CFL Optimized Forward–Backward Runge–Kutta Schemes for the Shallow-Water Equations

JEREMY R. LILLY,[a,b] DARREN ENGWIRDA,[c] GIACOMO CAPODAGLIO,[b] ROBERT L. HIGDON,[a]
AND MARK R. PETERSEN[b]

[a] *Department of Mathematics, Oregon State University, Corvallis, Oregon*
[b] *Computational Physics and Methods Group, Los Alamos National Laboratory, Los Alamos, New Mexico*
[c] *Fluid Dynamics and Solid Mechanics Group, Los Alamos National Laboratory, Los Alamos, New Mexico*

(Manuscript received 7 June 2023, in final form 29 August 2023, accepted 25 September 2023)

ABSTRACT: We present the formulation and optimization of a Runge–Kutta-type time-stepping scheme for solving the shallow-water equations, aimed at substantially increasing the effective allowable time step over that of comparable methods. This scheme, called FB-RK(3,2), uses weighted forward–backward averaging of thickness data to advance the momentum equation. The weights for this averaging are chosen with an optimization process that employs a von Neumann–type analysis, ensuring that the weights maximize the admittable Courant number. Through a simplified local truncation error analysis and numerical experiments, we show that the method is at least second-order in time for any choice of weights and exhibits low dispersion and dissipation errors for well-resolved waves. Further, we show that an optimized FB-RK(3,2) can take time steps up to 2.8 times as large as a popular three-stage, third-order strong stability-preserving Runge–Kutta method in a quasi-linear test case. In fully nonlinear shallow-water test cases relevant to oceanic and atmospheric flows, FB-RK(3,2) outperforms SSPRK3 in admittable time step by factors roughly between 1.6 and 2.2, making the scheme approximately twice as computationally efficient with little to no effect on solution quality.

SIGNIFICANCE STATEMENT: The purpose of this work is to develop and optimize time-stepping schemes for models relevant to oceanic and atmospheric flows. Specifically, for the shallow-water equations we optimize for schemes that can take time steps as large as possible while retaining solution quality. We find that our optimized schemes can take time steps between 1.6 and 2.2 times larger than schemes that cost the same number of floating point operations, translating directly to a corresponding speedup. Our ultimate goal is to use these schemes in climate-scale simulations.

KEYWORDS: Shallow-water equations; Numerical analysis/modeling; Model evaluation/performance

1. Introduction

Large-scale climate models, such as the U.S. Department of Energy's Energy Exascale Earth System Model (E3SM) (Golaz et al. 2022), grow in complexity every year as new numerical techniques are developed and as high-performance computing (HPC) systems become more powerful. Complex, climate-scale simulations can take days or weeks or more of real time to complete; it is vital that these run times are controlled without making undue sacrifices to overall model quality. In the present work, we develop a way to formulate and optimize a class of time-stepping schemes for the shallow-water equations (SWEs). Our ultimate goal is to improve the efficiency of simulations at the climate scale, but here we focus on the shallow-water equations as a starting point. Although this work is directed to models of ocean and atmosphere circulation, the ideas and methods developed here may generalize to other coupled systems.

The central issue here is the following. The Courant–Friedrichs–Lewy (CFL) condition is a well-known necessary condition for the stability of a given model under an explicit time-stepping scheme. In essence, the CFL condition places a restriction on the size of the model's time step, where this restriction depends on the model problem itself and the chosen time and space discretizations. Formally, it is necessary for stability that

$$\nu = c\frac{\Delta t}{\Delta x} \le \nu^{\max}, \tag{1}$$

where c is a speed such as a gravity wave speed. We refer to ν as the Courant number and $\nu^{\max}$ as the maximal Courant number for the given situation. In this work, we develop time-stepping schemes optimized to have maximal $\nu^{\max}$. We do this by starting with a three-stage second-order Runge–Kutta method from Wicker and Skamarock (2002), presently used in the Model for Prediction Across Scales-Atmosphere (MPAS-A) (Skamarock et al. 2012). This scheme is augmented by adding forward–backward averaging of the thickness tendencies used to update the fluid velocity. Resulting schemes have large $\nu^{\max}$ and good dispersion and dissipation relations, and they compare well against other schemes in both solution accuracy and effective maximal Courant number.

Denotes content that is immediately available upon publication as open access.

