of Eq. (27) (Niesen and Wright 2012; Koskela 2015; Vo and Sidje 2017):

$$\|\mathbf{e}^{At}\mathbf{b}_0 - \|\mathbf{b}_0\|\mathbf{B}_m\mathbf{e}^{\mathbf{H}_m t}\mathbf{e}_1\| \approx \|\mathbf{b}_0\|\|\mathbf{H}_m(m, m+1)t\mathbf{e}_m^{\mathrm{T}}\varphi_1(t\mathbf{H}_m)\mathbf{e}_1|, \tag{27}$$

where $\varphi_1(t\mathbf{H}_m)$ is one of the φ-function family (Niesen and Wright 2012). Note that the $\mathbf{e}^{t\mathbf{H}^m}$ is an exponential matrix, which means that we have to calculate it in each Lanczos iteration step, and the dimension of $\mathbf{H}_m$, i.e., m, increases with the iteration step, so it is not a very economic method to use Eq. (27) for error estimation. Here, we provide a new adaptive algorithm for determining the m.

For the problem of $\mathbf{b}(t) = \mathbf{e}^{\mathbf{A}t}\mathbf{b}_0$, $\mathbf{b}_0 \equiv \mathbf{b}(0)$,we have the following properties:

$$\frac{d^k\mathbf{b}}{dt^k} = \mathbf{A}^k\mathbf{b}, \quad k = 1, 2, 3, \ldots, \tag{28}$$

$$\frac{d^{k+1}\mathbf{b}}{dt^{k+1}} = \mathbf{A}\frac{d^k\mathbf{b}}{dt^k}, \quad k = 1, 2, 3, \ldots. \tag{29}$$

So, the $\mathbf{K}_m$ given by Eq. (25) can be expressed alternatively as

$$\mathbf{K}_m = \text{span}\left\{\mathbf{b}(0), \frac{d\mathbf{b}(0)}{dt}, \frac{d^2\mathbf{b}(0)}{dt^2}, \frac{d^3\mathbf{b}(0)}{dt^3}, \cdots, \frac{d^{m-1}\mathbf{b}(0)}{dt^{m-1}}\right\}. \tag{30}$$

On the other hand, we can use a Taylor series of $m-1$ degree to approximate $\mathbf{b}(t)$, as follows:

$$\mathbf{b}(t) = \mathbf{b}(0) + t\frac{d\mathbf{b}(0)}{dt} + \frac{t^2}{2!}\frac{d^2\mathbf{b}(0)}{dt^2} + \cdots + \frac{t^{m-1}}{(m-1)!}\frac{d^{m-1}\mathbf{b}(0)}{dt^{m-1}}. \tag{31}$$

By comparing Eqs. (30) and (31), we can find that Eq. (31) is actually the projection of $\mathbf{b}(t)$ onto subspace $\mathbf{K}_m$ given by (30), and the $t^k/k!||d^k\mathbf{b}(0)/dt^k||$, where $k = 0, 1, \ldots, m$, is the size of projection of $\mathbf{b}(t)$ in the direction of vector $d^k\mathbf{b}(0)/dt^k$, and $t^m/m!||d^m\mathbf{b}(0)/dt^m||$ is often used to measure the accuracy of this Taylor series. However, the $\mathbf{K}_m$ subspace (30) is an incomplete orthogonal space, which means that Eq. (31) is unable to converge fast. Actually, the action of Lanczos iteration is to transform subspace (30) into a fully orthogonal and normalized $\mathbf{K}_m$ subspace, whose basic vector matrix is $\mathbf{B}_m$, in which the Taylor series of $\mathbf{b}(t)$ becomes more efficient and converge faster than that in the subspace (30). If we could build this Taylor series then we can determine the dimension of $\mathbf{B}_m$, i.e., the dimension of $\mathbf{K}_m$ subspace adaptively by means of the Lanczos iteration. According to the "algorithm 1" (lines 7–12),we can have the recursion equation, as follows:

$$\mathbf{B}_m(k+1) = \frac{\mathbf{A}\mathbf{B}_m(k) + \|\mathbf{B}_m^*(k)\|\mathbf{B}_m(k-1)}{\|\mathbf{B}_m^*(k+1)\|}, \quad 1 \le k \le m-1, \tag{32}$$

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

where $\mathbf{B}_m(0) = \mathbf{b}_0/\|\mathbf{B}_m^*(0)\|$ and $\mathbf{B}_m(1) = \mathbf{A}\mathbf{b}_0/\|\mathbf{B}_m^*(1)\|!$. Let $F(0) = 1$ and $F(1) = \mathbf{A}$, then Eq. (32) can be written alternatively as follows:

$$\begin{cases} F(k+1) = \mathbf{A}F(k) + \|\mathbf{B}_m^*(k)\|^2 F(k-1) \\ \mathbf{B}_m(k+1) = \dfrac{F(k+1)\mathbf{b}_0}{\|\mathbf{B}_m^*(k+1)\|!} \end{cases} ; \quad 1 \le k \le m-1. \tag{33}$$

Because the $\mathbf{B}_m$ is orthonormal, we can get a set of fully orthogonal basis vectors, as follows:

$$\{\mathbf{b}_0, F(1)\mathbf{b}_0, F(2)\mathbf{b}_0, \cdots, F(m-1)\mathbf{b}_0\}. \tag{34}$$

By comparing Eq. (34) with the Krylov subspace defined by (25) and (30), the vector $\|\mathbf{B}_m^*(k)\|!\mathbf{B}_m(k) = F(k)\mathbf{b}_0$ can be regarded as the term $d^k\mathbf{b}(0)/dt^k$ in the orthonormal Krylov subspace whose basic vector matrix is the $\mathbf{B}_m$, so that we can have a Taylor series of m degree, and then use Eq. (35) as an error estimate to determine the dimension of $\mathbf{K}_m$ subspace in Lanczos iteration:

$$\|\mathbf{e}^{\mathbf{A}t}\mathbf{b}_0 - \|\mathbf{b}_0\|\mathbf{B}_m\mathbf{e}^{\mathbf{H}_m t}\mathbf{e}_1\| \approx \frac{t^m}{m!}\|\mathbf{B}_m^*(m)\|! < \mathrm{Tol}, \tag{35}$$

where the Tol is the error tolerance, The advantage of this error estimate is that Eq. (35) can be directly derived out in Lanczos iterations without need of other extra information. Then the "algorithm 1" can now be improved to an adaptive Lanczos iteration for antisymmetric matrix. See "algorithm 2" in the appendix. The differences between these two algorithms are the "do-loop" in line 2 and the adaptive error estimate in line 7.

Despite we can obtain $\mathbf{H}_m$ and $\mathbf{B}_m$ adaptively, the other problem that we have to face is the expensive computational cost of $\mathbf{AB}_m(k)$ in the "algorithm 2" because it seems that $\mathbf{AB}_m(k)$ is a multiplication of the large matrix $\mathbf{A}$ with the vector $\mathbf{B}_m(k)$. However, for a specific problem, the $\mathbf{AB}_m(k)$ can be calculated by some formula rather than computing the product of matrix $\mathbf{A}$ and vector $\mathbf{B}_m(k)$. Taking the transport Eq. (23) as an example, let $\mathbf{A}$ and $\mathbf{b}_0$, respectively, represent $[\mathbf{U}(t_n) + \mathbf{V}(t_n)]$, and $\boldsymbol{\rho}_G^*(t_{n-1})$, and then $\mathbf{AB}_m(k)$, $k = 1, 2, ..., m$, is equivalent to Eq. (29) whose discrete equation is similar to Eq. (12), so $\mathbf{AB}_m(k)$ can be obtained formulaically rather than using matrix multiplication. It also means that we do not need to build a very large matrix $\mathbf{A}$, but only need to build a m-by-$2I$-by-$(2J + 1)$ array to store $\mathbf{B}_m$ and a m-by-m array to store $\mathbf{H}_m$.

b. Scaling and squaring method

After obtaining the matrix $\mathbf{H}_m$, then $\mathbf{e}^{\mathbf{H}_m t}$ may be efficiently calculated by the well-known scaling and squaring method with the Padé approximation $R_{qq}(\cdot)$, as follows (Moler and Van Loan 2003):

$$\mathbf{e}^{\mathbf{H}_m t/2^S} \approx R_{qq}(\mathbf{H}_m t/2^S), \tag{36}$$

$$\mathbf{e}^{\mathbf{H}_m t/2^{s-1}} = (\mathbf{e}^{\mathbf{H}_m t/2^s})^2, \quad s \in [1, S]. \tag{37}$$

For such methods, there are two key parameters that affect the accuracy of $\mathbf{e}^{\mathbf{H}_m t}$, one is the scaling parameter S, which is an integer and is automatically determined by the restriction $\|\mathbf{H}_m^t\|/2^S \le 1/2$, the other is the degree of Padé approximation q, which can be automatically determined by

$$8\left\|\frac{\mathbf{H}_m t}{2^S}\right\|^{2q+1}\frac{q!q!}{(2q)!(2q+1)!} < \mathrm{Tol}_q, \tag{38}$$

where Tol_q is the error tolerance of Eq. (36) and set to be $\mathrm{Tol}_q = 10^{-15}$. Eq. (37) is a recurrence formula by which we can obtain $\mathbf{e}^{\mathbf{H}_m t}$ by repeating matrix multiplication S times after obtaining $\mathbf{e}^{\mathbf{H}_m t/2^S}$ by Eq. (36).

Here, another problem that we have to mention is the "hump" problem associated with the repeated multiplication of matrices in scaling and squaring methods. It has been proved theoretically that the "hump" problem does not exist for normal matrix exponential (Moler and Van Loan 2003). In an orthonormal $\mathbf{K}_m$ subspace, if $\mathbf{A}$ is antisymmetric, that is $\mathbf{A}^{\mathrm{T}} = -\mathbf{A}$, then it is easy to verify that $\mathbf{H}_m^{\mathrm{T}}\mathbf{H}_m = \mathbf{H}_m\mathbf{H}_m^{\mathrm{T}}$ due to $\mathbf{H}_m = \mathbf{B}_m^{\mathrm{T}}\mathbf{A}\mathbf{B}_m$, thus $\mathbf{H}_m$ is a normal matrix. Note that the matrix $[\mathbf{U}(t_n) + \mathbf{V}(t_n)]$ given by (22) is an antisymmetric matrix as well as a normal matrix, so the scaling and squaring method is a quite feasible choice. However, matrix multiplication is computationally expensive when the m is relatively large, so a more effective method is needed to solve Eq. (37) for higher spatial resolution or 3D transport problems.

