A Spectral Deferred Correction Method Applied to the Shallow Water Equations on a Sphere

JUN JIA, JUDITH C. HILL, KATHERINE J. EVANS, AND GEORGE I. FANN
Oak Ridge National Laboratory, Oak Ridge, Tennessee

MARK A. TAYLOR
Sandia National Laboratories, Albuquerque, New Mexico

(Manuscript received 13 February 2012, in final form 7 December 2012)

ABSTRACT

Although significant gains have been made in achieving high-order spatial accuracy in global climate modeling, less attention has been given to the impact imposed by low-order temporal discretizations. For long-time simulations, the error accumulation can be significant, indicating a need for higher-order temporal accuracy. A spectral deferred correction (SDC) method is demonstrated of even order, with second- to eighth-order accuracy and A-stability for the temporal discretization of the shallow water equations within the spectral-element High-Order Methods Modeling Environment (HOMME). Because this method is stable and of high order, larger time-step sizes can be taken while still yielding accurate long-time simulations. The spectral deferred correction method has been tested on a suite of popular benchmark problems for the shallow water equations, and when compared to the explicit leapfrog, five-stage Runge–Kutta, and fully implicit (FI) second-order backward differentiation formula (BDF2) time-integration methods, it achieves higher accuracy for the same or larger time-step sizes. One of the benchmark problems, the linear advection of a Gaussian bell height anomaly, is extended to run for longer time periods to mimic climate-length simulations, and the leapfrog integration method exhibited visible degradation for climate length simulations whereas the second-order and higher methods did not. When integrated with higher-order SDC methods, a suite of shallow water test problems is able to replicate the test with better accuracy.

1. Introduction

Accurate simulation over long time periods is necessary for the investigation of global-scale climate dynamics because the impacts and societal implications are great. These simulations require the resolution of atmospheric features with time scales covering multiple orders of magnitude. For example, variations in the strength of synoptic-scale weather features and tropical cyclones occur within a day; however, global climate change, under which these features could be affected, occurs over decades.

Currently, global climate models are used to perform a variety of experiments, but most fall into the following categories:

(i) comparison of simulated present-day climate (approximately spanning the range of years 1970–2011) with observations,

(ii) near-term future climate scenarios (within the next 100 yr) with varied settings to evaluate a model's response to external and internal forcings,

(iii) paleoclimate simulations that model the earth's climate during much earlier periods including previous geological times, and, more recently,

(iv) investigations of subglobal atmospheric features that require high spatial resolutions to resolve local phenomena, such as tropical cyclones.

Present versions of the dynamical component (dycore) of these models rely on explicit and semi-implicit stability-limited lower-order (second or lower) time-integration schemes so that the time-step size depends on the spatial resolution (Neale et al. 2010). Operationally, the largest stable time-step size is taken to minimize the time to solution, without considerable attention being paid to the temporal accuracy.

Recent experiments investigating the present-day record and near-term future climate (categories i and ii above) generally use $2^\circ$ to $1^\circ$ resolutions, which correspond...
to a maximum time-step size of approximately 20–10 min on relatively even grids (e.g., Collins et al. 2004), respectively. Future scenario simulations run for a longer time and generally use a 2° or use a 1° horizontal resolution as well. In contrast, climate science numerical experiments that fall into categories iii and iv are configured differently to produce relevant data. The paleoclimate simulations of category iii typically use coarse 3.75° grids and thus use a larger 30-min time step. However, the simulations extend 8000+ yr in length (e.g., Liu et al. 2009; Winguth et al. 2010), corresponding to about 10^8 time steps. Alternatively, present-day simulations in category iv are configured for regional and shorter-term climate features, and so they cover only several decades (McClean et al. 2011). Because fine-scale resolutions of about 1/8° to 1/4° horizontal resolution are stability limited and use 150 s and smaller time-step sizes to integrate the dynamical variables, a total of at least 6 × 10^6 times steps are needed for high-resolution simulations to remain stable.

The need for more accurate temporal discretizations has recently been receiving attention. The continuous Galerkin version of the High-Order Methods Modeling Environment (HOMME) model, which has become the default dynamical core option within the recently released Community Atmosphere Model (CAM), version 5.2 (Neale et al. 2010), uses a second-order five-stage Runge–Kutta (RK2) method. However, the size of the time step is still stability limited. A fully implicit (FI) second-order backward differentiation formula (BDF2) method has been implemented (Evans et al. 2010) into the shallow water version of the HOMME spectral element model. The FI option in HOMME uses the second-order BDF2 discretization of the Jacobian-free Newton–Krylov (JFNK) method to generate solutions to the available benchmark suite of shallow water test cases of Williamson et al. (1992) for a range of time steps above the stability limits that restrict the existing explicit and semi-implicit time integration schemes. Efficiency is yet to be achieved, but will be improved once preconditioning methods have been developed for these schemes; a discussion of this is provided at the end of the methods section.

Larger time steps reduce the temporal-error accumulation due to numerical round-off errors during the course of a computer simulation for long-time events; otherwise, the solution will lose precision as simulations proceed, even with high-order spatial approximations. To illustrate this, a basic test problem from Williamson et al. (1992), the linear advection of a Gaussian bell height anomaly, is extended to cover a longer time period to correspond to climate rather than weather time scales. In as little as 1 month of simulation time, which is about 10,000 time steps, the second-order-filtered explicit leapfrog method retains only two significant digits. For 10,000 simulation days, or about 3.5 million time steps, there are no longer any significant digits of accuracy, with an L_2 error of 0.65, as shown in Fig. 1, where the L_2 norm of a function f is defined as (∫ f^2)^(1/2).