Corresponding author: Jeremy R. Lilly, lillyj@oregonstate.edu

DOI: 10.1175/MWR-D-23-0113.1

The SWEs serve as a standard test problem for algorithms and methods designed for applications to ocean and atmosphere modeling; in both cases, the model equations used for climate-scale simulations are generalizations of the SWEs, in which additional vertical tendencies and forcing terms are included, and variable density is allowed. In recent years, strong stability preserving Runge–Kutta (SSPRK) methods (Gottlieb and Shu 1998) for solving the SWEs have become quite popular; they have the property that at each time level, a spatial norm of the solution is less than or equal to the norm of the solution at the preceding level. These methods are used to solve the SWEs in a variety of numerical paradigms and applications, such as with finite element spatial discretizations (Azerad et al. 2017; Dawson et al. 2013; Fu 2022), finite volume discretizations (Bao et al. 2014; Capodaglio and Petersen 2022; Hoang et al. 2019; Katta et al. 2015; Lilly et al. 2023; Roullet and Gaillard 2022; Ullrich et al. 2010), and layered models that use a time splitting of barotropic and baroclinic motions (Kärnä et al. 2018; Lan et al. 2022). Despite this widespread use, strong stability preserving (SSP) time-integration methods may not be the best choice for most large-scale geophysical SWE applications. The SSP property is useful in systems where strong shocks and discontinuities can develop and propagate, but such dynamics are rare in climate-scale oceanic and atmospheric flows, in which barotropic evolution is typically characterized by Rossby- and Kelvin-type wave modes. Numerous contemporary large-scale ocean and atmosphere models employ time-stepping methods that do not have the SSP property, motivating an investigation of its optimality for this class of flow. For example, to solve the barotropic equations in split barotropic/baroclinic systems, the Model for Prediction Across Scales-Ocean (MPAS-O) (Ringler et al. 2013; Petersen et al. 2019) and Model for Prediction Across Scales-Atmosphere (MPAS-A) (Skamarock et al. 2012) use a two-stage predictor-corrector method and a three-stage second-order Runge–Kutta method, respectively. The Community Atmosphere Model (CAM) (Collins et al. 2004), the Parallel Ocean Program (POP) (Kerbyson and Jones 2005), the Nucleus for European Modeling of the Ocean (NEMO) (Madec et al. 2017), and the Hybrid Coordinate Ocean Model (HYCOM) (Chassignet et al. 2007) all use variations of semi-implicit leapfrog time integration schemes for their barotropic systems. The Regional Ocean Modeling System (ROMS) (Shchepetkin and McWilliams 2005) uses a multistep forward–backward Adams–Bashforth/Adams–Moulton method.

With the potential performance benefits of the ability to take larger time steps in mind, we turn our attention toward methods that are optimal in this sense. Taking advantage of the structure of the SWEs, we employ a strategy similar to that used by Mesinger (1977), wherein the thickness equation is advanced in time with a simple forward Euler step, then the momentum equation is advanced with an explicit backward Euler step using the recently obtained data for the thickness. The method, referred to by Mesinger as the forward–backward method, was more computationally efficient than popular leapfrog methods of the time. The idea of advancing the variables of a system of ODE separately, using the most recently computed data from previously advanced variables, has been employed by the developers of ROMS. Specifically, Shchepetkin and McWilliams (2005, 2009) present an extension of a two-stage second-order Runge–Kutta scheme that, within each RK stage, first advances the thickness variable, then uses a weighted average of the most recently computed data with previously computed data to advance the momentum equation in a shallow-water system. The coefficients of this scheme are selected to optimize the allowable time-step length for a one-dimensional gravity wave system based on a von Neumann–type stability analysis. In this work, we develop a new, forward–backward weighted three-stage Runge–Kutta method with favorable CFL limits using a similar approach. The optimal weights for our forward–backward scheme are obtained by von Neumann–type analysis on the two-dimensional system, including a linear Coriolis term and linearization of the advection operators about a nonzero mean state. Using a numerical optimization algorithm, we obtain a scheme that is very efficient in terms of its maximal Courant number, as well as its dissipation and dispersion characteristics.

The paper is structured as follows. We begin by introducing our new time-stepping scheme, referred to as FB-RK(3,2), for a general system of ODEs. Then, we describe the process by which we obtain optimal weights for the FB averaging within each RK stage. This process includes the derivation of a von Neumann formulation of the linearized SWEs with a nonzero mean flow and the formulation of an optimization problem. Next, we present the results of this optimization and give a brief discussion of the dispersion and dissipation errors of the optimal FB-RK(3,2) schemes. Finally, we perform a number of numerical experiments that demonstrate the computational efficiency and efficacy of the optimal FB-RK(3,2) schemes as compared to the three-stage third-order SSPRK scheme (SSPRK3) from Gottlieb and Shu (1998). These experiments include an investigation of CFL performance across five shallow-water test cases, convergence tests, and comparisons of solution quality. We show that an optimized FB-RK(3,2) scheme can take time steps up to 2.2 times as large as SSPRK3 in nonlinear test cases while retaining solution quality, and because both schemes consist of three RK stages, this translates directly to a corresponding increase in efficiency.

2. CFL optimization of FB-RK(3,2)

Here, we introduce a time-stepping scheme for solving the shallow-water equations (SWEs) that uses weighted forward–backward averaging in the layer thickness data to advance the momentum equation. We then use a von Neumann analysis and a numerical optimization scheme to tune the forward–backward weights so that the scheme has a large Courant number when applied to the linearized SWEs.

a. *FB-RK(3,2)*

The time-stepping scheme presented here is an extension of the three-stage, second-order Runge–Kutta time-stepping scheme RK(3,2) from Wicker and Skamarock (2002). RK(3,2) was originally presented as a third-order method and referred to as RK3, but it is shown by Purser (2007) that the method is third-order only on linear problems, reducing to second-order on nonlinear problems. When applied to the SWEs, our extension of RK(3,2) allows us to use the most recently obtained data for the layer thickness to update the momentum data within each Runge–Kutta stage of the method. This is done by taking a weighted average of layer thickness data at the old time level t^n and the most recent RK stage, then applying this to the momentum equation. For the remainder of the manuscript, we will call our extension of RK(3,2) that uses forward–backward averaging FB-RK(3,2), as it is a three-stage, second-order scheme.