4. Central skip-point difference scheme for polar problem

In latitude–longitude coordinate systems, the meridians converge when approaching the pole and making the longitudinal grid interval approach zero at the pole where the coordinate system becomes singular. Therefore, relatively shorter waves can exist in high latitude regions, and the wavelength in λ direction becomes shorter and shorter as $|\theta|$ increase, especially near the polar points, which means that a sufficiently large $\mathbf{K}_m$ subspace is required to describe these short waves, and then amount of computer time is needed. In practice, those shorter waves, however, are not what we need for solving a specific problem. To deal with this problem, we attempt to extend Eq. (10) to a central skip-point difference scheme (Zhang et al. 2013) for the discretization in λ direction near the North and South Poles, as follows:

$$L_{\mathrm{lat}}(\lambda_i, \theta_j, t) = \frac{1}{2d_j}\left(\frac{u_{i-N_j,j} + u_{i,j}}{2}, -\frac{u_{i+N_j,j} + u_{i,j}}{2}\right)\begin{pmatrix}\rho^*_{i-N_j,j} \\ \rho^*_{i+N_j,j}\end{pmatrix}, \tag{39}$$

$$N_j = \begin{cases} \mathrm{int}\left(\dfrac{I}{2}\dfrac{\cos\theta_J}{\cos\theta_j}\right) & \mathrm{if}(\cos\theta_j \le \beta) \\ 1 & \mathrm{if}(\cos\theta_j > \beta) \end{cases}, \tag{40}$$

where the function int$(\cdot)$ returns the integer part of its input argument, N_j is the number of those grids skipped toward the west or east along the latitude θ_j, and $d_j = N_j r\Delta\lambda\cos\theta_j$ is the longitudinal distance between (λ_i, θ_j) and $(\lambda_{i+N_j}, \theta_j)$, which is N_j times the longitudinal distance $(r\Delta\lambda\cos\theta_j)$ used in Eq. (10). The N_j is dependent of j, where $j \in [-J, J]$. When $N_j = 1$, Eq. (39) is actually the Eq. (10), while when $N_j > 1$, Eq. (39) is

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

employed to discretize the zonal transport terms of transport equations in the polar cap regions. Here, the polar cap regions are defined as the regions whose latitude satisfy $\cos\theta_j \le \beta$, where β is a very important critical value, which will be addressed in section 6. It is important to note that N_J is $I/2$, and the ratio of d_J for latitude θ_J to $d_0 = r\Delta\lambda$ for equator is about $\pi/4$, so Eq. (39) is reasonable and feasible to be applied to the discretization for the polar problems associated with longitudinal grid distance. In addition, it can be easily proved that Eq. (39) satisfy square conservation for ρ_j^*. and we do not need to consider the problems of transport at the North and South Poles, because both of the two singularities are not defined in the spherical mesh.

5. Least squares filter method for polar problem

Equation (2) can be written alternatively as follows:

$$\frac{\partial \rho^{1/2}}{\partial t} = \frac{1}{2} C_{\mathrm{ur}} \rho^{1/2} - \frac{1}{2r}\left(\frac{\partial v\rho^{1/2}}{\partial \theta} + v\frac{\partial \rho^{1/2}}{\partial \theta} + \frac{\partial u\rho^{1/2}}{\cos\theta \partial \lambda} + u\frac{\partial \rho^{1/2}}{\cos\theta \partial \lambda}\right), \tag{41}$$

where $C_{\mathrm{ur}} = v/rtg\theta$. Eq. (41) is equivalent to Eq. (1) and Eq. (2) though they have different form. Because the area of a grid cell is $\Delta s = r^2 \cos\theta \Delta\theta \Delta\lambda$, we can get $C_{\mathrm{ur}} = -(d\Delta s dt^{-1})/\Delta s$. Therefore, the C_{ur} is actually the relative change rate of Δs, which means that ρ increase when transporting to the higher latitude region where the Δs is relatively smaller, while ρ decrease when transporting to the lower latitude region. The term $(1/2)C_{\mathrm{ur}}\rho^{1/2}$ can be negligible in low and middle latitude regions due to the relatively smaller $|tg\theta|$, but in high latitude regions, $|tg\theta|$ becomes more and more important because $|tg\theta|$ increase rapidly as the $|\theta|$ increase. According to the homogeneous solution of Eq. (41), $\rho^{1/2} = \exp\left[\int (1/2)C_{\mathrm{ur}}\, dt\right]\rho_{t=0}^{1/2}$, $C_{\mathrm{ur}} > 0$ is an unstable condition, namely, the numerical solution of $\rho^{1/2}$ is very sensitive to the error of $\rho^{1/2}$ when going through the polar regions due to the large $|tg\theta|$ there. To suppress the error growth caused by the large $|tg\theta|$, we design a least squares filter method (LSFM), whose filter function is given by Eq. (42), where the coordinate origin is set to be at the north (south) pole, $r\Delta\theta$ is normalized to be unit 1, and N_p is a parameter that controls the size of area to be filtered:

$$\begin{cases} \tilde{\rho}_{i,j} = a_0 + a_1 x_{ij} + a_2 y_{ij} + a_3 x_{ij}^2 + a_4 y_{ij}^2 + a_5 x_i y_j \\ x_{ij} = (J - |j| + 0.5)\cos\lambda_i \\ y_{ij} = (J - |j| + 0.5)\sin\lambda_i \end{cases}, \quad \text{for} \quad \begin{matrix} J - N_p \le |j| \le J \\ 0 \le i \le 2I - 1 \end{matrix}. \tag{42}$$

Taking the implementation of filter for the north polar region as an example to explain the LSFM, Assume that N_p is known, the unknown coefficients a_0, a_1, a_2, a_3, and a_4 in Eq. (42) can be obtained by minimizing Eq. (43), then we can obtain the solution of Eq. (42). The constraint condition for preserving mass conservation is Eq. (44), by which the correction coefficient μ is able to be determined. Note that the indices j is from $J - N_p + 1$ to J in Eq. (44), while j is from $J - N_p$ to J in Eq. (43), which means that the those $\rho_{i,j}^{1/2}$, whose j is from $J - Np + 1$ to J, are required to be corrected, and $\rho_{i,j-N_p}^{1/2}$ is used as a transitional area between filtered area and unfiltered area. The implementation of LSFM is given by Eq. (45):

$$J(a_1 \cdots a_5) = \sum_{j=J-Np}^{J} \sum_{i=0}^{2I-1} \frac{1}{2}\left[\rho_{i,j}^{1/2} - \tilde{\rho}_{i,j}\right]^2, \tag{43}$$

$$\mu^2 \sum_{j=J-Np+1}^{J} \sum_{i=0}^{2I-1} \left[\tilde{\rho}_{ij}^2 \cos\theta_j\right] - \sum_{j=J-Np+1}^{J} \sum_{i=0}^{2I-1} (\rho_{i,j}^*)^2 = 0, \tag{44}$$

$$\rho_{i,j}^{1/2} = \mu\tilde{\rho}_{i,j}, \quad J - N_p + 1 \le j \le J, \quad 0 \le i \le 2I - 1. \tag{45}$$

Obviously, N_P is an important parameter for LSFM method, it is associated with the coefficient C_{ur}, which can cause rapid error growth. The criterion for choosing N_p is given by

$$\begin{cases} \min\left\{E(N_p) = \sum_{i=0}^{2I-1} \left(\rho_{i,J-Np-1}^{1/2} - \tilde{\rho}_{i,J-Np-1}\right)^2, N_p \in [1, \mathrm{Max}N_p]\right\} \\ C_{\mathrm{ur}} = \dfrac{\max(|v|)tg[\Delta\theta(J - \mathrm{Max}N_p)]}{r} = 0.000\,18 \end{cases}, \tag{46}$$

where the parameter $\mathrm{Max}N_p$ can be determined by solving the equation $C_{\mathrm{ur}} = 0.000\,18$, which is obtained by a number of numerical tests. That is to say, the errors of ρ often increase noticeably in the region of $|C_{\mathrm{ur}}| > 0.000\,18$. However, how to determine a more accurate boundary of the region to be filtered needs more specific method. Eq. (46) is an enumeration algorithm, if N_p, respectively, is set to be 1 to $\mathrm{Max}N_p$, then we can obtain $\{E(1), E(2), \ldots, E(\mathrm{Max}N_p)\}$. Notice that the $\rho_{i,j-N_p-1}^{1/2}$ in Eq. (46) is the neighboring boundary of Eq. (43), so the idea of Eq. (46) is that minimizing the coupling error between the filtered area and the unfiltered area. That is to say, the best N_p is the one that make $E(N_p)$ smallest.

The primary subject of this paper is the SCEIM, this Least squares Filter Method is only used to improve the result of SCEIM by suppressing the error growth associated with polar problem and we will further investigate it later date.

6. Numerical test

The purpose of the numerical test is to evaluate the SCEIM method for solving transport equation. The forward model is constructed on the base of Eq. (23). For the backward mode, the time step is set to be a negative value. Since SCEIM is positive-definite, there is no need for any positive-definite correction techniques. The classical cosine bell tests and deformation flow tests are utilized to evaluate SCEIM. In these tests, the scale factor for the tangential velocity is set to be $v_0 = 2\pi/T$ as that in Nair and Jablonowski (2008), where T is total time for simulation and set to be $T \equiv 12$ days. The normalized

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

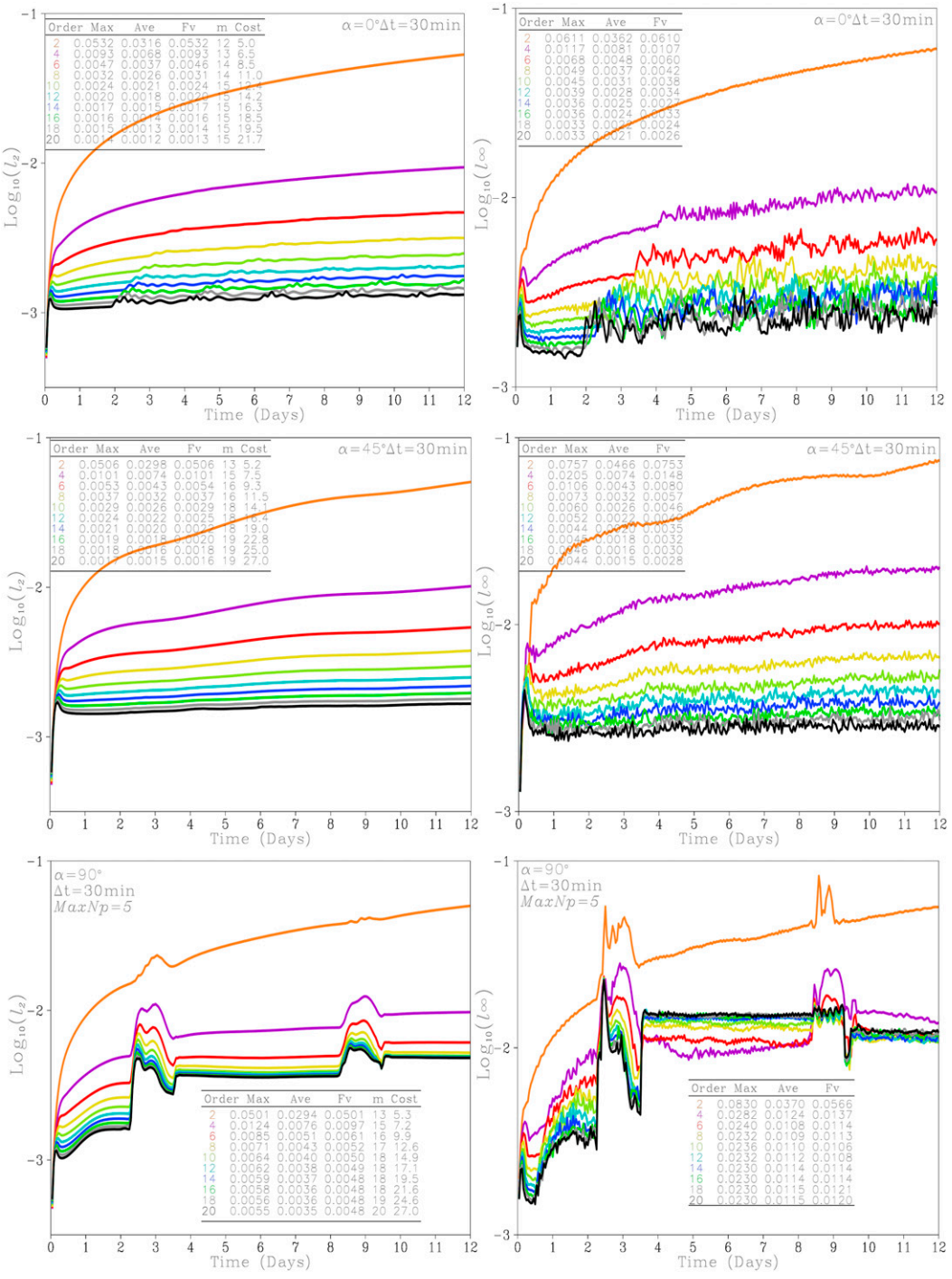

FIG. 2. The time evolution of error norms of cosine bell tests in the case of various-order spatial central difference schemes when the spatial resolution is 0.5°. The inset tables show the maximum value (Max), average value (Ave), final value (Fv), average dimension of $\mathbf{K}_m$ subspace (m), and the computational cost (Cost) with unit of minutes.