By implementing an implicit spectral-in-time spectral deferred correction (SDC) method of even order, from second to eighth order, for the two-dimensional shallow water equations in HOMME, we demonstrate below that high-order time-stepping methods improve the long time accuracy of the shallow water test cases presented. The accuracy maintained even when taking the large time steps permitted by SDC for modeling climate
dynamics is advantageous because the temporal error accumulation is reduced for long-time simulations compared to the temporal discretizations currently used in practice.

The SDC method, which was originally introduced by Dutt et al. (2000), constructs an arbitrarily high-order time-integration method based on a chosen low-order method, such as the first-order forward Euler method. It combines the ideas of iterative deferred correction (Zadunaisky 1966; Pereyna 1968), the Picard integral equation, and spectral integration. Recently, an arbitrary high-order SDC method has been applied successfully to the solution of the Euler equations, the gas-dynamics equations, and the Navier–Stokes equations (Minion 2004; Layton and Minion 2004; Jia and Liu 2011).

This paper is organized as follows. A brief overview of the shallow water equation dycore and the SDC method as implemented in HOMME is summarized in section 2. The results from a temporal error analysis of a suite of benchmark shallow water test cases in HOMME using the SDC method are presented in section 3 and we discuss these numerical results and their implications in section 4.

2. Methodology

In this section, we introduce the shallow water formulation and underlying discretization scheme used in HOMME. Then, we briefly discuss the mathematical underpinnings of the SDC method and provide a brief analysis of the computational cost incurred by this higher-order method.

a. Shallow water formulation in the HOMME model

Global atmospheric modelers typically implement and test novel solution schemes with the two-dimensional shallow water equations because they are simpler than the full atmospheric model, yet still retain many of the important physical scales and interactions (e.g., Lin and Rood 1997; Bonaventura and Ringler 2005). The two-dimensional shallow water equations in HOMME are constructed in vector invariant form,

$$\frac{\partial \mathbf{v}}{\partial t} = - (\zeta + f) \mathbf{k} \times \mathbf{v} - \mathbf{V} \left( \frac{1}{2} \mathbf{v} \cdot \mathbf{v} + gH \right)$$

$$\frac{\partial z}{\partial t} = - \nabla \cdot (z \mathbf{v}),$$  \hspace{1cm} (1)

with some initial conditions, where $\mathbf{v} = (u, v)^T$ is the two-dimensional velocity vector. The vorticity, gradient, and divergence operators are given by $\zeta$, $\mathbf{V}$, and $\nabla$, respectively. The unit vector $\mathbf{k}$ is the normal vector to the $x$–$y$ plane. The geopotential height of the fluid is $gH$, where $g$ is gravity and $H = z + z_s$, where $z$ is the thickness of the fluid above the bottom topography $z_s$. The Coriolis parameter is $f = 2\Omega \sin\theta$, where $\Omega$ is the angular rate of rotation and $\theta$ is the latitude.

The spherical surfaces in the HOMME dycore are divided by inscribing a cube within the sphere and using an equal-angular gnomonic projection to map the cube to the sphere (Rancic et al. 1996). Each cube face is further subdivided into a user-specified number of quadrilateral elements. Refer to the left side of Fig. 2 for an illustration. Each element is represented with a nodal...
basis using a tensor product of Gauss–Lobatto–Legendre (GLL) quadrature points [see earlier development citations, e.g., Nair et al. (2009), for a fuller explanation of this discretization]. This layout allows for spatial refinement through the variation in element count along a cube edge \( ne \) and the spectral order within an element \( np \), each of which has benefits and downsides (Taylor et al. 1997). To illustrate this discretization, an example with 8 elements per cube face edge \( (ne = 8) \) and fourth spectral order \( (np = 4) \) is given in Fig. 2. The horizontal layout of HOMME is referred to as an unstructured grid with a “cube–sphere” layout, so the difference in element size is minimal, as shown by the color bar in Fig. 2. Conversely, the intra-element edge spatial layout becomes highly irregular with increasing spectral order due to the location of the nodal basis points, so stability-limited time-integration methods can be severely restricted. Descriptions of the discretization details for the spectral element (SE) method in HOMME as well as other attributes of the scheme, including the local conservation of mass and moist total energy, are outlined in detail in Evans et al. (2013) and other articles cited within. The efficiency of the SE method is maximized by distributing the elements on supercomputing platforms within. The efficiency of the SE method within CAM has shown scalability and improved efficiency at more than 100 000 cores (Dennis et al. 2012; Worley et al. 2011) when solving the full primitive hydrostatic equations.

**b. The spectral deferred correction method**

The flexibility in the spatial distribution of points around the sphere can be fully realized with equally flexible time-discretization options. The SDC method is a variable-order temporal discretization scheme that converts a discretized time-dependent ordinary differential equation or a partial differential equation such as (1) to a corresponding Picard integral equation. The time step is subdivided, and an error equation is initially approximated on each substep using Gauss–Legendre quadrature points. In essence, an interpolating polynomial based on the quadrature points is used to approximate the solution in time, similar to the SE method in space; the Gauss–Legendre quadrature points are now exactly the collocation points on the time axis. A recursive sequence of error equations of the same form is derived for each application of a solution method. For efficiency, the error equation can be solved approximately using a low-order method, which can then be used to iteratively correct the solution to higher accuracy. Each error correction improves the accuracy and the precision of the initially approximated solution.

A schematic demonstration is shown in Fig. 3. For more detail about SDC, please refer to Dutt et al. (2000) and Huang et al. (2006), and the references therein.