Consider a general system of ODEs in independent variables $u = u(t)$ and $h = h(t)$ of the following form:

$$\frac{du}{dt} = \Phi(u, h),$$
$$\frac{dh}{dt} = \Psi(u, h)\,, \tag{2}$$

where t is the time coordinate. Let $u^n \approx u(t^n)$ and $h^n \approx h(t^n)$ $h^n \approx h(t^n)$ be the numerical approximations to u and h at time $t = t^n$. Let Δt be a time step such that $t^{n+1} = t^n + \Delta t$. Also, let $t^{n+1/m} = t^n + (\Delta t/m)$ for any positive integer m. Then, FB-RK(3,2) is given by

$$\begin{aligned}\overline{h}^{n+1/3} &= h^n + \frac{\Delta t}{3}\Psi(u^n, h^n),\\ \overline{u}^{n+1/3} &= u^n + \frac{\Delta t}{3}\Phi(u^n, h^*),\\ h^* &= \beta_1\overline{h}^{n+1/3} + (1-\beta_1)h^n,\end{aligned} \tag{3a}$$

$$\begin{aligned}\overline{h}^{n+1/2} &= h^n + \frac{\Delta t}{2}\Psi(\overline{u}^{n+1/3}, \overline{h}^{n+1/3}),\\ \overline{u}^{n+1/2} &= u^n + \frac{\Delta t}{2}\Phi(\overline{u}^{n+1/3}, h^{**}),\\ h^{**} &= \beta_2\overline{h}^{n+1/2} + (1-\beta_2)h^n,\end{aligned} \tag{3b}$$

$$\begin{aligned}h^{n+1} &= h^n + \Delta t\Psi(\overline{u}^{n+1/2}, \overline{h}^{n+1/2}),\\ u^{n+1} &= u^n + \Delta t\Phi(\overline{u}^{n+1/2}, h^{***}),\\ h^{***} &= \beta_3 h^{n+1} + (1-2\beta_3)\overline{h}^{n+1/2} + \beta_3 h^n\,\cdot\end{aligned} \tag{3c}$$

The weights β_1, β_2, and β_3 are called the forward–backward (FB) weights. The best choice for the FB weights β_i is a primary concern of this work.

The choice to calculate data at time $t^{n+1/3}$ and $t^{n+1/2}$ in the first and second stages comes from the original RK(3,2) scheme; the treatment of the thickness variable h in (3) shows the formulation of RK(3,2). FB-RK(3,2) can be approximately reduced to RK(3,2) by taking $\beta_1 = 0$, $\beta_2 = 2/3$, and $\beta_3 = 0$. This produces an approximation to RK(3,2) as, while the first and third stages are exact, the FB averaged $h^{**} = (2/3)\overline{h}^{n+1/2} + (1/3)h^n$ data in the second stage is an approximation to $\overline{h}^{n+1/3}$. The FB averages are chosen in this way so that in each stage, data are symmetrically distributed in time as preliminary experiments showed that this provided the best CFL performance. In the first two RK stages, the FB averaging is done on data from the beginning and the end of the current time interval, i.e., in the first stage, we compute values for h and $\mathbf{u}$ at time $t^{n+1/3}$, so the FB averaging is done with data at time t^n and time $t^{n+1/3}$. In the third RK stage, we use data from the beginning, middle, and end of the current time interval.

In the context of this work, h would be the fluid layer thickness and u the fluid velocity. We have chosen to use an FB average of thickness to advance the velocity variable. However, one could reasonably define a scheme similar to that given by (3) by taking FB averages of u to advance h instead. The subsequent analysis would remain largely the same, but the optimal choices for the FB weights could change substantially, along with the performance of the scheme as a whole. What happens in this case is an open question.

As stated above, the original RK(3,2) scheme is $\mathcal{O}[(\Delta t)^2]$. For any choice of the FB weights, FB-RK(3,2) remains $\mathcal{O}[(\Delta t)^2]$ while greatly increasing the maximum allowable time step for most problems. A generalized local truncation error analysis is outside the scope of this paper, so we show this holds for a particular problem of interest in section a of the appendix.

b. *Linearized shallow-water equations*

A primary concern of this work is the application of FB-RK(3,2) to the nonlinear SWEs on a rotating sphere. Application of a von Neumann–type analysis requires a linear set of PDEs, and we consider the two-dimensional system with linear Coriolis and linearized advection operators to ensure the time-step optimization is conducted on a system that is a close approximation to the fully nonlinear equations. The linearized SWEs centered about a nonzero mean flow are given by

$$\frac{\partial\tilde{\mathbf{u}}}{\partial t} + f(\tilde{\mathbf{U}} + \tilde{\mathbf{u}})^\perp + \tilde{\zeta}\tilde{\mathbf{U}}^\perp = -\tilde{\mathbf{U}}\cdot\nabla\tilde{\mathbf{u}} - g\nabla\tilde{h},$$
$$\frac{\partial\tilde{h}}{\partial t} + H(\nabla\cdot\tilde{\mathbf{u}}) + \nabla\cdot(\tilde{h}\tilde{\mathbf{U}}) = 0\,, \tag{4}$$

where $\tilde{\mathbf{u}} = [\tilde{u}(x, y, t), \tilde{v}(x, y, t)]$ is a perturbation of the horizontal fluid velocity, x and y are the spatial coordinates, t is the time coordinate, $f = 2\Omega\sin\varphi$ is the Coriolis parameter, where Ω is the angular velocity of the rotating sphere and φ denotes latitude, $\tilde{\mathbf{U}} = (\tilde{U}, \tilde{V})$ is the constant horizontal mean fluid velocity, $\tilde{\mathbf{u}}^\perp = (-\tilde{v}, \tilde{u})$, $\tilde{\zeta} = (\partial\tilde{v}/\partial x) - (\partial\tilde{u}/\partial y)$ is the vertical component of vorticity $\nabla\times\tilde{\mathbf{u}}$, g is the gravitational constant, $\tilde{h} = \tilde{h}(x, y, t)$ is a perturbation of the layer thickness, and H is the (constant in space) fluid layer thickness when at rest. Note that while $\tilde{\mathbf{u}}$ and $\tilde{h}$ depend on x and y throughout this work, we will often suppress this dependence in the notation. The full derivation of these equations is given in section b of the appendix.