l_1, l_2, and l_∞ error norms of the transported scalar for each test are defined in Williamson et al. (1992). The numerical model is compiled in FORTRAN language and run serially on an IBM computer with an AMD Opteron (TM) processor 6212.

a. *Cosine bell test*

Advection of a cosine bell is widely used to evaluate numerical scheme considered for global modeling. The cosine bell is initially specified by Eq. (47), its center is set to be

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

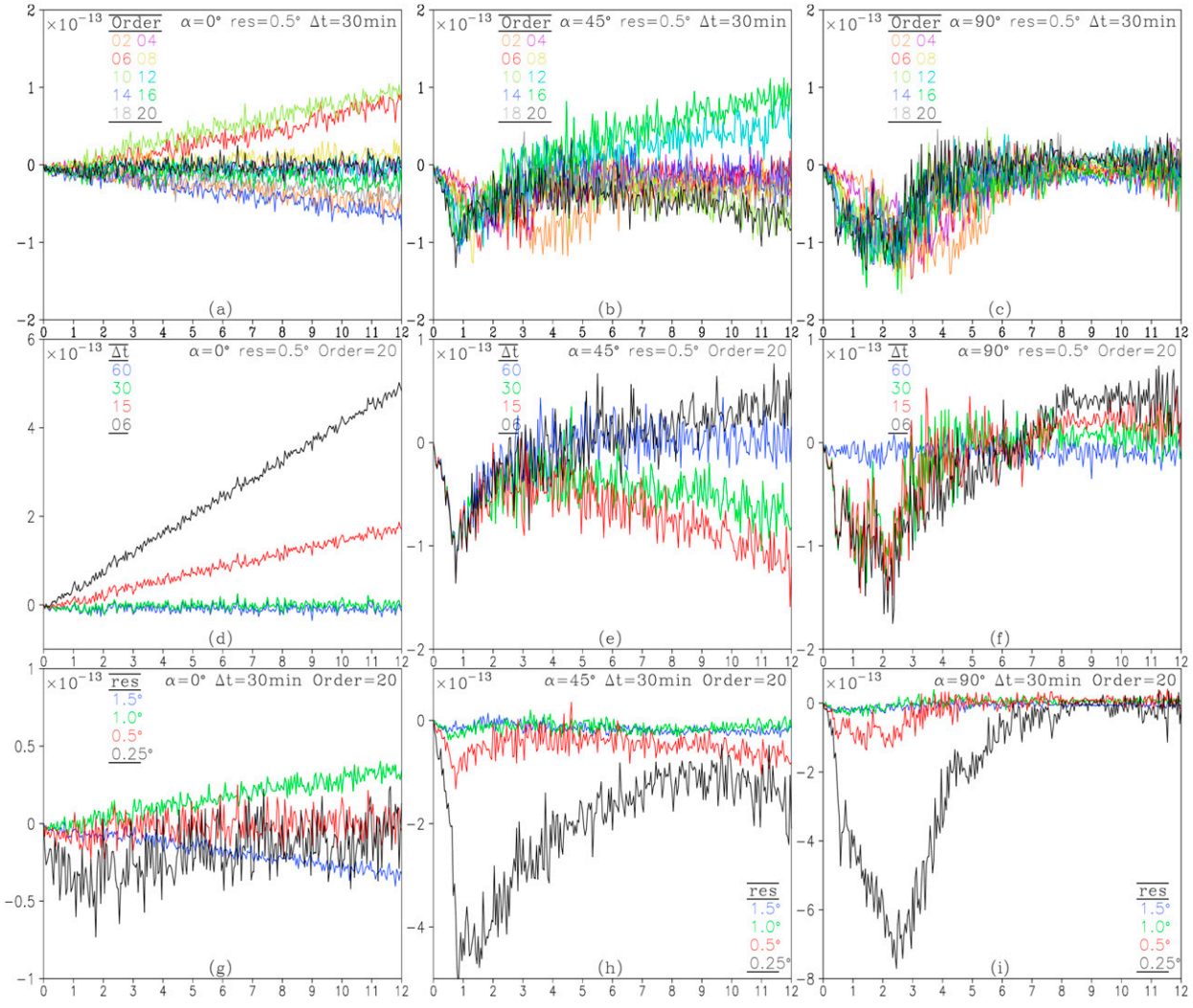

FIG. 3. The time evolution of relative errors of total mass of tests in the case of (a)–(c) 10 types of spatial central difference schemes, (d)–(f) 4 types of time step, and (g)–(i) 4 types of spatial resolution. The total mass are 1.52×10^5, 3.41×10^5, 1.36×10^6, and 5.43×10^6 in the case of the spatial resolution of 1.5°, 1.0°, 0.5°, and 0.25°, respectively.

$(\lambda_c, \theta_c) = (\pi/2, 0)$ and the ζ is the distance between (λ_c, θ_c) and (λ, θ) on the sphere. The driving wind field is given by Eq. (48), in which α is a parameter for controlling flow orientation:

$$\rho(\lambda,\theta) = \begin{cases} 500\left[1+\cos\left(\dfrac{3\pi\zeta}{r}\right)\right], & \zeta < r/3 \\ 0, & \zeta \geq r/3 \end{cases}, \tag{47}$$

$$\begin{cases} u_c = v_0 r(\cos\theta\cos\alpha + \sin\theta\cos\lambda\sin\alpha) \\ v_c = -v_0 r\sin\lambda\sin\alpha \end{cases}. \tag{48}$$

1) TEST FOR HIGH-ORDER SPATIAL CENTRAL DIFFERENCE SCHEME

For solving linear problems by means of exponential integration method, the numerical errors are primarily contributed by the spatial discretization, so it is necessary to investigate the influence of high-order spatial discretization schemes on the accuracy and efficiency of the SCEIM. In this test, we choose $\Delta t = 60$ min, Max$N_p = 5$, $\beta = 0.25$, and spatial resolution of 0.5°. The orders of 10 types of spatial central difference schemes are from second order to twentieth order. The Fig. 2 shows the evolution of l_2 and l_∞ of cosine bell tests with these 10 types of spatial central difference schemes. It can be seen that l_2 and l_∞ decrease as the order increase, the accuracy became better and better and finally converge to the error norms of twentieth-order spatial central difference. Both l_2 and l_∞ of the second-order scheme are the worst. Compared with the other schemes, the twentieth-order scheme is obviously the best. Compared to the test results with fourth-order scheme in the case of $\alpha = 0°$ and 45°, the average l_2 and l_∞ with twentieth-order scheme are improved by 5.7 times, 4.9 times, 4.0 times, and 4.9 times, respectively. For the test with $\alpha = 90°$, though l_∞ of twentieth-order scheme is equivalent to that of fourth-order

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

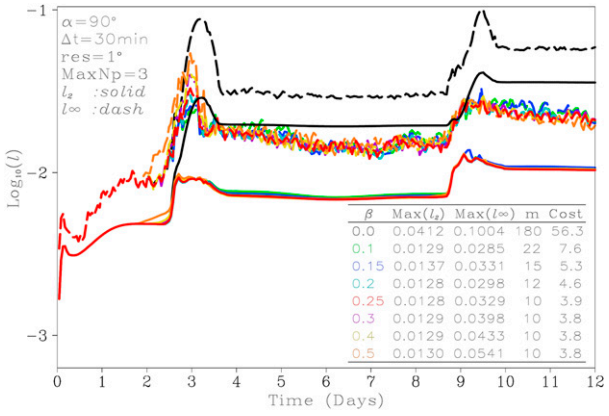

FIG. 4. The time evolution of error norms with various values of β in the case of spatial resolution (res) of 1°. The column names of the inset table are as in Fig. 2.

scheme, the l_2 of twentieth-order scheme is 1/2 times that of fourth-order scheme, which means that higher-order difference schemes are helpful to improve polar problems. From the change of the dimension of $\mathbf{K}_m$ subspace and the computational cost, which are denoted by "m" and "Cost" in the inner table, respectively, it can be seen that the average dimension of $\mathbf{K}_m$ subspace with twentieth-order difference scheme is only $m = 20$. Although the computational cost increased from around 5 to 27 min, we think such level of computational cost is acceptable, so we use the twentieth-order spatial central difference scheme in all of the following tests.

Another property to be evaluated is the total mass conservation of transported scalar based on SCEIM method. Figures 3a–c show the time evolution of relative error of total mass of cosine bell with second-order to twentieth-order spatial central difference schemes. The relative errors are not exceeding 2×10^{-13} for all of the tests. According to the definition of ρ^*, the density ρ is calculated by $\rho = (\rho^*)^2/\cos\theta$, so the numerical solution of ρ based on SCEIM is not only conservative but also everywhere positive. However, it seems that relative errors of total mass error are not improved obviously by the high-order of spatial central difference scheme.

2) TEST FOR CENTRAL SKIP-POINT DIFFERENCE SCHEME

As mentioned in section 3, if $\cos\theta_j < \beta$ then we use the central skip-point difference scheme instead of the central

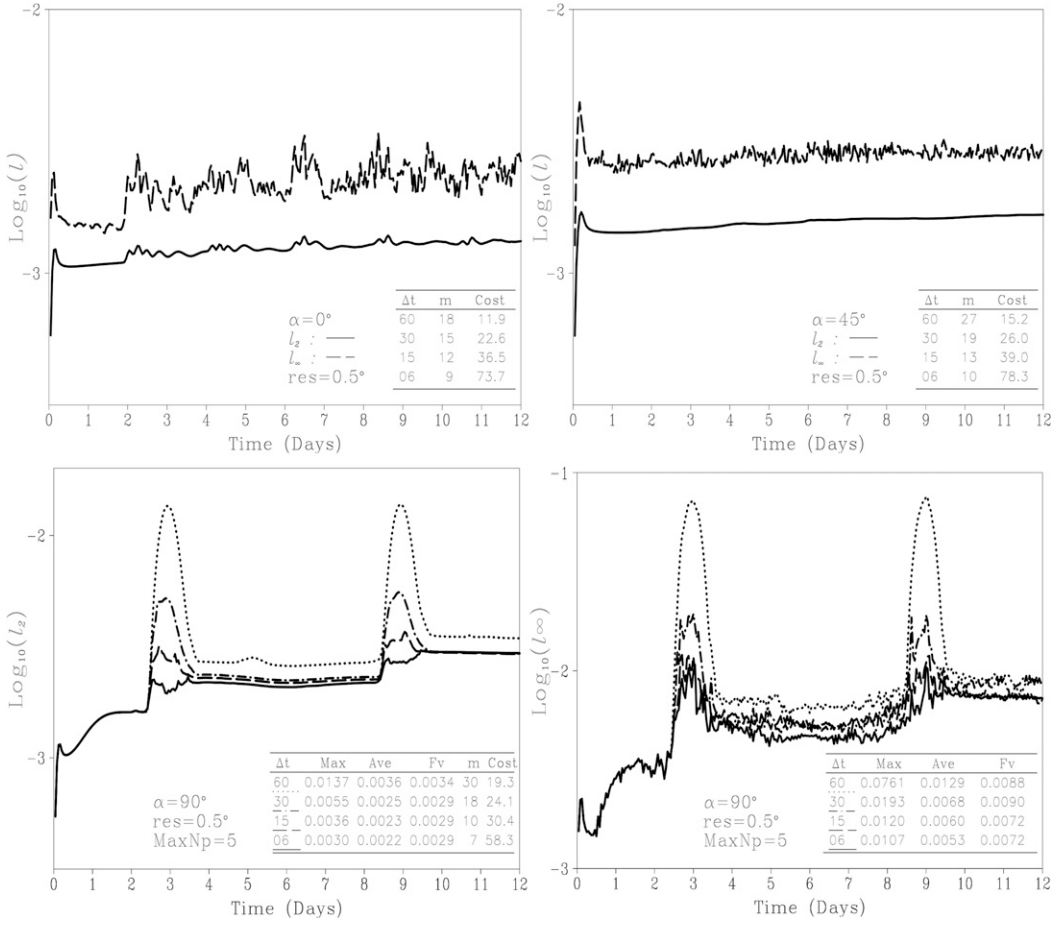

FIG. 5. The time evolution of error norms in the case of four types of time steps. The spatial resolution (res) is held 0.5° and the column names of the inset tables are as in Fig. 1.