The resulting spectral deferred correction implicit time-integration method is A-stable and L-stable (Dutt et al. 2000; Huang et al. 2006), with a larger region of convergence when compared with other higher-order methods. Presently, the backward Euler (BE) method is chosen to be the low-order solver for constructing an implicit SDC. The BE method is implicit and robust for large time steps applied to stiff problems and also has low storage requirements. Explicit forward Euler time stepping may also be used to develop an explicit SDC scheme for nonstiff problems. Presently, we demonstrate the accuracy of the SDC using the A-stable implicit formulation since the stiffness is not known a priori in a production environment, especially as the method is extended to more complex and complete fluid equations.

1) **THE MODIFIED PICARD INTEGRAL EQUATION AND THE ERROR EQUATION**

Without loss of generality, we rewrite the shallow water equations (1) in terms of the solution vector \( \varphi = (u, v, z)^T \). The right-hand side of (1) can be then denoted as \( G(\varphi) \), and the equation becomes

\[
\frac{d\varphi}{dt} = G(\varphi). \tag{2}
\]

We also assume an initial value \( \varphi_0 \) is given at time \( t = 0 \).

Following Huang et al. (2007), to apply the SDC method, the first step is to derive the error equation equivalent to (2) in Picard integral form. Defining \( \psi = \varphi - \varphi_0 \), we have

\[
\varphi = \varphi_0 + \int_0^t \psi \, dt. \tag{3}
\]
Rewriting (2) in terms of $\psi$, we obtain a Picard integral type equation for $\psi$

$$\psi = G(\varphi_0 + \int_0^t \psi \, dt).$$

(4)

Suppose that an approximate solution $\varphi^{(0)}$ to (2) is given in terms of $\varphi^{(0)}$ as

$$\varphi^{(0)} = \varphi_0 + \int_0^t \psi^{(0)} \, dt.$$  

(5)

Let $\varepsilon$ denote the error:

$$\varepsilon = \psi - \psi^{(0)}.$$  

(6)

Assuming no error at time zero, substituting $\psi = \psi^{(0)} + \varepsilon$ into (4) and recognizing that (5) yields the modified Picard integral equation for the error produces

$$\psi^{(0)} + \varepsilon = G\left[\varphi^{(0)} + \int_0^t \varepsilon \, dt\right].$$

(7)

The error equation (7) is equivalent to (2), and solving this error equation is not any easier than solving the original one. However, this form gives us an opportunity to improve the approximate solution iteratively until $\varepsilon$ reaches a desired tolerance.

2) Euler’s Methods Using Gaussian Quadrature Nodes

Given (2) in the form of (7), we present the SDC framework for one time-step interval $(t_n, t_{n+1})$. The SDC method divides $(t_n, t_{n+1})$ into subintervals $t_n \leq \tau_1 < \tau_2 < \cdots < \tau_p \leq t_{n+1}$ by $p$ grid points $\tau_1, \ldots, \tau_p$. An approximate solution for $\varphi^{(0)}(\tau_m)$ is computed for $m = 1, \ldots, p$ based on an order-$k$ method B (where A could be backward Euler, Runge–Kutta, etc.). Once we have an initial approximate solution $\psi^{(0)}$, iteratively a sequence of $l$ solutions to the correction equation (7) at $\tau_m$ is solved by an order-$k$ method B, improving the approximate solution. We approximate the integral in (5) using the $\tau_m$ as quadrature points. Therefore, the temporal order of accuracy is $\min(O_p, j + lk)$, where $O_p$ is the accuracy of the quadrature (Huang et al. 2006; Hagstrom and Zhou 2006). The SDC framework is general; it leaves the choice of the methods A and B and the quadrature to the user.

The temporal dependence is explicitly used for each of the functions in the time discretization. For convenience, defining $\tau_0 = t_n$, the initial condition gives $\varepsilon(\tau_0) = 0$. At grid point $\tau_m$, we use (7) to obtain

$$\varepsilon(\tau_m) = G\left[\varphi^{(0)} + \int_0^{\tau_m} \varepsilon \, dt\right] - \psi^{(0)}.$$  

(8)

Letting $e_m = \varepsilon(\tau_m)$, the backward Euler discretization of (8) is

$$e_m = G[\varphi^{(0)}_{m} + S_\varepsilon(m)] - \psi^{(0)}_m,$$  

(9)

where $S_\varepsilon(m) = \sum_{j=1}^{m-1} e_j(\tau_j - \tau_{j-1})$ is the right Riemann sum approximating the integral of $\varepsilon$ from 0 to $\tau_m$.

Denoting the “low order” approximation of $e(t)$ by $e^{(1)}(t) = (e_1, e_2, \ldots, e_p)$, a refined solution is given by $\psi^{(1)} = \psi^{(0)} + e^{(1)}$ and the order of accuracy of the approximate solution increases by one, which is the order of the underlying Euler methods. The SDC method iterates until the error or correction $\varepsilon$ is smaller than a prescribed tolerance.