We can write these in terms of dimensionless quantities by writing $c = \sqrt{gH}$, $\mathbf{u} = (u, v) = (\tilde{\mathbf{u}}/c)$, $\mathbf{U} = (\tilde{\mathbf{U}}/c)$, and $\eta = \tilde{h}/H$. Then, the dimensionless, linearized SWEs centered about a nonzero mean flow are given by

$$\frac{\partial\mathbf{u}}{\partial t} + f(\mathbf{U} + \mathbf{u})^\perp + c\zeta\mathbf{U}^\perp = -c\mathbf{U}\cdot\nabla\mathbf{u} - c\nabla\eta,$$
$$\frac{\partial\eta}{\partial t} + c(\nabla\cdot\mathbf{u}) + c[\nabla\cdot(\eta\mathbf{U})] = 0. \tag{5}$$

c. von Neumann analysis

In the analysis that follows, we assume that the system (5) is discretized on a rectangular, staggered Arakawa C-grid, where η is computed at cell centers, u is computed at the left and right edges of cells, and v is computed at the top and bottom edges of cells (Arakawa and Lamb 1977). For brevity, we skip some details of calculations relating to the spatial discretization in the main text and instead provide them in section c of the appendix.

With the goal of using a von Neumann–type analysis to obtain an optimal choice of FB weights, we apply a Fourier transform in x and y to the linearized SWEs (5). This is equivalent to seeking solutions of the following form:

$$\begin{aligned} u(x, y, t) &= \hat{u}(k, \ell, t)e^{ikx+i\ell y}, \\ v(x, y, t) &= \hat{v}(k, \ell, t)e^{ikx+i\ell y}, \\ \eta(x, y, t) &= \hat{\eta}(k, \ell, t)e^{ikx+i\ell y}, \end{aligned}$$

where k is the wavenumber with respect to x, and ℓ is the wavenumber with respect to y. In doing this, spatial derivatives are now all of the form $(\partial/\partial x)(e^{ikx+i\ell y})$ or $(\partial/\partial y)(e^{ikx+i\ell y})$. Assuming a rectangular C-grid, using centered differences these spatial derivatives become

$$\frac{\partial}{\partial x}(e^{ikx+i\ell y}) = \frac{i2\sin\left(k\dfrac{\Delta x}{2}\right)}{\Delta x}e^{ikx+i\ell y}, \tag{6}$$

$$\frac{\partial}{\partial y}(e^{ikx+i\ell y}) = \frac{i2\sin\left(\ell\dfrac{\Delta y}{2}\right)}{\Delta y}e^{ikx+i\ell y}. \tag{7}$$

To compute the Coriolis terms, we need the values of u at v points and the values of v at u points; call these u_v and v_u, respectively. On a rectangular C-grid, we achieve this with unweighted four-point averages of the nearest data, which gives

$$u_v(x, y, t) = \cos\left(k\frac{\Delta x}{2}\right)\cos\left(\ell\frac{\Delta y}{2}\right)\hat{u}(k, \ell, t)e^{ikx+i\ell y}, \tag{8}$$

$$v_u(x, y, t) = \cos\left(k\frac{\Delta x}{2}\right)\cos\left(\ell\frac{\Delta y}{2}\right)\hat{v}(k, \ell, t)e^{ikx+i\ell y}. \tag{9}$$

The details of these calculations are given in section c of the appendix. Set $K = 2\sin[k(\Delta x/2)]$, $L = 2\sin[\ell(\Delta y/2)]$, and $\psi = f\cos[k(\Delta x/2)]\cos[\ell(\Delta y/2)]$. Then, accounting for the Fourier transform and the C-grid spatial discretization, (5) becomes

$$\begin{aligned} \frac{\partial\hat{u}}{\partial t} &= fV + \psi\hat{v} - \left(iU\frac{cK}{\Delta x} + iV\frac{cL}{\Delta y}\right)\hat{u} - i\frac{cK}{\Delta x}\hat{\eta}, \\ \frac{\partial\hat{v}}{\partial t} &= -fU - \psi\hat{u} - \left(iU\frac{cK}{\Delta x} + iV\frac{cL}{\Delta y}\right)\hat{v} - i\frac{cL}{\Delta y}\hat{\eta}, \\ \frac{\partial\hat{\eta}}{\partial t} &= -\left(i\frac{cK}{\Delta x}\hat{u} + i\frac{cL}{\Delta y}\hat{v}\right) - \left(iU\frac{cK}{\Delta x} + iV\frac{cL}{\Delta y}\right)\hat{\eta}\,\cdot \end{aligned} \tag{10}$$