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

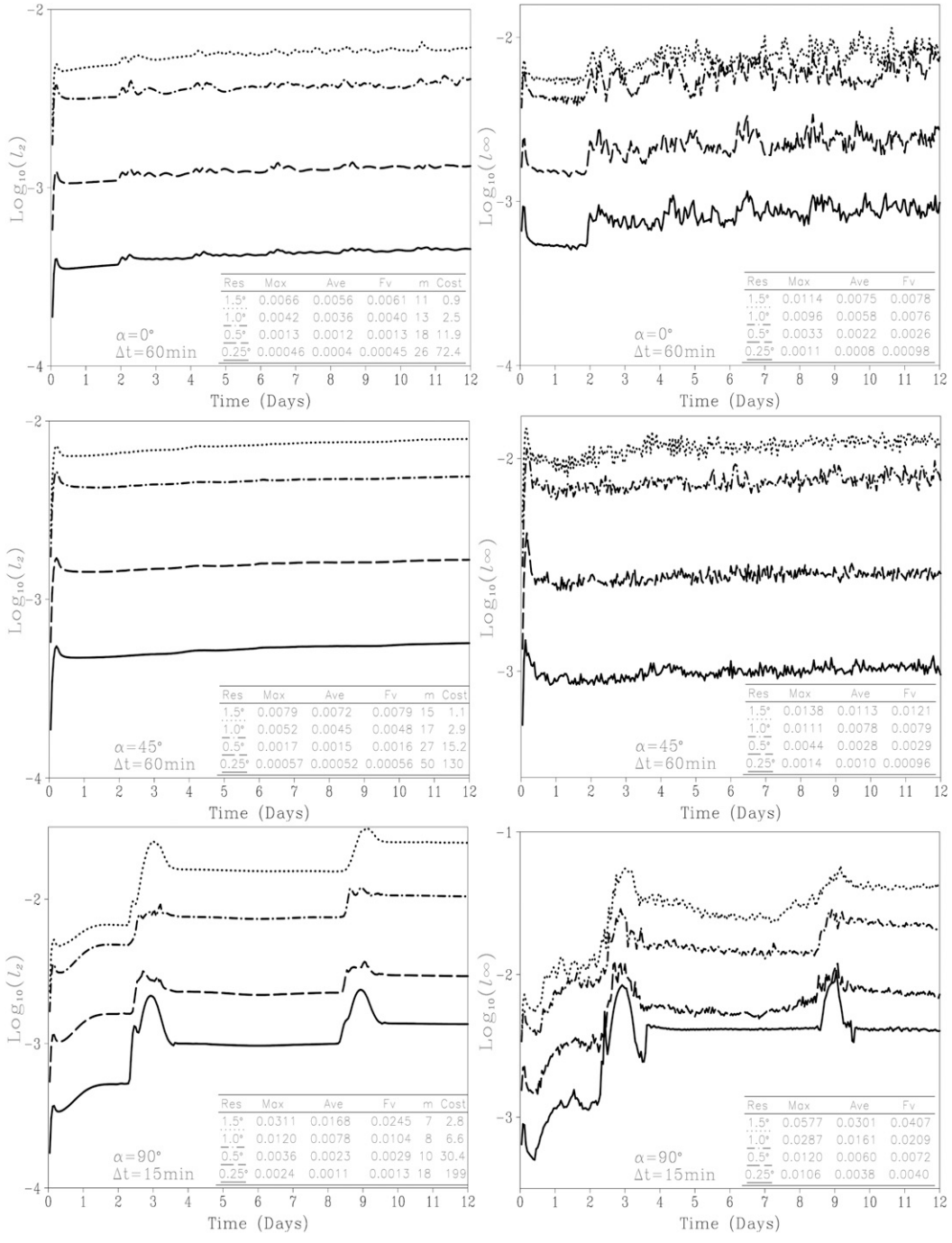

FIG. 6. The time evolution of error norms in the case of four types of spatial resolutions (res). The column names of the inset table are as in Fig. 1.

difference scheme, so the β is a very important parameter. In this viewpoint, we use eight values of β to evaluate the effectiveness and efficiency of central skip-point difference scheme for polar problems. The eight values of β, respectively, are 0.0, 0.1, 0.15, 0.2, 0.25, 0.3, 0.4, and 0.5, in which the $\beta = 0.0$ represents the spatial central difference scheme. To compare the central skip-point difference scheme and central difference scheme under a same condition, we utilize a test with spatial resolution of 1.0° and $\Delta t = 30$ min to evaluate these two types of scheme. The longitudinal distance along latitude θ_J is $d_J = 0.965$ km when $\beta = 0$, while $d_J = 0.965 \times 90 = 86.85$ km when $\beta \neq 0.0$. Figure 4 shows that the l_2 and l_∞ are the worst when $\beta = 0.0$, moreover, the value of m is up to 180, and the computational cost is more than 56.3 min, which is too expensive to

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

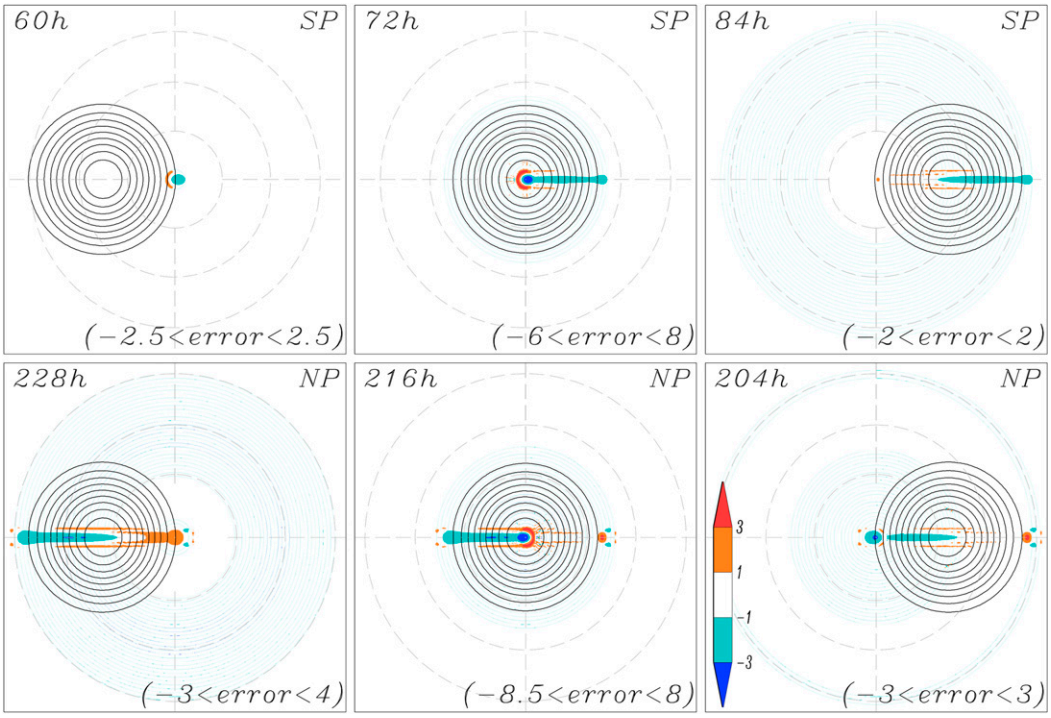

FIG. 7. The errors (shaded) and numerical solutions (contours) for cosine bell passing through the South Pole (SP) and the North Pole (NP) in the case of spatial resolution of 0.25°. The contour intervals are 100 from 100 to 900.

tolerate for this low spatial resolution. In contrast, for other values of β, the error norms are similar but the computational cost and the m decrease as β increase. We found that $\beta = 0.25$ is an appropriate value because the computational cost is not cheaper when $\beta > 0.25$. Moreover, it can be seen from the change of dimension of K_m subspace, the m decrease as β increase from 0.0 to 0.25, but keeps 10 as β increase from 0.3 to 0.5. For the latitude circle where β is 0.25, the longitudinal distance is 1/4 times that of the equator, and only three grid points are skipped in the λ direction. So, we think $\beta = 0.25$ is an appropriate choice, and it was used in all of the following tests.

3) TEST FOR TIME CONVERGENCE

we performed the time-convergence tests, in which the spatial resolution is held 0.5° × 0.5° and the forward model is integrated with 4 types of time step, i.e., 6, 15, 30, and 60 min. The corresponding Courant numbers, respectively, are 0.3125, 0.625, 1.25, and 2.5. For cosine bell test, which is a linear transport problem, Eq. (23) is an analytical solution whose error is primarily caused by the spatial discretization and independent of time. Figure 5 shows that the l_2 and l_∞ with $\alpha = 0°$ and $\alpha = 45°$ are not affected by the change of Δt. However, for the tests with $\alpha = 90°$, the results are very different, the l_2 and l_∞ decrease noticeably as the Δt decrease, and converge toward the l_2 and l_∞ with $\Delta t = 6$ min. This convergence is caused by the action of LSFM method, because the operation of LSFM is implemented after each time integration step, that is to say, if $\Delta t = 6$ min then the filter operation is implemented 10 times within a 1-h interval, while the filter operation is only implemented once when $\Delta t = 60$ min. It also demonstrates that action of LSFM is positive and effective and it is very important for improving the polar problem. For the tests with $\alpha = 90°$,the most noticeable phenomenon is the two peaks of the evolution of l_2 and l_∞ passing through the North Pole and South Pole, and the peaks also become better and converge as the Δt decrease. The change of m and the computational cost show that large time-step Δt can save amount of computation time despite it require a relatively larger m. However, considering the improvement of polar problem associated with the filter frequency, $\Delta t = 15$ min is an appropriate choice. Figures 3d and 3e show that for any type of time step from 6 to 60 min, the relative error of total mass is always within the 10^{-13} order of magnitude.