To complete the discretization, the terms $\varphi^{(0)}_{m}$ are computed with $\varphi^{(0)}_{m}$ on the $p$ quadrature points $\tau_m$ using spectral integration. To define the spectral integration, we form an interpolating polynomial by the values of $\psi^{(0)}$:

$$LP[\psi^{(0)}, t] = \sum_{m=1}^{p} c_{m}(t),$$

where

$$c_{m}(t) = \prod_{k \neq m} \frac{t - \tau_k}{\tau_m - \tau_k}.$$  

(10)

The spectral integration is a linear map from the function values $\psi^{(0)}$ to $\psi^{(0)}$, the values of the integral of $LP[\psi^{(0)}, t]$ at those $p$ points, so (5) becomes

$$\varphi^{(0)}_{m} = \varphi_0 + \int_{t_n}^{\tau_m} LP[\psi^{(0)}, t] \, dt = \varphi_0 + (t_{n+1} - t_n) \sum_{m=1}^{p} \psi^{(0)}_{m} \int_{0}^{\tau_m-t_n} (t' + t_n - t'_{m-1}) \, c_{m}(t') \, dt'.$$

This map can be represented by the so-called spectral integration matrix. Denoting the $p \times p$ spectral integration matrix by $S$, (3) in matrix form is

$$\varphi^{(0)} = \varphi_0 + (t_{n+1} - t_n) S \psi^{(0)} = \varphi_0 + (t_{n+1} - t_n) S \psi^{(0)},$$

(10)

where $\varphi_0$ is $\varphi_0$ replicated into $p$ dimensions. In this case, $S$ is dense and can be precomputed once the quadrature points are chosen.

Each backward Euler step on the quadrature nodes, given by (9), is solved to obtain $e_{m+1}$ using the nonlinear

\[1\text{Similarly, for the forward Euler method, the integral is computed by the left Riemann sum: } S_\varepsilon(m) = \sum_{j=1}^{m-1} e_j(\tau_j - \tau_{j-1}).\]
Jacobian-free Newton–Krylov (JFNK) method with the generalized minimal residual method (GMRES) linear solver option from the NOX package of the Trilinos libraries (Heroux et al. 2005). Other choices are possible, but this method is already in use with FI BDF2 in HOMME, and given the present goal of establishing the accuracy and stability of SDC, this method provided the necessary stability and accuracy from which to extend to SDC. A more detailed description of FI is given in Knoll and Keyes (2004).

Presently, the goal is to establish that SDC methods can provide improved accuracy while taking large, stable step sizes for the shallow water equations within a cubed-sphere spectral-element framework. It has been shown that explicit Euler-based SDC methods have a similar or larger absolute stability region than comparable RK methods (Dutt et al. 2000), and so we present a similar extension of the JFNK method with the aim of performing additional development to optimize the efficiency of the scheme. A key issue is whether the improved accuracy and stability maintains a reasonable computational cost; otherwise, the method cannot be practical for production use.

As mentioned, improved efficiency for SDC depends on the overall cost of a set of large time steps over the course of a simulation compared to an equivalent set of smaller time steps of the stability-limited and lower-order method (assuming it also has acceptable temporal accuracy) for the problem to be solved. With SDC, the overall computational cost hinges mostly on the cost of the integration method used for the provisional and correction steps of the algorithm, and that step has not been optimized in terms of best utilization of the solver library and the use of scalable preconditioning, a key determination of the overall computational cost (Knoll and Keyes 2004). We do not address parallel computational issues here since the current implementation has not been optimized, but we can estimate whether a fully optimized implementation is worth pursuing by analyzing the number of implicit solves and their embedded right-hand-side (rhs) evaluations compared to current schemes. These issues are also an area of active research, but are mostly an independent question from the accuracy and stability of the SDC method presented here.

3) ALGORITHM COST

As an analysis of relative expense, we summarize the SDC algorithm in terms of its computational cost and the performance of its components to motivate further research into this scheme as an option to gain solution efficiency and accuracy. The current implementation of the SDC method for one time step using the notation presented above is described in algorithm 1 in the appendix.

For simplicity, we use the same BE first-order method to compute both the provisional solution and the corrections (method A is the same as method B).

As a gross measure of computational potential, the number of rhs evaluations of (1) needed to solve with this iterative scheme can be compared to the number of rhs evaluations for an explicit scheme, which enables an estimate of the relative cost. Thus, for $(2p)$th-order SDC, we can bound the computational cost for one time step at a maximum of $2p^2$ BE implicit solves times the average number of rhs evaluations per solve.

As mentioned above, the iteration and solver overhead with each iteration motivates work to develop and implement scalable preconditioning schemes such as those used in Yang and Cai (2011). Note that implementing a five-stage RK2 in HOMME to match the method used in the full three-dimensional CAM-SE simulation provides second-order accuracy, as shown in the test cases above; the stability region is smaller than for the leapfrog method, so its step size is reduced to about 80% of the leapfrog value. To evaluate SDC performance within this context, the key determination is whether the extra SDC expense can be overcome by the larger time steps that can be used while maintaining a desired accuracy, whether it be a method with a larger time-step size but lower accuracy or the converse.

There are also options to speed up the SDC calculations yet to be investigated for this problem. The Krylov deferred correction (KDC) method, which is an accelerated version of SDC, is expected to provide further speed up. Implicit KDC reduces the number of iterations to convergence and removes the order-reduction phenomenon for very stiff systems. Furthermore, KDC improves the stability properties of its semi-implicit–explicit SDC version (Huang et al. 2006, 2007). In the case of explicit KDC, the stability has been improved so dramatically that it can be used with mildly stiff systems. In any case, there is significant algorithmic complexity in the SDC algorithm, and each portion should be optimized to see maximum computational performance.