Let $\nu_x = c(\Delta t/\Delta x)$ and $\nu_y = c(\Delta t/\Delta y)$ be the Courant numbers with respect to x and y, respectively. Also, set $\varphi = \Delta t\psi$. Then, we apply FB-RK(3,2) to (10):

$$\begin{aligned} \overline{\hat{\eta}}^{n+1/3} &= \hat{\eta}^n - \frac{1}{3}\left[iK\nu_x\hat{u}^n + iL\nu_y\hat{v}^n + (iUK\nu_x + iVL\nu_y)\hat{\eta}^n\right], \\ \overline{\hat{u}}^{n+1/3} &= \hat{u}^n + \frac{1}{3}\Big[\Delta tfV + \varphi\hat{v}^n - (iUK\nu_x + iVL\nu_y)\hat{u}^n \\ &\quad - iK\nu_x\hat{\eta}^*\Big], \\ \overline{\hat{v}}^{n+1/3} &= \hat{v}^n + \frac{1}{3}\Big[-\Delta tfU - \varphi\hat{u}^n - (iUK\nu_x + iVL\nu_y)\hat{v}^n \\ &\quad - iK\nu_x\hat{\eta}^*\Big], \\ \hat{\eta}^* &= \beta_1\overline{\hat{\eta}}^{n+1/3} + (1 - \beta_1)\hat{\eta}^n, \end{aligned} \tag{11a}$$

$$\begin{aligned} \overline{\hat{\eta}}^{n+1/2} &= \hat{\eta}^n - \frac{1}{2}\Big[iK\nu_x\overline{\hat{u}}^{n+1/3} + iL\nu_y\overline{\hat{v}}^{n+1/3} + (iUK\nu_x \\ &\quad + iVL\nu_y)\overline{\hat{\eta}}^{n+1/3}\Big], \\ \overline{\hat{u}}^{n+1/2} &= \hat{u}^n + \frac{1}{2}\Big[\Delta tfV + \varphi\overline{\hat{v}}^{n+1/3} - (iUK\nu_x + iVL\nu_y)\overline{\hat{u}}^{n+1/3} \\ &\quad - iK\nu_x\hat{\eta}^{**}\Big], \\ \overline{\hat{v}}^{n+1/2} &= \hat{v}^n + \frac{1}{2}\Big[-\Delta tfU - \varphi\overline{\hat{u}}^{n+1/3} - (iUK\nu_x + iVL\nu_y)\overline{\hat{v}}^{n+1/3} \\ &\quad - iK\nu_x\hat{\eta}^{**}\Big], \\ \hat{\eta}^{**} &= \beta_2\overline{\hat{\eta}}^{n+1/2} + (1 - \beta_2)\hat{\eta}^n, \end{aligned} \tag{11b}$$

$$\begin{aligned} \hat{\eta}^{n+1} &= \hat{\eta}^n - \Big[iK\nu_x\overline{\hat{u}}^{n+1/2} + iL\nu_y\overline{\hat{v}}^{n+1/2} + (iUK\nu_x \\ &\quad + iVL\nu_y)\overline{\hat{\eta}}^{n+1/2}\Big], \\ \hat{u}^{n+1} &= \hat{u}^n + \Big[\Delta tfV + \varphi\overline{\hat{v}}^{n+1/2} - (iUK\nu_x + iVL\nu_y)\overline{\hat{u}}^{n+1/2} \\ &\quad - iK\nu_x\hat{\eta}^{***}\Big], \\ \hat{v}^{n+1} &= \hat{v}^n + \Big[-\Delta tfU - \varphi\overline{\hat{u}}^{n+1/2} - (iUK\nu_x + iVL\nu_y)\overline{\hat{v}}^{n+1/2} \\ &\quad - iK\nu_x\hat{\eta}^{***}\Big], \\ \hat{\eta}^{***} &= \beta_3\hat{\eta}^{n+1} + (1 - 2\beta_3)\overline{\hat{\eta}}^{n+1/2} + \beta_3\hat{\eta}^n. \end{aligned} \tag{11c}$$

Note that (11) forms a linear system in $\hat{u}^n$, $\hat{v}^n$, and $\hat{\eta}^n$, that is, there is a matrix $\mathbf{G}$ and column vector $\mathbf{b}$ such that

$$\hat{\mathbf{w}}^{n+1} = \mathbf{G}\hat{\mathbf{w}}^n + \mathbf{b}, \tag{12}$$

where $\hat{\mathbf{w}}^n = (\hat{u}^n, \hat{v}^n, \hat{\eta}^n)^{\mathrm{T}}$. It follows from (12) that

$$\hat{\mathbf{w}}^n = \mathbf{G}^n\hat{\mathbf{w}}^0 + (\mathbf{I} - \mathbf{G}^n)(\mathbf{I} - \mathbf{G})^{-1}\mathbf{b}, \tag{13}$$

where $\mathbf{I}$ is the identity matrix. Note that in the above, the superscript on $\hat{\mathbf{w}}$ is an index on the time step, whereas the superscript on $\mathbf{G}$ is an exponent. The matrix $\mathbf{G}$ is called the amplification matrix; properties of von Neumann analysis tell us that a time-stepping method is stable if the eigenvalues of its amplification matrix reside within in the unit circle centered at the origin in the complex plane (LeVeque 2007). This property is fundamental to the analysis in section 2d.

d. CFL optimization

With FB-RK(3,2) applied to the von Neumann formulation of the linearized SWEs (11) and the corresponding amplification matrix $\mathbf{G}$ from (12), we can more clearly state the goal of the analysis presented in this section. We are interested in selecting the FB weights β_1, β_2, and β_3 in FB-RK(3,2) so that the Courant numbers $\nu_x = c(\Delta t/\Delta x)$ and $\nu_y = c(\Delta t/\Delta y)$ can be taken as large as possible while retaining stability of the method.