4) TEST FOR GRID CONVERGENCE

The grid-convergence tests were carried out by using 4 types of spatial resolution, i.e., 1.5°, 1°, 0.5°, and 0.25°, in which $\Delta t =$ 60 min is used for the tests with $\alpha = 0°$ and $\alpha = 45°$, while $\Delta t =$ 15 min is used for the test with $\alpha = 90°$ due to polar problems. When $\Delta t = 60$ min and $\Delta t = 15$ min are employed in the case of spatial resolution of 0.25°, the Courant numbers are 5.0 and 1.25, respectively. From Fig. 6 it can be seen that both l_2 and l_∞ decrease obviously as the spatial resolution increase. When $\alpha = 0°$ and $\alpha = 45°$, the average l_2 and l_∞ with resolution of 0.25° is only about 1/3 of those with resolution of 0.5°, even for $\alpha = 90°$, both l_2 and l_∞ are about 1/2 of those with resolution of 0.5°. The tables in Fig. 6 show the changes of the m and

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

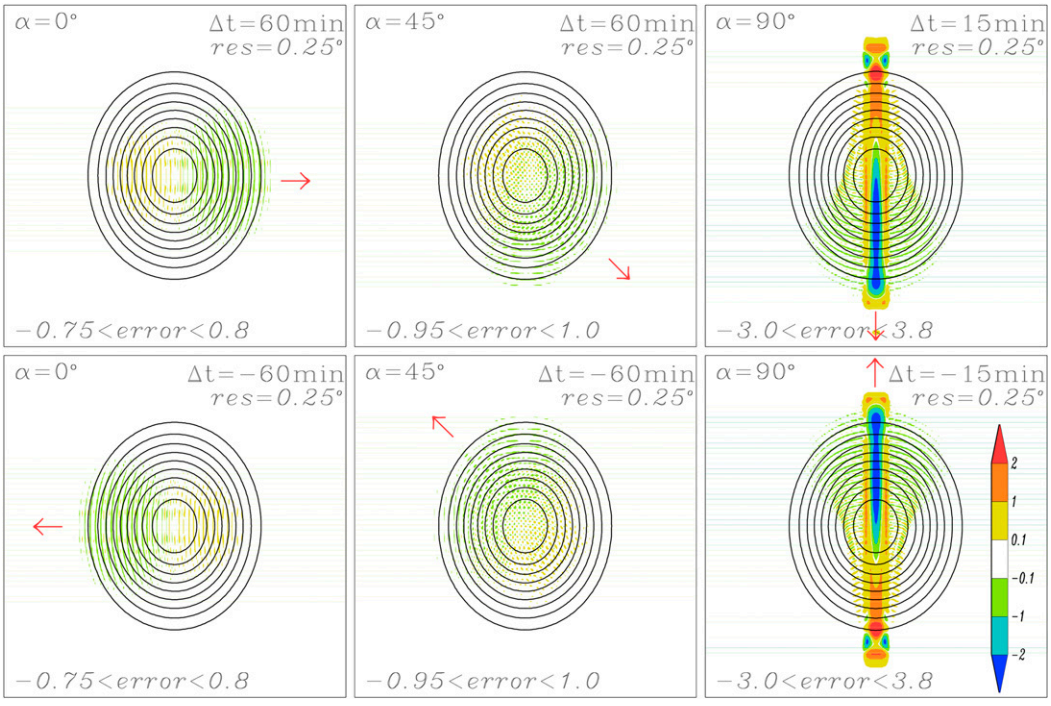

FIG. 8. The errors (shaded) and numerical solutions (contours) of forward and backward simulations of cosine bell tests with resolution of 0.25° after one complete rotation in three directions. The red arrows represent the moving direction of bells. The contour intervals are 100 from 100 to 900.

computational cost, the m increase as the spatial resolution increase, though m is only 18 in the test with $\alpha = 90°$ due to $\Delta t =$ 15 min, it took our 199 min to finish this test with spatial resolution of 0.25°. Figures 3g–i show the time evolution of relative error of total mass, for any type of spatial resolution from 1.5° to 0.25°, the relative error is always within the 10^{-13} order of magnitude.

To demonstrate the effectiveness of the combination of SCEIM, central skip-point difference scheme and LSFM for dealing with polar problem, Fig. 7 shows the error and contour lines of cosine bell, whose spatial resolution is 0.25° and the longitudinal grid distance is only 9.65 m near pole points. The top pictures show the cosine bells at 60, 72, and 80 h when going through south polar, and the bottom pictures show the bells at 204, 216, and 228 h when going through north polar. It can be seen that each contour lines presents better shape of circle in vision, even when its center is at the North Pole and South Pole. The negative errors are observed near the poles, and the positive errors follow the negative errors in the moving direction of the cosine bell. Although the error norms, especially l_∞, grow rapidly in polar regions, the shape of the bell is not distorted obviously, and the error norms decrease quickly after its center passed through the South and North Poles (Fig. 6).

5) TEST FOR INVERSE PROBLEM OF COSINE BELL

The most significant advantage of SCEIM is that it can change the forward model to the backward model by setting negative time step. This is an advantage that other known schemes do not have. For comparison, Fig. 8 shows the errors and numerical solutions of forward and backward simulations of cosine bell tests with resolution of 0.25° after one complete rotation in three directions. In the tests of inverse problem, the cosine bells rotate with reversed wind and return to their initial position after −12 days. It can be seen that the simulated contour lines depicted in the Fig. 8 are very smooth and similar; the error distribution is related to the moving direction of the cosine bell, which mainly occurs in the front of cosine bells. The errors of tests with $\alpha = 90°$ are quite different and larger compared with the two others due to the affection of polar problem, but the maximum error is only 3.8 for both forward and backward simulations. The error norms and other information of backward simulations are given in Fig. 9, which are identity to those with 0.25° resolution given in Fig. 5. Based on the above results, we think the backward model based on SCEIM method can solve the inverse problems of cosine bell tests.

6) COMPARISON OF SCEIM WITH OTHER PUBLISHED SCHEMES

We compared these results with those in Baba et al. 2010, (see their Table 1 and Table 2), and those in Li et al. 2016 (see their Fig. 4). The error norms of SCEIM are significantly better (Table 1). Note that Yin–Yang grid system was used to deal with the polar problem in both Baba's scheme and Li's schemes, however, the SCEIM exhibits competitiveness for the tests with $\alpha = 90°$. We also compared the results with those based on SLICE method proposed by Zerroukat et al. (2004), which had been compared with the published results from 7 schemes, though the spatial resolution of 2.8125° was used in

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

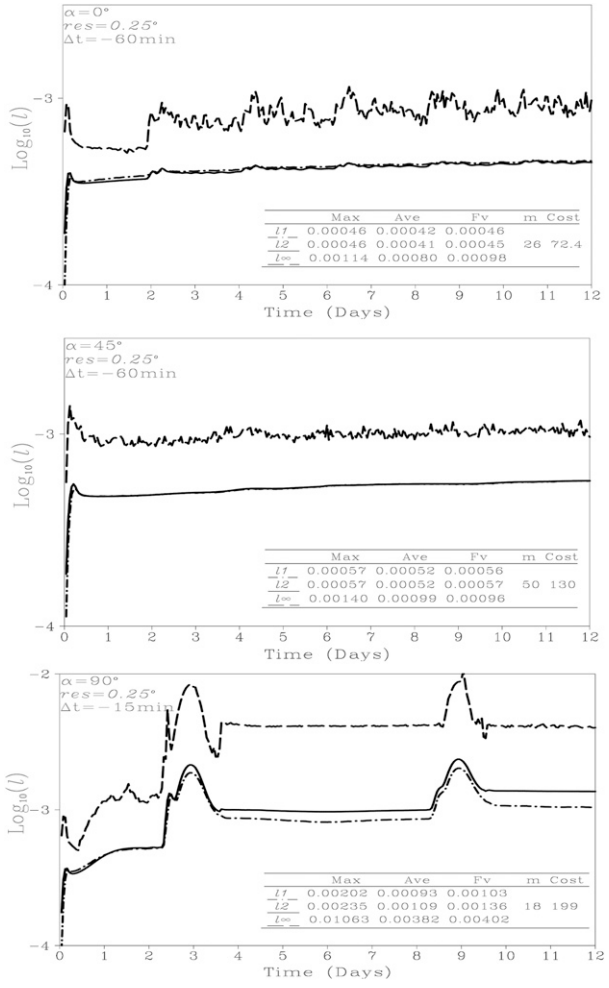

FIG. 9. The time evolution of error norms for backward simulation of cosine bell tests.

their SLICE, the l_1 and l_2 of SCEIM with spatial resolution of 1° are less than 1/7 of those of SLICE. For the backward simulations corresponding to the inverse problems of the cosine bells, though we have not found any published result in the literature for comparison, we achieved the quite identical accuracy to that of forward simulations.

b. *Deformational flow test*

The deformational flow test is also referred to as the frontogenesis test. The forward model is to simulate the evolution of frontogenesis, in which a weak front developed into a sharp front under the force action of vortex flow, whereas the backward model is to solve the inverse problem of frontogenesis, namely, how the sharp front is regressed back into the weak front. The analytical solution for ρ_T is given by Eq. (49), and the velocity components for deformational flow (u, v) is given by Eqs. (50) and (51) as those in Nair and Jablonowski (2008):

$$\rho_T(\lambda', \theta', t) = 1 - \tanh[0.2\rho \sin(\lambda' - wt)], \qquad (49)$$

TABLE 1. Comparison of error norms from three schemes for cosine bell tests after one complete rotation.

		$\alpha = 0°$			$\alpha = 45°$			$\alpha = 90°$		
		l_1	l_2	l_∞	l_1	l_2	l_∞	l_1	l_2	l_∞
1°	SCEIM	4.09×10^{-3}	4.06×10^{-3}	7.57×10^{-3}	4.95×10^{-3}	4.88×10^{-3}	7.91×10^{-3}	9.32×10^{-3}	1.04×10^{-2}	2.09×10^{-2}
	Baba	5.02×10^{-2}	3.42×10^{-2}	2.86×10^{-2}	5.33×10^{-2}	3.25×10^{-2}	2.29×10^{-2}	5.50×10^{-2}	3.59×10^{-2}	2.91×10^{-2}
	Li	1.63×10^{-2}	1.17×10^{-2}	1.19×10^{-2}	1.68×10^{-2}	1.05×10^{-2}	7.5–3	1.92×10^{-2}	1.31×10^{-2}	1.24×10^{-2}
0.5°	SCEIM	1.34×10^{-3}	1.32×10^{-3}	2.65×10^{-3}	1.66×10^{-3}	1.66×10^{-3}	2.87×10^{-3}	2.54×10^{-3}	2.93×10^{-3}	7.25×10^{-3}
	Baba	9.08×10^{-3}	7.31×10^{-3}	7.83×10^{-3}	9.98×10^{-3}	7.12×10^{-3}	6.64×10^{-3}	1.01×10^{-3}	7.71×10^{-3}	7.93×10^{-3}
0.25°	SCEIM	4.61×10^{-4}	4.52×10^{-4}	9.82×10^{-4}	5.69×10^{-4}	5.70×10^{-4}	9.64×10^{-4}	1.03×10^{-3}	1.36×10^{-3}	4.02×10^{-3}
	Baba	1.67×10^{-3}	1.77×10^{-3}	2.83×10^{-3}	1.87×10^{-3}	1.75×10^{-3}	1.96×10^{-3}	1.86×10^{-3}	1.86×10^{-3}	2.39×10^{-3}