3. Numerical tests of the SDC method

A number of previous studies have used some or all of the Williamson et al. (1992) test cases to evaluate novel numerical schemes for global, spherical fluid flow models. Efforts to develop HOMME to improve the spatial accuracy and efficiency for global atmospheric modeling (Dennis et al. 2005) first demonstrated solution quality with various configurations within the shallow water dynamical core using these tests as the benchmark for acceptance (Taylor et al. 1997; Thomas and Loft 2002; Nair et al. 2005). These tests are familiar to the shallow water
methods community and span a range of problem complexity, so we use them as well in a temporal context to assess the behavior of the SDC method. We have focused on test cases 1, 5, and 6 in the Williamson et al. (1992) suite of cases to match what has been implemented and evaluated for the HOMME dynamical core to date (Thomas and Loft 2002; Levy et al. 2007; Evans et al. 2010). Although they are basic tests, we illustrate that second-order and higher methods in time are needed to attain sufficient accuracy when simple extensions are made in the simulation length to isolate the accumulation error in the temporal dimension. Although these tests were not specifically designed to evaluate temporal accuracy, a small but relevant subset of studies has used them for this purpose (Rouhi et al. 1999; Naughton et al. 1993). As further analysis, we also assess the implicit BE SDC method applied to a highly nonlinear flow problem that evolves on two time scales (Galewsky et al. 2004). Although this problem is very sensitive to the choice of diffusion, which is not the main focus here, it provides a benchmark for time-stepping methods for a problem with more complex dynamical features.

Note that the second test case within the Williamson suite was designed to assess stability-limited temporal methods because it solves the steady-state nonlinear flow around the globe for a 15-day period. As with the implicit BE error correction scheme that provides the base update for the higher-order SDC method, test case 2 is a trivial problem because the initial conditions that drive the flow have no Courant–Friedrichs–Lewy (CFL) restriction, so the initial solution is sufficiently close to the final solution independent of time-step size. It has been demonstrated that iterative convergence to a steady-state solution using any time step size is all that is required for an accurate solution to this problem when using fully implicit schemes (Evans et al. 2010).

a. Linear advection (TC1)

The first test case to evaluate the SDC method within the shallow water dynamical core of HOMME is the linear advection of a cosine anomaly around the globe over a 12-day period using prescribed zonal velocity fields, denoted TC1. A simple case, it is included because it allows a comparison to an analytical solution. A spatial discretization of four elements along each cube edge (\(ne = 4\)), which is 96 elements on the sphere, with a basis of degree 15 (\((np = 16)\) polynomials, is used to match earlier analyses of the spatial and temporal discretizations of HOMME (Taylor et al. 1997; Thomas and Loft 2002). Integrated with the CFL-limited time-step size of 30 s used by default explicit methods, the \(L_2\) norm of the error relative to the analytical solution is \(1.75 \times 10^{-3}\) for the FI-BDF2 scheme described in section 2 and the even-order SDC method (second, fourth, sixth, and eighth) at the end of the 12-day simulation. This suggests that the spatial errors are dominating for this configuration, presumably because of the discontinuity in the cosine anomaly that is unable to be exactly represented by the spectral (spatial) basis. Alternatively, as in Levy et al. (2007), a \(C^1\) continuous Gaussian anomaly is advected with the same temporal and spatial discretizations and the \(L_2\) errors are \(1.785 \times 10^{-6}\), \(1.087 \times 10^{-11}\), \(2.363 \times 10^{-11}\), and \(4.386 \times 10^{-11}\) for the second-, fourth-, sixth-, and eighth-order SDC methods, respectively. The solution error for the fourth-order SDC is marginally lower than the sixth-order and higher SDC methods, suggesting that the spatial error becomes dominant except for the second-order one when using such a small time step. In this scenario, lower-order SDC methods are slightly more accurate (but all have a very small error) due to the fact that they take fewer substeps per time step, which minimizes error accumulation. To analyze the temporal discretization errors, the simulations were run with each of the SDC orders and with a 10-min time step, giving \(L_2\) errors of \(7.141 \times 10^{-4}\), \(1.112 \times 10^{-8}\), \(1.710 \times 10^{-10}\), and \(7.384 \times 10^{-11}\), as above. As expected, the larger time step yields solutions with temporally dominated errors, so the higher-order temporal

![Fig. 4. An \(L_2\) error convergence analysis for the linear advection test case TC1 described in section 3. In this plot, 96 elements along a cube edge were used, with degree 7 (\(np = 8\) polynomials within each element for the spatial discretization; the Gaussian hump was advected with prescribed wind speed for 1 day. As expected, the higher-order SDC method reproduces the analytical answer more accurately, even with larger time-step sizes. Where spatial error dominates (for higher-order SDC, which is determined with an insensitivity to the error with time-step size), the temporal error accumulates slightly as the number of time steps is taken for the same method, or as the number of substeps is taken per time step for different methods at the same step size.](http://journals.ametsoc.org/doi/pdf/10.1175/MWR-D-12-00048.1)
discretization yields more accurate results. To isolate the
time-discretization error and represent a scenario where
the time-accumulation error is more relevant, a refined
spatial discretization of $ne = 96$ (corresponding to 55,296
total elements) and degree 7 ($np = 8$) polynomials is
used for the following tests. Figure 4 presents this effect
graphically with the order of convergence obtained for
a range of even-order SDC schemes for the Gaussian TC1
after a 1-day simulation. As expected, the higher-order
SDC methods exhibit relatively better accuracy for large
time-step sizes. Further, in the regime where the tempo-
ral error dominates, here the SDC second- and fourth-
order discretizations, the expected order of accuracy
was achieved. For the eighth-order SDC, the error is
close to the spatial error and the level of machine pre-
cision such that over time, small error values are accu-
mulating and eighth-order accuracy is hard to observe
for TC1. Note that in Fig. 4, at small enough step sizes,
where spatial error dominates, the fourth-order SDC is
more accurate than even higher-order ones, as we dis-
cussed before. Additionally, for the sixth- and eighth-
order SDC methods, the error remains rather constant
with time-step size, and even increases slightly at the
smallest time-step sizes within the same method. This is
again due to the accumulation of error associated with
a larger number of time steps contributing to an other-
wise identical simulation.