The eigenvalues of the amplification matrix $\mathbf{G}$ depend on the Courant numbers ν_x and ν_y and on β_1, β_2, and β_3. To simplify the analysis, we will assume that $\nu_x = \nu_y = \nu$ is the Courant number (this is equivalent to assuming that the spatial discretization is on a grid consisting of sqaure cells). Then, we can write $\mathbf{G} = \mathbf{G}(\nu, \beta_1, \beta_2, \beta_3)$. For a given choice of β_1, β_2, and β_3, let $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$ be the largest value of the Courant number for which FB-RK(3,2) is stable, i.e., largest value of the Courant number for which the eigenvalues of the amplification matrix $\mathbf{G}$ reside within the unit circle. Symbolically, $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$ is the largest number such that

$$|\lambda_{\max}[\mathbf{G}(\nu, \beta_1, \beta_2, \beta_3)]| \le 1, \text{ for all } \nu \in \left(0, \nu^{\max}_{(\beta_1,\beta_2,\beta_3)}\right], \tag{14}$$

where $\lambda_{\max}(\mathbf{G})$ is the largest eigenvalue, in absolute value, of the matrix $\mathbf{G}$.

While the matrix $\mathbf{G}$ is too complex to analyze directly, we can use numerical optimization tools to find optimal FB weights. That is, we can formulate this problem explicitly as an optimization problem with an appropriate cost function and constraints, then use numerical optimization algorithms to search the space of admittable FB weights. We can maximize $\nu^{\max}$ by minimizing an appropriate cost function:

$$C = C(\beta_1, \beta_2, \beta_3), \tag{15}$$

subject to the constraint that the eigenvalues of $\mathbf{G}$ satisfy (14).

The particular numerical optimization scheme we employ here is the simplicial homology global optimization (SHGO) algorithm developed by Endres et al. (2018), implemented in the Python programming language via the open-source Python library SciPy (Virtanen et al. 2020). SHGO is suitable for so-called black-box optimization problems wherein the cost function can be nonsmooth and discontinuous. Further, SHGO is guaranteed to converge to a global optimum as the number of iterations increases. As such, SHGO is an appropriate choice for our purposes, where the eigenvalues of the matrix $\mathbf{G}$ are sensitive to small changes in the FB weights.

The matrix $\mathbf{G}$ also depends on additional parameters Δtf, $k\Delta x$, and $\ell\Delta y$ as can be seen in (11). For the purposes of the numerical optimizations we perform here, we set $\Delta tf = 10^{-2}$ since taking $f = 10^{-4}$ represents the value of the Coriolis parameter at midlatitudes and we expect a FB-RK(3,2) to be capable of a time step on the order of 10^2 in the nonlinear SWE test cases described in section 3. Preliminary tests showed that the choice of value for Δtf had little to no effect on the optimizations in practice. We take $k\Delta x = \ell\Delta y = \pi$, which corresponds to the case of gridscale waves. This choice is informed by the fact that instability often comes from gridscale processes. That is, we perform the optimization on gridscale waves, which represents a worst-case with the hope that this provides the best overall stability.

1) THE CHOICE OF COST FUNCTIONS

Given a particular choice of β_1, β_2, and β_3, we can easily estimate the value of $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$ to arbitrary precision. Our goal is to maximize this parameter, so we seek to minimize the cost function:

$$C_1(\beta_1, \beta_2, \beta_3) = \frac{1}{\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}}. \tag{16}$$

In the special case where the mean flow $\mathbf{U} = (U, V)$ is zero, we additionally consider a second choice of cost function. When $\mathbf{U} = 0$ the column vector $\mathbf{b}$ in (12) is also zero, so (13) simplifies to

$$\hat{\mathbf{w}}^n = \mathbf{G}^n\hat{\mathbf{w}}^0. \tag{17}$$

In this case, it is easy to show that the numerical amplification matrix $\mathbf{G}$ approximates $\tilde{\mathbf{G}} := e^{\mathbf{A}\Delta t}$, where

$$\mathbf{A} = \begin{pmatrix} 0 & f & -ick \\ -f & 0 & -ic\ell \\ -ick & -ic\ell & 0 \end{pmatrix}. \tag{18}$$

The matrix $\tilde{\mathbf{G}}$ is the analytical counterpart to the numerical amplification matrix $\mathbf{G}$ that arises from solving a spatially continuous version of (10). In short, if we assume that our time-stepping method, which is determined by the matrix $\mathbf{G}$, produces the exact solution at time $t = \Delta t$, then $\mathbf{G} = \tilde{\mathbf{G}}$. We give the details of this in appendix, section d.

Since we wish for $\mathbf{G}$ to approximate $\tilde{\mathbf{G}}$ for a given choice of FB weights, we aim to minimize a second choice of cost function:

$$C_2(\beta_1, \beta_2, \beta_3) = \frac{1}{\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}} + \int_0^{\pi/6} \|\tilde{\mathbf{G}}(\nu) - \mathbf{G}(\nu, \beta_1, \beta_2, \beta_3)\| d\nu, \tag{19}$$

where $\|\cdot\|$ is the Frobenius norm:

$$\|\mathbf{A}\| = \sqrt{\textstyle\sum_{i,j}(a_{ij})^2}. \tag{20}$$

This choice of cost function seeks to maximize $\nu^{\max}$ while also producing a scheme that has favorable dissipation and dispersion behavior. The choice of norm in the integral term of C_2 was chosen as it is quick to compute during the optimization process. There is a certain freedom in the choice of limits of integration for the integral term in C_2; this term requires that the difference between $\mathbf{G}$ and $\tilde{\mathbf{G}}$ is minimal as for small values of ν, which correspond to well-resolved waves. In essence, while C_1 only considers the CFL performance of the resulting method, C_2 does this while also considering that $\mathbf{G}$ should accurately approximate its analytical counterpart $\tilde{\mathbf{G}}$.