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

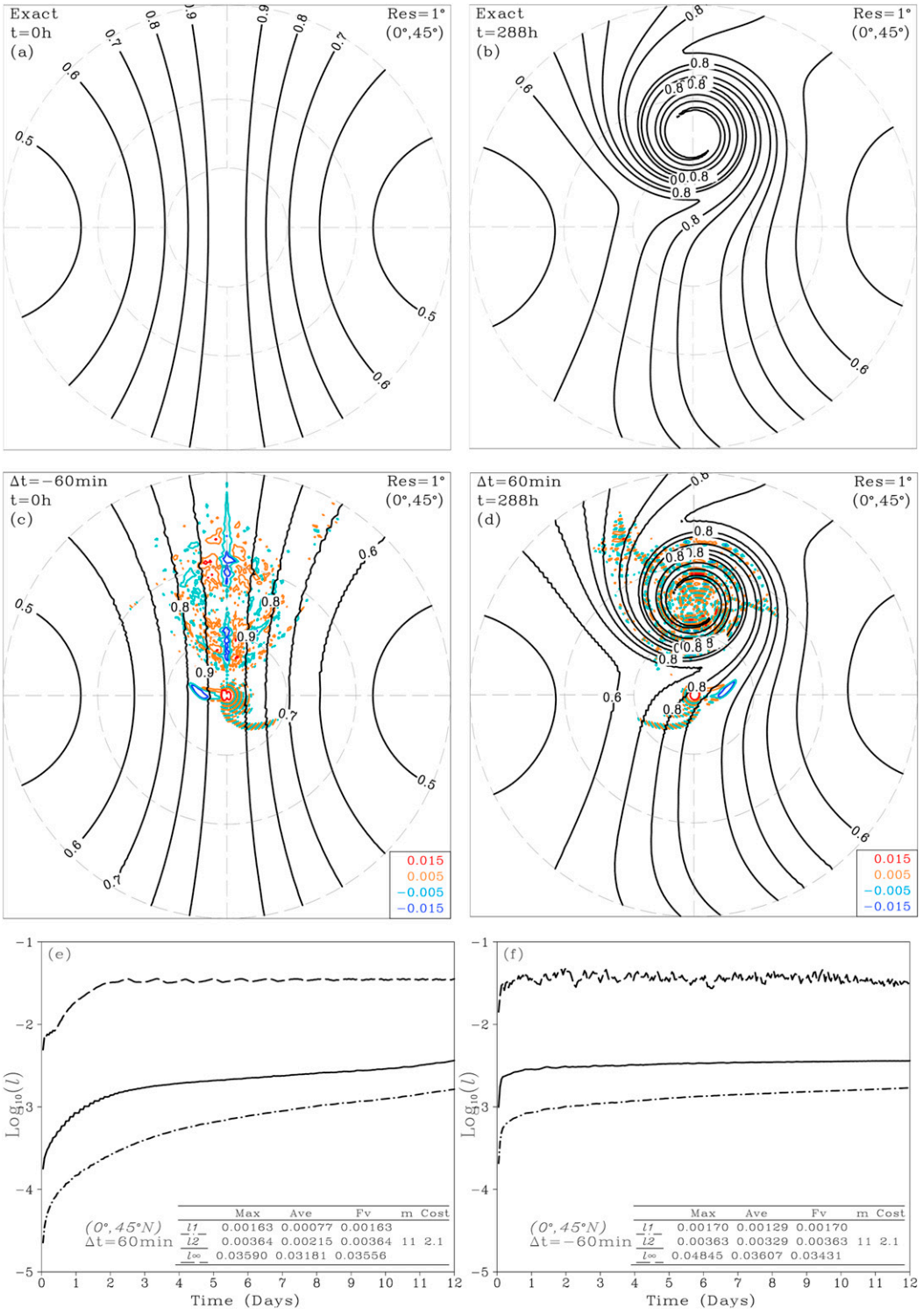

FIG. 10. The errors (colored) and solutions of deformational flow tests with vortex center located at 0°, 45°N for (a),(b) exact solutions; (c) the solution of backward simulation whose initial field is (b) and the exact solution is (a); (d) the solution of forward model whose initial field is (a) and the exact solution is (b); and (e),(f) the time evolution of error norms of forward and backward simulation, respectively.

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

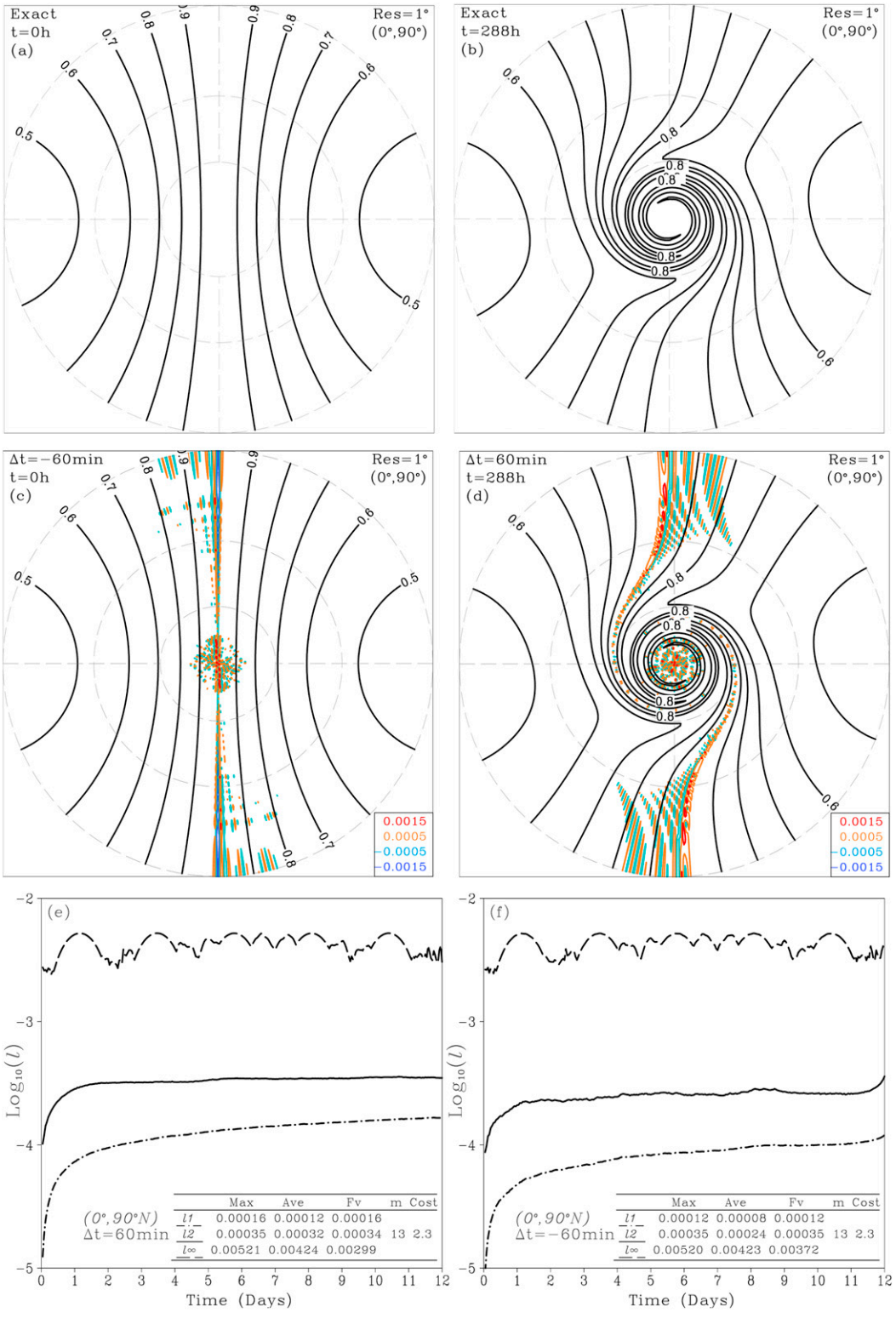

FIG. 11. As in Fig. 10, but for vortex center located at the North Pole.

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

TABLE 2. Comparison of error norms from the published schemes for the deformational flow tests when the vortex center is located at the North Pole.

	Day 6			Day 12			Day 20		
	l_1	l_2	l_∞	l_1	l_2	l_∞	l_1	l_2	l_∞
SCEIM 1°	1.35×10^{-4}	3.45×10^{-4}	3.90×10^{-3}	1.65×10^{-4}	3.49×10^{-4}	2.99×10^{-3}			3.49×10^{-3}
SLICE-S 2.1825°	8.6×10^{-4}	2.56×10^{-3}	1.55×10^{-2}	3.44×10^{-3}	1.14×10^{-2}	8.02×10^{-2}			
EPI2 1.0°	—	—	1.38×10^{-3}	—	—	—	—	—	3.47×10^{-3}

$$\begin{cases} u_d = rw(\theta')[\sin\theta_p \cos\theta - \cos\theta_p \cos(\lambda - \lambda_p)\sin\theta] \\ v_d = rw(\theta')[\cos\theta_p \sin(\lambda - \lambda_p)] \end{cases}, \tag{50}$$

$$\begin{cases} w(\theta') = v_0 \dfrac{3\sqrt{3}}{2\rho} \operatorname{sech}^2(\rho)\tanh(\rho) \\ \rho = 3.0\cos(\theta') \end{cases}. \tag{51}$$

Here (λ_p, θ_p) is the center of the vortex, which is independent of time in this stationary setup, and (λ', θ') is a position in a rotated coordinate system whose north pole is at (λ_p, θ_p).

1) TEST FOR FORWARD AND BACKWARD SIMULATIONS

The spatial resolution of tests is chosen to be $1° \times 1°$, and the time step is set to be $\Delta t = 60$ min for forward simulation tests and $\Delta t = -60$ min for backward simulation tests. Two types of test with stationary vortex were preformed, one is the test with $(\lambda_p, \theta_p) = (0°, 45°\text{N})$, the other is the test with $(\lambda_p, \theta_p) = (0°, 90°\text{N})$. For the latter, the LSFM method is not used because the meridional velocity field given by Eq. (50) is zero, which means that the coefficient $C_{ur} = v/rtg\theta = 0$, thus there is no rapid error growth for the ρ in polar region.

First, all of the tests show that the total mass is able to be accurately maintained, whose relative errors are indeed very small, not exceeding 1×10^{-14} (Figure omitted). Figure 10 shows the test results whose initial vortex is located at $(\lambda_p, \theta_p) = (0°, 45°\text{N})$. For the forward test, the front is gradually distorted with growing gradient under the force action of vortex flow field, and developed into the sharp front shown in Fig. 10d, which exhibits a very similar structure to the exact solution depicted in Fig. 10b. To solve the inverse problems of frontogenesis showed in Fig. 10b, it was used as the initial field of the backward model and integrated with $\Delta t = -60$ min, the numerical solution of this inverse problem is depicted in Fig. 10c, whose exact solution is shown in Fig. 10a. It can be seen from Fig. 10c that the errors are mainly distributed outside of the vortex center and near the North Pole, and the contour lines are not smooth. Nevertheless, the overall shape is in agreement with the exact solution. Figures 10e and 10f show that the error norms only grow rapidly in the initial stage, both the l_1 and l_2 are smooth, while the l_∞ present the shape of wave. The error norms of forward simulation are similar to those of backward simulation but little better. Both the forward and backward simulation have the same dimension of $\mathbf{K}_m$ subspace and the computational cost ($m = 11$ and Cost = 21 min). Figure 11 shows the test results with $(\lambda_p, \theta_p) = (0°, 90°\text{N})$. It can be seen that the results depicted in the Figs. 11c and 11d are in quite agreement with their exact solutions depicted in Figs. 11a and 11b, respectively. The error norms shown in Figs. 11e and 11f are better than those in Figs. 10e and 10f, moreover, Cost = 23 min and $m = 13$ are little more than the tests with $(\lambda_p, \theta_p) = (0°, 45°\text{N})$. Note that for the test with $(\lambda_P, \theta_P) = (0°, 90°)$, the zonal advection surround the north polar plays an important role and the LSFM is not used, it demonstrates that the central