In any case, these error levels are not near a level
significant for most climate simulation configurations
although it is interesting to note what appears to be a
temporal accumulation error when spatial errors are
minimized. Thus, for this simple advection problem,
higher-order time discretizations allow the use of large
time steps to be taken relative to traditional lower-order
implicit and explicit methods and yet attain superior
accuracy.

Although the 12-day period is the benchmark test
used to establish spatial accuracy for TC1, and for all the
methods evaluated it shows low levels of temporal error,
it does not characterize the error accumulation using
lower-order methods for longer simulations more akin
to global climate simulations. For a 912-day simulation,
which corresponds to 76 rotations of the anomaly around
the sphere, the height field anomaly for simulations using
the different temporal schemes are presented in Fig. 5,
and are measurably different over the period of the
simulation. After 12 days (1 rotation), the explicit leapfrog
RK2 method and the fourth-order SDC method maintain

FIG. 5. Simulation of a Gaussian geopotential height anomaly that has been advected around the earth as per TC1 explained in the text
with (from left to right) both explicit leapfrog and fourth-order SDC time-integration schemes after 12 and 912 days. Note the ability of the
SDC method to maintain the anomaly’s shape and magnitude after a long time period. In this plot, 48 elements ($ne = 48$) along a cube edge
were used, with degree 3 ($np = 4$) polynomials within each element for the spatial discretization. The time-step size in all cases was 180 s.
the height and shape of the geopotential height anomaly quite well. However, after 912 days, only the higher-order methods have small enough errors to maintain the shape of the anomaly. The explicit scheme has diffused the anomaly both in magnitude and shape relative to the fourth-order SDC. As shown in Fig. 1, the anomaly simulated using the explicit scheme continues to lose its shape and magnitude with additional rotations.

b. Zonal flow over an isolated mountain (TC5)

The next test case implemented in HOMME solves the full nonlinear shallow water equation set given by (1) and does not have an analytical solution from which to compare, so relative statistics of these well-known problems are presented. This test, referred to as TC5 since it is the fifth within the suite, evaluates the invariant quantities of mass, energy, and potential vorticity to assess whether numerical methods of interest can maintain a balanced simulation. Over a 15-day period, it simulates the behavior of an initially prescribed geopotential field as it is advected with nonlinear flow around a sphere with a single mountain in the surface topography. The conservation of each field is evaluated. All orders of the SDC method conserve both mass and potential vorticity, and the global integral of the divergence remains zero to machine precision. For a 1-h time step and a spatial resolution of \( ne = 48 \) and \( np = 4 \), the errors for the fourth-order SDC method are \( 2.451 \times 10^{-10} \).
4.094 × 10^{-6}, and 5.381 × 10^{-7}, for the total energy, potential enstrophy, and RMS divergence, respectively, after 15 days. The geopotential and vorticity fields at day 15 are shown in Fig. 6. A time-trace error of mass, potential enstrophy, and energy is also shown in Fig. 7. The total energy is conserved within the temporal truncation error and the potential enstrophy and RMS divergence are conserved to within the spatial and temporal truncation errors. This is the expected pattern of behavior for the method and does not degrade the previously reported conservation properties for the spatial discretization of HOMME (Taylor et al. 1997).

c. Rossby–Haurwitz wave (TC6)

Another test case that addresses equation set (1), TC6, solves the nondivergent velocity field specified in Williamson et al. (1992) with a wavenumber 4. It is favorable for computing the time-step convergence behavior because the test evaluates the ability of methods to maintain the wave structure properly. The reference solution used for comparison was computed with the default leapfrog method with a 3-s time-step size and using the same spatial resolution as was used for TC5. Figure 8 displays the temporal convergence plot of the $L_2$ errors for the various orders of the SDC method after 1 day. The patterns of error convergence behavior of the SDC methods are displayed along with the previously reported second-order FI BDF2 method (Evans et al. 2010), for comparison. Although both SDC and FI BDF2 are run with a time-step size of 2400 s, the eighth-order SDC method yields an error of 7.24 × 10^{-8}, which is significantly more accurate than the FI method with the same time-step size.

As with TC1, the maintenance of the nondivergent wave field over the 14-day time period specified for TC6 does not necessarily require the accuracy provided by higher-order methods; however, as mentioned in the introduction, operational climate experiments in categories iii and iv may benefit from greater temporal fidelity. To illustrate this, Fig. 9 is a spatial convergence plot that compares the $L_2$ norm of the error for different time-integration methods and time-step sizes for a suite of spatial resolutions in HOMME. The horizontal axis shows the number of elements along a cube edge, which was refined from $ne = 4$ to 64 (for ease of analysis, the $np = 4$ was kept constant), corresponding to approximately 1000–221 000 total grid points. The explicit leapfrog method and the eighth-order SDC method used a 30-s time step ($\times$ and $\star$ marked lines, respectively) for the range of problem sizes, and both demonstrate the robust $np = 4$ spatial resolutions coarser than about 1° ($ne = 32$). For these two simulations, the explicit curve flattens at finer spatial resolutions, indicating that the temporal error is dominant in those settings, while the error in the SDC method is still predominantly spatial. The same simulations are performed using the second-order FI method and eighth-order SDC method with a 3600-s time step. As with the 30-s time-step simulations,
the spatial error decreases with $np = 4$ spatial discretization for coarse resolutions; however, the F1 error flattens out markedly with grid refinement for 2° resolution ($ne = 16$) and finer. Again, this is because the temporal error dominates the simulation. With the eighth-order SDC method, the spatial error is the dominant discretization error until about $ne = 64$, which corresponds to about $\frac{1}{2}$° resolution. This does not define what level of error is acceptable for shorter-term weather or climate simulations at these resolutions, but rather establishes at what resolution the spatial-discretization scheme is no longer the largest source of error for a given temporal scheme. When the spatial error is the dominant source of error, the choice of the time discretization should be based on efficiency. However, for highly refined spatial discretizations, low-order temporal schemes will introduce temporal errors that may lead to a waste of computer resources or unacceptably inaccurate solutions.