In summary, we consider two choices of cost function given by C_1 in (16) and C_2 in (19). The function C_1 is valid in any case, while C_2 applies only when the mean flow $\mathbf{U}$ is taken to be zero.

2) Optimization results

In the context of the dimensionless Eq. (5), the parameter $|\mathbf{U}|$ is called the Froude number. We perform the optimization described above five total times, considering different values of the Froude number $|\mathbf{U}|$ that appear in oceanic and atmospheric flows.

As noted above, the SHGO algorithm used here is guaranteed to converge to a globally optimal solution if run for enough iterations. As our problem is relatively complex and computing resources finite, it is more likely in practice that the method will convergence to a local extremum. For example, consider the first two rows of Table 1 and notice that the value of $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$ given by optimizing C_2 is greater than that given by optimizing C_1. If these were globally optimal solutions, we would have that $\nu^{\max}_{(0.500,0.500,0.344)} \geq \nu^{\max}_{(0.516,0.532,0.331)}$ since C_1 consists only of a term that asks for a maximal $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$, while C_2 is the sum of the same term with a nonnegative term that asks for $\mathbf{G}$ to approximate $\tilde{\mathbf{G}}$. Despite the problem being sufficiently complex that globally optimal solutions are difficult to obtain, the results from section 3 show that the choices of FB weights from Table 1 are extremely efficient in practice.

A similar Runge–Kutta-based method using forward–backward averaging in the momentum equation is developed by Shchepetkin and McWilliams (2005) (from here on denoted ShMc). In ShMc, the efficiency of their optimized FB method is reported in terms of the parameter $\alpha := ck\Delta t$, by considering the maximal value $\alpha^{\max}$ for which the method is stable. Translating this into the notation used here, we obtain

$$\begin{aligned}\alpha &= ck\Delta t \\ &= \nu k\Delta x. \end{aligned} \tag{21}$$

For their optimal RK2-based FB method, ShMc reports that $(\nu k\Delta x)_{\max} \approx 2.141$; since their scheme uses two RK stages, this translates to what we refer to as an effective CFL number of $2.141/2 \approx 1.071$. Here, we have taken $k\Delta x = \pi$ and have $\nu_{\max} = 1.804$ in the best case, so $(\nu k\Delta x)_{\max} = 1.804\pi \approx 5.667$. FB-RK(3,2) uses three RK stages, so we obtain an effective CFL number of $5.667/3 \approx 1.889$, an increase of approximately 1.764 times over ShMc's RK2 FB scheme. Note that ShMc's $(\nu k\Delta x)_{\max}$ comes from their FB scheme as applied to a one-dimensional wave system analogous to (22), whereas ours is obtained from the two-dimensional linearized SWEs. The comparison between these two values is not one-to-one, but do serve as a general comparison of the efficiency of both methods.

TABLE 1. Results of optimizing FB-RK(3,2) for the given cost functions, rounded to three decimal places. Note that the reported values of $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$ are the maximal Courant numbers in the context of a problem consisting only of gridscale waves, i.e., the problem for which FB-RK(3,2) was optimized.

$\lvert\mathbf{U}\rvert$	Cost function	β_1, β_2, β_3	$\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$
0	C_1	0.500, 0.500, 0.344	1.767
0	C_2	0.516, 0.532, 0.331	1.804
0.05	C_1	0.531, 0.531, 0.313	1.319
0.15	C_1	0.359, 0.578, 0.234	1.025
0.25	C_1	0.656, 0.938, 0.188	0.853

To visualize FB-RK(3,2)'s treatment of gravity waves within the SWEs, we can apply the scheme to a simple wave system and analyze the resulting dissipation and dispersion errors (Fig. 1). Applying a Fourier transform in space to the 1D linearized SWEs (A1) gives

$$\begin{aligned}\frac{\partial\hat{\eta}}{\partial t} &= -ick\hat{u} \\ \frac{\partial\hat{u}}{\partial t} &= -ick\hat{\eta}\,\cdot \end{aligned} \tag{22}$$

Choose a time step Δt and let $\tilde{K} = k\Delta x$ and $\nu = c(\Delta t/\Delta x)$. The FB-RK(3,2) scheme for this system is then given by

$$\begin{aligned}\overline{\eta}^{n+1/3} &= \eta^n - \frac{i\tilde{K}\nu}{3}u^n, \\ \overline{u}^{n+1/3} &= u^n - \frac{i\tilde{K}\nu}{3}[\beta_1\overline{\eta}^{n+1/3} + (1-\beta_1)\eta^n], \end{aligned} \tag{23a}$$

$$\begin{aligned}\overline{\eta}^{n+1/2} &= \eta^n - \frac{i\tilde{K}\nu}{2}\overline{u}^{n+1/3}, \\ \overline{u}^{n+1/2} &= u^n - \frac{i\tilde{K}\nu}{2}[\beta_2\overline{\eta}^{n+1/2} + (1-\beta_2)\eta^n], \end{aligned} \tag{23b}$$