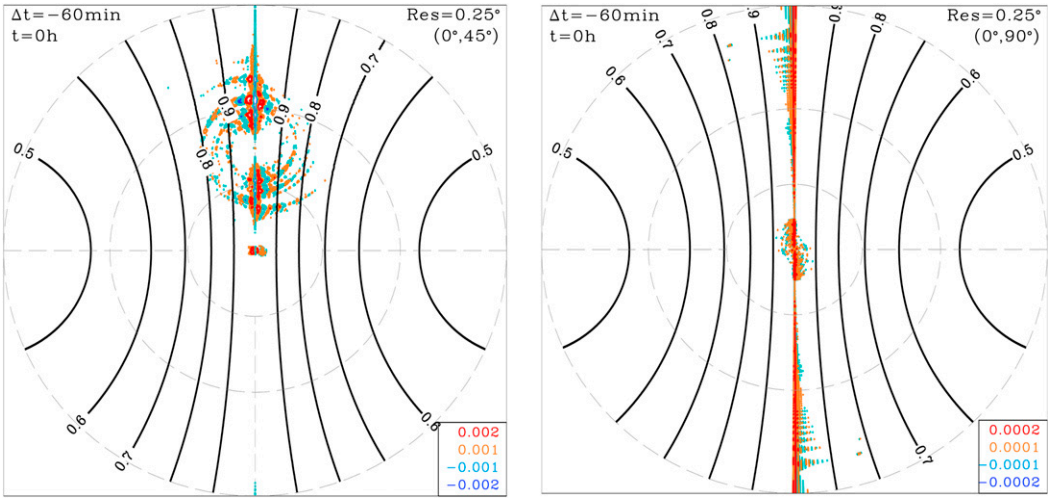

FIG. 12. The errors (shaded) and solutions (contours) of backward simulations of deformational flow tests whose initial fields are Figs. 10b and 11b, respectively, but the spatial resolution is 0.25°.

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

skip grid difference is an effective method to improve the problem of longitudinal grid distance near polar points.

Table 2 shows the comparison of error norms from SCEIM, SLICE-S in Zerroukat et al. (2004) and EPI2 in Clancy and Pudykiewicz (2013), in which EPI2 is also an exponential integration method, but implemented in an icosahedral model. Although the error norms of SLICE-S method is resulted from the test with resolution of 2.1825°, the errors of SCEIM method with resolution of 1° is about one order of magnitude smaller than those of SLICE-S method at day 12 and we also achieved comparable accuracy of l_∞ to EPI2 scheme at the day 6 and 20.

2) Test for grid convergence of backward simulations

For the backward tests with spatial resolution of 1°, the larger errors make the contour lines not smooth enough. It is closely related to the steep gradient in initial field of backward simulations. If the spatial resolution of the model is not sufficiently high, the errors of the simulated front gradient is worse, which would lead to the rapid growth of l_∞ in the initial stage (Fig. 10f). Consequently, the spatial resolution of the backward model is improved to 0.5° × 0.5° and 0.25° × 0.25° to test whether backward simulations are able to be improved. Figure 12 shows the results of the backward simulations with resolution of 0.25°. Compared with the results with resolution of 1° (Figs. 10c and 11c), the errors depicted in Fig. 12 are significantly improved by about one order of magnitude, the contours are smoother and visually identical to the exact solutions. Figure 13 shows that error norms became better and better as the spatial resolution increase, and it can be seen from the table that the average values of l_2 with resolution of 0.25° are about 1/7 of those with resolution of 1°.

3) Test for time convergence of nonlinear transport problems

It is important to note that the wind vector given by Eq. (48) for the cosine bell tests and that given by Eq. (50) are both independent of time, which means that the two kinds of tests are linear transport problems. The results from Fig. 5 showed that the SCEIM is very exact when solving linear transport problems so the error norms are independent of Δt without considering the influence the polar problems. Therefor it is necessary to verify the time convergence of SCEIM method for solving nonlinear transport problems. For this purpose, we choose the moving vortex tests, whose time-dependent wind field is given by $(u_m, v_m) = (u_c + u_d, v_c + v_d)$, in which the deforming vortices move along the tracks of cosine bells (Nair and Jablonowski 2008). In order not to be seriously affected by the polar problems, the moving direction is set to be $\alpha = 0$. Moreover, three Adams–Bashforth schemes are used to linearize Eq. (21) and compare them with the leapfrog scheme of Eq. (23), as follows:

$$\int_{t_n}^{t_{n+1}} [\mathbf{U}(t) + \mathbf{V}(t)]\, dt = \sum_{k=1}^{N_A} h_k [\mathbf{U}(t_{n-k+1}) + \mathbf{V}(t_{n-k+1})], \qquad (52)$$

where the N_A is the order of Adams–Bashforth scheme. The first-order Adams–Bashforth scheme [AB_O(Δt)], in which

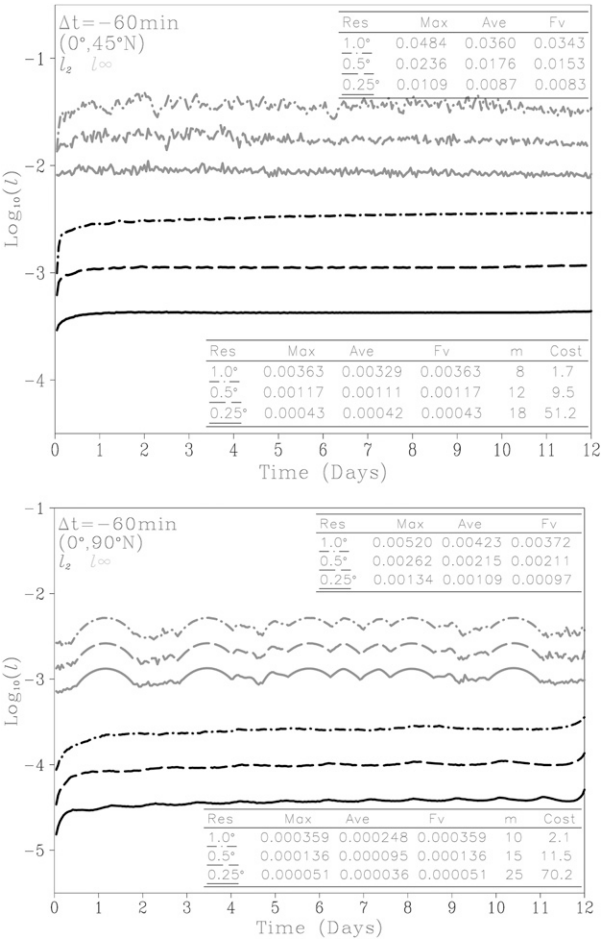

FIG. 13. The time evolution of error norms of backward simulations of deformational flow tests.

$h_1 = 1.0$, is the familiar Eular difference. For the second-order scheme [AB_O(Δt^2)],the coefficients are set to be $h_1 = 1.5$ and $h_2 = -0.5$, respectively. The third-order scheme [AB_O(Δt^3)], in which $h_1 = 23/12$, $h_2 = -4/3$, and $h_3 = 5/12$, is discussed by Durran (1991). The second-order leapfrog scheme, which is used in Eq. (23), is denoted by LF_O(Δt^2).

Table 3 shows the error norms of moving vortex tests with the four types of Linearization schemes and various Δt in the case of 0.25° spatial resolution and twentieth-order spatial difference scheme. It can be seen that the AB_O(Δt) scheme is the worst, and the l_2 and l_∞ are almost 10 times and 6 times as much as those of AB_O(Δt_2) when $\Delta t \geq 40$ min. Although the first-order scheme has faster time convergence for error norms, the l_2 and l_∞ with $\Delta t = 5$ min are worse than those of AB_O(Δt^3) with $\Delta t = 60$ min and LF_O(Δt^2) with $\Delta t = 40$ min. For the other schemes, although AB_O(Δt^2) and LF_O(Δt^2) are both second-order schemes, AB_O(Δt^2) is better than LF_O(Δt^2) when $\Delta t = 60$ min is used, while LF_O(Δt^2) is better than AB_O(Δt^2) when $\Delta t = 40$ min is used. In the case of $\Delta t \leqq 20$ min, the two second-order schemes have similar accuracy. By comparing with the error norms of LF_O(Δt^2), the error norms of AB_O(Δt^3) are better than those of LF_O(Δt^2) when

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

TABLE 3. Error norms of moving-vortex tests with four linearization schemes at day 6 and day 12 in the case of 1.0 spatial resolution and twentieth-order spatial central difference scheme. The AM and LF represent the Adams–Bashforth and the leapfrog scheme, respectively.

		AM_O(Δt)		AM_O(Δt^2)		AM_O(Δt^3)		LF_O(Δt^2)	
	Δt (min)	l_2	l_∞	l_2	l_∞	l_2	l_∞	l_2	l_∞
Day 6	60	0.007 04	0.042 28	0.000 70	0.006 05	0.000 36	0.005 32	0.001 61	0.019 93
	40	0.004 74	0.029 15	0.000 43	0.005 51	0.000 35	0.005 46	0.000 37	0.004 83
	20	0.002 41	0.015 95	0.000 33	0.005 47	0.000 33	0.005 47	0.000 32	0.005 51
	10	0.001 24	0.009 51	0.000 33	0.005 30	0.000 32	0.005 30	0.000 33	0.005 46
	5	0.000 69	0.006 17	0.000 33	0.005 29	0.000 32	0.005 30	0.000 33	0.005 47
Day 12	60	0.013 52	0.070 21	0.001 51	0.013 55	0.000 95	0.012 62	0.003 78	0.038 65
	40	0.009 20	0.050 09	0.001 06	0.012 64	0.000 92	0.012 65	0.001 00	0.011 21
	20	0.004 74	0.029 61	0.000 92	0.012 65	0.000 91	0.012 65	0.000 92	0.012 65
	10	0.002 52	0.018 81	0.000 91	0.012 65	0.000 91	0.012 65	0.000 91	0.012 60
	5	0.001 50	0.013 91	0.000 91	0.012 65	0.000 91	0.012 65	0.000 91	0.012 60

$\Delta t = 60$ min, while their error norms are close to each other for other time steps. For nonlinear transport problems, these four linearization schemes are all time-convergent schemes, and the third-order Adams–Bashforth scheme is slightly better than the second-order LF_O(Δt^2) scheme. However, the third-order Adams–Bashforth scheme is the best choice for using large time step.

7. Conclusions

SCEIM has been successfully generalized in a two-dimensional spherical latitude–longitude coordinate system. The most significant advantage of SCEIM is to change the forward model to the backward model by setting negative time step, which is an advantage that other published schemes do not have. The main emphasis in the development of SCEIM has been on conservation, positive-definite, and reversibility as well as achieving comparable accuracy to other published schemes. Moreover, the polar problem has also been resolved by using a simple effective central skip-point difference scheme and an LSFM without major penalty on the overall effectiveness of SCEIM.