d. Nonlinear evolution of an unstable barotropic wave (SJ)

Although the cases above are the traditional test cases for evaluation of shallow water models on the sphere, a relevant test of a temporal-discretization scheme should explore multiple time scales. The test proposed by Galewsky et al. (2004) explores the development of barotropic instabilities within an analytically specified midlatitude jet with a perturbation applied. Gravity waves develop early on, and then the flow segues into a slower-developing set of waves that exhibit strong gradients in vorticity, the strength of which are dependent on the choice of diffusion. This test has been popular with studies that analyze local and/or adaptive mesh refinement strategies because there is localized behavior as a function of latitude (e.g., St.-Cyr et al. 2008; Ringler et al. 2011). A recent look at a higher-order time-stepping method for a discontinuous Galerkin formulation of the shallow water equations on the sphere showed the ability of the model to capture the slower-scale wave structure with larger time-step sizes (Archibald et al. 2011), although the method used was still restricted by the stability limits of the system. Presently, the SDC method was applied to this problem within the HOMME model by using the same values as the non-locally refined setup of HOMME used in St.-Cyr et al. (2008), except that instead of the Boyd–Vandeven filter, we used the more recently developed hyperviscosity
diffusion available for the explicit and Runge–Kutta integration options.

The details of the initial conditions of the case are spelled out in Galewsky et al. (2004). The benefit of the nonstability-limited method is that it is not necessary to resolve the short time-scale gravity waves to accurately capture the slower-evolving waves within the barotropic flow. Figure 10 displays the vorticity field of the test-case solution after 6 days for the explicit leapfrog, RK2, FI-BDF2, and second- (SDC2) and fourth-order (SDC4) methods (left, from top to bottom). These solutions have been configured to match earlier published results to the extent possible, using the same resolution \( (ne = 24, np = 8) \) and time-step size \( (10\text{s}, \text{in the case for explicit leapfrog}) \) because the solution was determined to be spatially converged (St.-Cyr et al. 2008, Fig. 13d). On the right in Fig. 10, the difference in the plots compared to the benchmark leapfrog case is presented from the method itself. Benchmark is defined here as a comparison, whereby an equivalent accuracy to explicit leapfrog is attained for a method using a much larger time step, not an assessment of the accuracy of one method compared to another. In the case of RK, the stability region is slightly smaller than in the leapfrog, but is fully second-order accurate. All the methods presented here produce similar solutions, but with different time-step sizes, namely 10\text{s} for leapfrog and RK2, and 1200\text{s} for FI-BDF2, SDC2, and SDC4, respectively.

e. Performance discussion

While the main purpose of this paper is to establish the SDC’s accuracy improvement applied to the simulations, also important for the relevance of the method is to investigate the performance characteristics for the test cases analyzed above. As a preliminary performance test, we display computational results based on a number of rhs evaluations. Tables 1 and 2 display the numbers of rhs evaluations for a suite of simulations of TC1 run for 1 day with the same \( np = 16 \) and \( ne = 4 \) resolutions used for section 3a. These simulations used the leapfrog (LF), RK2, and SDC methods, employing a time-step size to provide a similar level of error with about 5 and 10 digits of accuracy, respectively. For example, in Table 2, both RK2 and SDC8 give an accuracy of about \( 6^{-11} \), while the step sizes are 2 and 7200, respectively. RK2 takes 5 rhs evaluations per time step and according to the last section, SDC8 takes a maximum of \( 2 \times 4^2 \) BE solves per step, thus SDC8 would be more efficient than RK2 if it takes fewer than \( 5 \times (7200/2)/(2 \times 4^2) = 562.5 \) rhs evaluations per BE solve. The “/per solve” part of “No. of rhs” in the tables assumes the maximum, \( 2 \times p^2 \) times of BE solves per step, are taken for SDC-2p, and for
SDC8 in this case, \( \frac{8793}{(86 \times 400/7200)/(2 \times 4^2)} \approx 23 \) rhs evaluations are taken per solve. Therefore, SDC is showing some performance gain comparing to RK2 in terms of the numbers of rhs evaluations.

Tables 1 and 2 illustrate the following for this simple setup: 1) for both levels of accuracy, higher-order methods (including RK2) are more efficient than LF (with filtering) and 2) compared to RK2, SDC is less efficient for the lower accuracy threshold, but not by more than a factor of 2, showing promise that preconditioning will enable it to be a more efficient option for a more typical climate setup. It also shows that SDC is especially promising when high-accuracy computation is needed (e.g., longer paleoclimate-type runs and/or very high spatial resolution), because assuming linear error accumulation, 10 digits for a 1-day simulation will translate into 6 digits in 3 yr and 3 digits in 1000 yr.

So, there can be efficiency benefits when using the backward-Euler-based SDC method for a proof-of-principle linear advection problem, as long as the iteration count is controlled. However, for other, more complex test cases, issues such as preconditioners and careful

**TABLE 1. Simulations of TC1 with LF, RK2, SDC4, and SDC6 using the same resolution as in section 3a with time-step sizes to produce a solution with about the same level of accuracy to five significant digits.**

<table>
<thead>
<tr>
<th>Method</th>
<th>( L_2 ) error ((\times 10^{-6}))</th>
<th>Step size (s)</th>
<th>No. of RHS (total/per solve)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LF</td>
<td>2.9</td>
<td>2.25</td>
<td>38 401</td>
</tr>
<tr>
<td>RK2</td>
<td>2.7</td>
<td>450</td>
<td>961</td>
</tr>
<tr>
<td>SDC4</td>
<td>2.9</td>
<td>7200</td>
<td>1499/16</td>
</tr>
<tr>
<td>SDC6</td>
<td>3.9</td>
<td>10 800</td>
<td>2744/19</td>
</tr>
<tr>
<td>SDC8</td>
<td>1.5</td>
<td>21 600</td>
<td>4353/34</td>
</tr>
</tbody>
</table>
parallel implementation and solver optimization to maintain scalability must be considered. For example, in the nonlinear barotropic wave (SJ) test case outlined in section 3d, the nonlinearity causes the number of iterations needed for an implicit solve to be increased dramatically without preconditioning. So, the backward-Euler-based SDC approach for all orders presented here is slower than RK2 without a preconditioner. Also, for very nonlinear problems, when stability permits, using forward Euler’s method or a semi-implicit method instead of backward Euler in SDC could provide efficiency gains, but then there are still time-step size restrictions to consider.

4. Conclusions

Presently, a higher-order spectral time discretization is used in conjunction with SDC methods to improve the accuracy of the shallow water benchmark cases in HOMME for refined spatial discretizations. The second-order SDC method yields the same accuracy as the second-order FI BDF2 solver, verifying the method’s expected behavior. More accurate results are obtained with higher-order temporal discretization and up to eighth order is demonstrated. This approach provides a similar accuracy level for a significantly larger time step than the existing explicit or implicit methods. With optimized preconditioning within the BE methods used for the error correction step in SDC, we hope that efficiency gains for this method can be realized.

For climate simulations on the order of decades or longer, the temporal-error accumulation could be significant using low-order methods. As illustrated with a spatial refinement study, shallow water simulations with spatial resolution finer than $1/2^6$ ($ne = 64$) using the existing explicit leapfrog scheme are dominated by temporal error. Larger time steps can allow both efficiency gains and reduce accumulation error; however, temporal error is the largest error for simulations finer than about $2^5$ ($ne = 16$) using larger time-step sizes unless higher-order SDC methods are used. SDC methods increase the allowable time-step size and decrease the total number of time steps required, thus reducing the amount of error accumulation and, potentially, the computation time. Current work to include Krylov deferred correction (KDC) methods (Huang et al. 2007; Jia and Huang 2008) in HOMME could provide more computational efficiency while maintaining accuracy, and this work is under way.

Acknowledgments. The authors thank the three anonymous reviewers who provided very useful comments and suggestions. This work was supported by the Joint Oak Ridge National Laboratory and Sandia National Laboratories Institute for Advanced Sciences and Architecture and the Math/CS Institute EASI. This research used the resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract DE-AC05-00OR22725. The submitted manuscript has been authored by a contractor of the U.S. government. Accordingly, the U.S. government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. government purposes.

APPENDIX

SDC Algorithm

Algorithm 1: A summary of the SDC algorithm for one time step

1) Compute $\psi^{(i)}$ using the backward Euler method. ($p$ implicit solves performed.)

2) $i = 0.$

3) $\varepsilon_0 = 0.$ (Assuming the initial value is always correct.)

4) Repeat.

5) $\varphi^{(i)} = S\psi^{(i)} + \varphi_0.$ (Compute the integral by spectral integration.)

6) For $m = 1$ to $p$, do

7) \[ \varepsilon_m = G[\varphi_m^{(i)} + S_c(m)] - \psi^{(i)} \]

8) \[ S_c(m) = \sum_{j=1}^{m} \varepsilon_j (\tau_j - \tau_{j-1}) \] for $\varepsilon_m$ (one implicit solve performed),

9) end for

10) $\psi^{(i+1)} = \psi^{(i)} + \varepsilon^{(i)},$

11) until ($i = 2p$ or $||\varepsilon^{(i+1)}|| < \text{tol}$) (tol is a prescribed tolerance),

12) return $\varphi^{(i)} = S\psi^{(i)} + \varphi_0.$

APPENDIX

SDC Algorithm

Algorithm 1: A summary of the SDC algorithm for one time step

1) Compute $\psi^{(i)}$ using the backward Euler method. ($p$ implicit solves performed.)

2) $i = 0.$

3) $\varepsilon_0 = 0.$ (Assuming the initial value is always correct.)

4) Repeat.

5) $\varphi^{(i)} = S\psi^{(i)} + \varphi_0.$ (Compute the integral by spectral integration.)

6) For $m = 1$ to $p$, do

7) \[ \varepsilon_m = G[\varphi_m^{(i)} + S_c(m)] - \psi^{(i)} \]

8) \[ S_c(m) = \sum_{j=1}^{m} \varepsilon_j (\tau_j - \tau_{j-1}) \] for $\varepsilon_m$ (one implicit solve performed),

9) end for

10) $\psi^{(i+1)} = \psi^{(i)} + \varepsilon^{(i)},$

11) until ($i = 2p$ or $||\varepsilon^{(i+1)}|| < \text{tol}$) (tol is a prescribed tolerance),

12) return $\varphi^{(i)} = S\psi^{(i)} + \varphi_0.$
REFERENCES