SCEIM is evaluated by standard cosine bell tests and deformational flow tests. The cosine bell tests show that SCEIM is a time-convergence method as well as a grid-convergence method, which also demonstrate that SCEIM has a strong shape-preserving ability. The error norms of the forward simulations and backward simulations are reasonable and very similar. The deformational flow tests show that the forward model can simulate frontogenesis with high accuracy, and the backward model can solve the inverse frontogenesis problems. The error norms of the inverse frontogenesis problems are improved significantly as the spatial resolution increased, which demonstrate that the backward model is also a grid-convergence like as the forward model. For nonlinear transport problems, although the SCEIM with leapfrog scheme is a time-convergence method, the third-order Adams–Bashforth scheme is the best choice for SCEIM when large time step is used

Acknowledgments. The authors thank the two anonymous reviewers for their careful reviews and suggestions on improving the presentation and numerical experiments of this paper. This research was supported jointly by Heavy Rain and Drought-Flood Disasters in Plateau and Basin Key Laboratory of Sichuan Province (Grant SZKT2017005), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant XDA23090101), and Zhejiang Public Welfare Research Project (Grant LGF19D050002)

APPENDIX

Algorithms 1 and 2

a. Algorithm 1: Lanczos Iteration for Antisymmetric Matrix

1: $\mathbf{H}_m = 0.0$; $\mathbf{B}^*_m(0) = \mathbf{b}_0$; $\mathbf{B}_m(0) = \mathbf{b}_0/\|\mathbf{B}^*_m(0)\|$! $\mathbf{B}_m(-1) \equiv 0$; ! $\mathbf{H}_m(-1, 0) \equiv 0$
2: do $k = 0$, m ! Assume that m is given
3: if $\mathbf{A}$ is antisymmetric then
4: $\mathbf{B}^*_m(k+1) = \mathbf{A}\mathbf{B}_m(k) - \mathbf{H}_m(k, k-1)\mathbf{B}_m(k-1)$
5: $\mathbf{H}_m(k, k+1) = \|\mathbf{B}^*_m(k+1)\|$; $\mathbf{H}_m(k+1, k) = -\mathbf{H}_m(k, k+1)$
6: else if $\mathbf{A}$ is symmetric then
7: $\mathbf{H}_m(k, k) = \mathbf{B}^{\mathrm{T}}_m(k)\mathbf{A}\mathbf{B}_m(k)$
8: $\mathbf{B}^*_m(k+1) = \mathbf{A}\mathbf{B}_m(k) - \mathbf{H}_m(k, k-1)\mathbf{B}_m(k-1) - \mathbf{H}_m(k, k)\mathbf{A}\mathbf{B}_m(k)$
9: $\mathbf{H}_m(k, k+1) = \|\mathbf{B}^*_m(k+1)\|$; $\mathbf{H}_m(k+1, k) = \mathbf{H}_m(k, k+1)$
10: end if
11: $\mathbf{B}_m(k+1) = \mathbf{B}^*_m(k+1)/\|\mathbf{B}^*_m(k+1)\|$
12: $k = k + 1$
13: enddo

b. Algorithm 2: Adaptive Lanczos Iteration for Antisymmetric Matrix

1: $\mathbf{H}_m = 0.0$; $\mathbf{B}^*_m(0) = \mathbf{b}_0$; $\mathbf{B}_m(0) = \mathbf{b}_0/\|\mathbf{B}^*_m(0)\|$; error = $\|\mathbf{B}^*_m(0)\|$; Tol = 10^{-7}; $k = 0$
2: do while (error > Tol)
3: $\mathbf{B}^*_m(k+1) = \mathbf{A}\mathbf{B}_m(k) - \mathbf{H}_m(k, k-1)\mathbf{B}_m(k-1)$
4: $\mathbf{H}_m(k, k+1) = \|\mathbf{B}^*_m(k+1)\|$;
5: $\mathbf{H}_m(k+1, k) = -\mathbf{H}_m(k, k+1)$
6: $\mathbf{B}_m(k+1) = \mathbf{B}^*_m(k+1)/\|\mathbf{B}^*_m(k+1)\|$
7: error = error$\|\mathbf{B}^*_m(k+1)\|t/(k+1)$
8: $m = k$; $k = k + 1$
9: enddo

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC

REFERENCES

Arakawa, A., 1972: Design of the UCLA general circulation model: Numerical simulation of weather and climate. Department of Atmospheric Sciences Tech. Rep. 7, University of California, Los Angeles, 116 pp.

Baba, Y., K. Takahashi, and T. Sugimura, 2010: Dynamical core of an atmospheric general circulation model on a Yin–Yang grid. *Mon. Wea. Rev.*, **138**, 3988–4005, https://doi.org/10.1175/2010MWR3375.1.

Beylkin, G., J. M. Keiser, and L. Vozovoi, 1998: A new class of time discretization schemes for the solution of nonlinear PDEs. *J. Comput. Phys.*, **147**, 362–387, https://doi.org/10.1006/jcph.1998.6093.

Bhatt, A., and B. E. Moore, 2017: Structure-preserving exponential Runge–Kutta methods. *SIAM J. Sci. Comput.*, **39**, A593–A612, https://doi.org/10.1137/16M1071171.

——, and ——, 2019: Exponential integrators preserving local conservation laws of PDEs with time-dependent damping/driving foresee. *J. Comput. Appl. Math.*, **352**, 341–351, https://doi.org/10.1016/j.cam.2018.12.003.

Botchev, M. A., 2013: A block Krylov subspace time-exact solution method for linear ordinary differential equation systems. *Numer. Linear Algebra Appl.*, **20**, 557–574, https://doi.org/10.1002/nla.1865.

Bott, A., 1989: A positive definite advection scheme obtained by nonlinear renormalization of the advective fluxes. *Mon. Wea. Rev.*, **117**, 1006–1016, https://doi.org/10.1175/1520-0493(1989)117<1006:APDASO>2.0.CO;2.

Clancy, C., and J. A. Pudykiewicz, 2013: On the use of exponential time integration methods in atmospheric models. *Tellus*, **65A**, 20898, https://doi.org/10.3402/tellusa.v65i0.20898.

Durran, D. R., 1991: The third-order Adams–Bashforth method: An attractive alternative to leapfrog time difference. *Mon. Wea. Rev.*, **119**, 702–720, https://doi.org/10.1175/1520-0493(1991)119<0702:TTOABM>2.0.CO;2.

Frommer, A., K. Lund, and D. B. Szyld, 2017: Block Krylov subspace methods for functions of matrices. *Electron. Trans. Numer. Anal.*, **47**, 100–126.

Gaudreault, S., G. Rainwater, and M. Tokman, 2018: KIOPS: A fast adaptive Krylov subspace solver for exponential integrators. *J. Comput. Phys.*, **372**, 236–255, https://doi.org/10.1016/j.jcp.2018.06.026.

Gong, J., B. Wang, and Z. Ji, 2013a: Three-step difference scheme for solving nonlinear time-evolution partial differential equations. *Atmos. Oceanic Sci. Lett.*, **6**, 423–427, https://doi.org/10.1080/16742834.2013.11447119.

——, ——, and ——, 2013b: Generalized square conservative multistep finite difference scheme incorporating historical observations. *Atmos. Oceanic Sci. Lett.*, **6**, 223–226, https://doi.org/10.1080/16742834.2013.11447084.

Hochbruck, M., C. Lubich, and H. Selhofer, 1998: Exponential integrators for large systems of differential equations. *SIAM J. Sci. Comput.*, **19**, 1552–1574, https://doi.org/10.1137/S1064827595295337.

Ji, Z., and Q. C. Zeng, 1982: The construction and application of difference scheme of evolution equations. *Sci. Atmos. Sin.*, **6**, 88–94.

——, and B. Wang, 1991: A further discussion on the construction and application of difference scheme of evolution equations (in Chinese). *J. Atmos. Sci.*, **15**, 72–81.

Koskela, A., 2015: Approximating the matrix exponential of an advection–diffusion operator using the incomplete orthogonalization method. *Numerical Mathematics and Advanced Applications—ENUMATH 2013*, A. Abdulle et al., Eds., Lecture Notes in Computational Science and Engineering, Vol. 103, Springer, 345–353.

Li, D., Y. Cong, and K. Xia, 2016: Flexible exponential integration methods for large systems of differential equations. *J. Appl. Math. Comput.*, **51**, 545–567, https://doi.org/10.1007/s12190-015-0919-1.

Loffeld, J., and M. Tokman, 2013: Comparative performance of exponential, implicit and explicit integrators for stiff systems of ODEs. *J. Comput. Appl. Math.*, **241**, 45–67, https://doi.org/10.1016/j.cam.2012.09.038.

Moler, C., and C. Van Loan, 1978: Nineteen dubious ways to compute the exponential of a matrix. *SIAM Rev.*, **20**, 801–836, https://doi.org/10.1137/1020098.

——, and ——, 2003: Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Rev.*, **45**, 3–49, https://doi.org/10.1137/S00361445024180.

Nair, R. D., and C. Jablonowski, 2008: Moving vortices on the sphere: A test case for horizontal advection problems. *Mon. Wea. Rev.*, **136**, 699–711, https://doi.org/10.1175/2007MWR2105.1.

Niesen, J., and W. M. Wright, 2012: Algorithm 919: A Krylov subspace algorithm for evaluating the φ-functions appearing in exponential integrators. *ACM Trans. Math. Software*, **38**, 1–19, https://doi.org/10.1145/2168773.2168781.

Skamarock, W. C., and M. L. Weisman, 2009: The impact of positive-definite moisture transport on NWP precipitation forecasts. *Mon. Wea. Rev.*, **137**, 488–494, https://doi.org/10.1175/2008MWR2583.1.

Tokman, M., 2006: Efficient integration of large stiff systems of ODEs with Exponential Propagation Iterative (EPI) methods. *J. Comput. Phys.*, **213**, 748–776, https://doi.org/10.1016/j.jcp.2005.08.032.

Vo, H. D., and R. B. Sidje, 2017: Approximating the large sparse matrix exponential using incomplete orthogonalization and Krylov subspaces of variable dimension. *Numer. Linear Algebra Appl.*, **24**, e2090, https://doi.org/10.1002/nla.2090.

Wang, B., and Z. Z. Ji, 1993: The harmonious dissipative operator and the completely square conservative difference scheme in explicit way. *Sci. China*, **37B**, 462–471, https://doi.org/10.1360/yb1994-37-4-462.

——, ——, and Q. C. Zeng, 1993: A kind of time-saving explicit numerical computational method. *Chin. Sci. Bull.*, **37**, 766–768.

——, Q. Zeng, and Z. Z. Ji, 1995: Square conservation systems and Hamiltonian systems. *Sci. China*, **38A**, 1211–1219.

——, H. Wan, Z. Ji, X. Zhang, R. Yu, Y. Yu, and H. Liu, 2004: Design of a new dynamical core for global atmospheric models based on some efficient numerical methods. *Sci. China*, **47A**, 4–21, https://doi.org/10.1360/04za0001.

Williamson, D. L., J. B. Drake, J. J. Hack, R. Jakob, and P. N. Swarztrauber, 1992: A standard test set for numerical approximation to the shallow water equations in spherical geometry. *J. Comput. Phys.*, **102**, 211–224, https://doi.org/10.1016/S0021-9991(05)80016-6.

Xiao, F., and X. D. Peng, 2004: A convexity preserving scheme for conservative advection transport. *J. Comput. Phys.*, **198**, 389–402.

Zeng, Q. C., and Z. Z. Ji, 1981: On the computational stability of evolution equations. *Chin. Comput. Math.*, **3**, 79–86.

Zerroukat, M., N. Wood, and A. Staniforth, 2004: SLICE-S: A Semi-Lagrangian Inherently conserving and Efficient scheme for transport problems on sphere. *Quart. J. Roy. Meteor. Soc.*, **130**, 2649–2664, https://doi.org/10.1256/qj.03.200.

Zhang, Y., R. Yu, J. Li, and H. M. Chen, 2013: An implementation of a leaping-point two-step shape-preserving advection scheme in high-resolution spherical latitude-longitude grid (in Chinese). *Acta Meteor. Sin.*, **71**, 1089–1102, https://doi.org/10.11676/qxxb2013.085https://doi.org/10.1007/s10114-013-2274-1.

Brought to you by NOAA Central Library | Unauthenticated | Downloaded 07/19/23 06:17 PM UTC