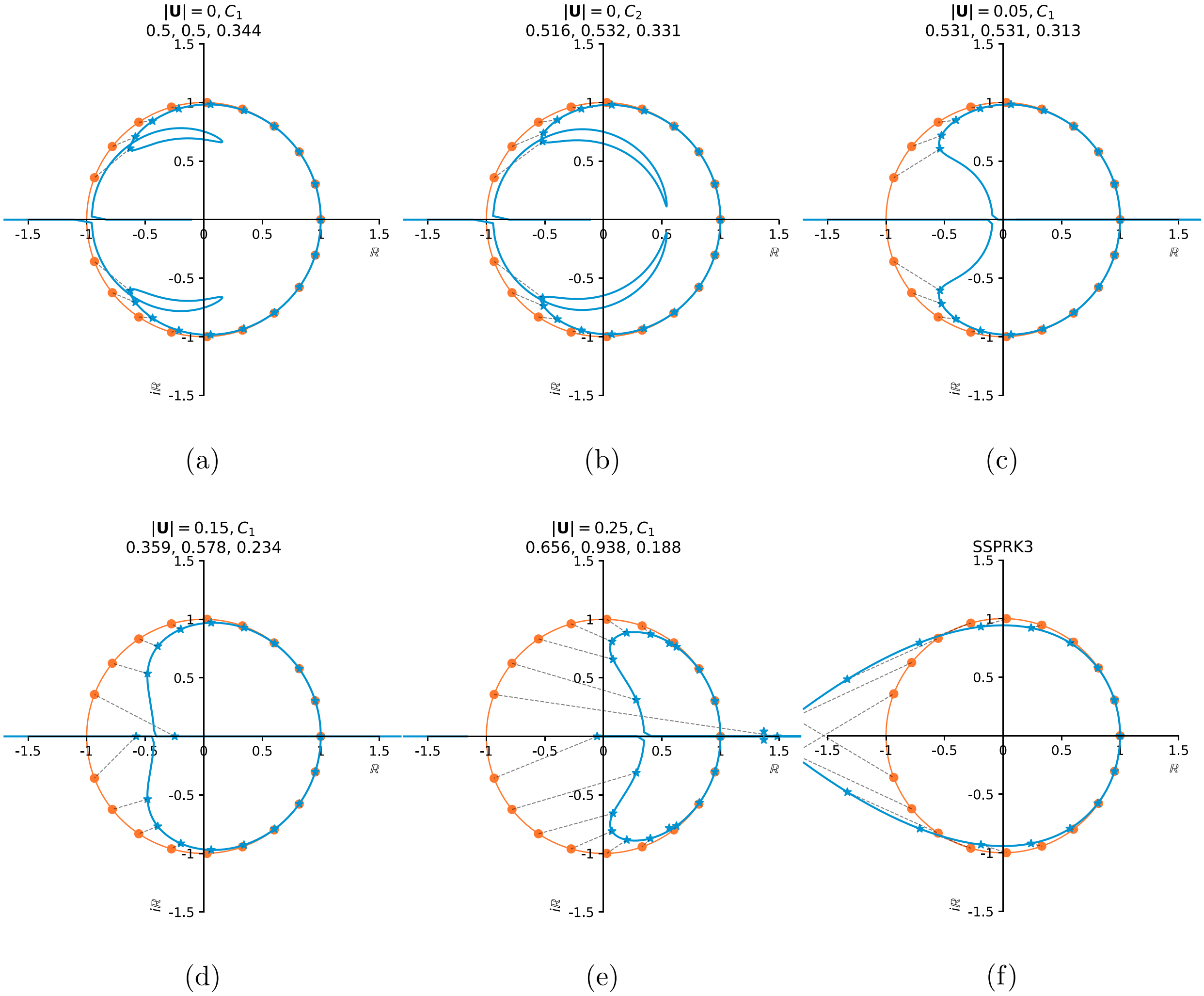
$$\begin{aligned}\eta^{n+1} &= \zeta^n - i\tilde{K}\nu\overline{u}^{n+1/2}, \\ u^{n+1} &= u^n - i\tilde{K}\nu[\beta_3\eta^{n+1} + (1-2\beta_3)\overline{\eta}^{n+1/2} + \beta_3\eta^n]. \end{aligned} \tag{23c}$$

Equation (23) defines a system of linear equations, which in turn defines a numerical amplification matrix $\mathbf{G}_0$ similar to that in (12). One can show that the general solution to the system (22) is

$$\begin{bmatrix}\eta(t)\\ u(t)\end{bmatrix} = \exp\left[t\begin{pmatrix}0 & -ick\\ -ick & 0\end{pmatrix}\right]\begin{bmatrix}\eta(0)\\ u(0)\end{bmatrix}, \tag{24}$$

and that eigenvalues of the matrix exponential above are $\lambda = e^{\pm ickt}$. At time $t = \Delta t$, these eigenvalues are $\lambda = e^{\pm i\tilde{K}\nu}$.

Note that although we have used the Courant number $\nu = c(\Delta t/\Delta x)$ in this discussion in an effort to use notation analogous to the rest of this section, the system considered here is not discretized in space; we have that $\tilde{K}\nu = k\Delta x \times c(\Delta t/\Delta x) = ck\Delta t$. Values of $\tilde{K}\nu$ close to zero correspond to well-resolved waves, while values approaching π correspond to temporal gridscale waves. Dissipation error can be read as the distance a numerical eigenvalue is from the unit circle, resulting in the corresponding wave being damped out of the system. Dispersion error can be read as the difference in the angles made with the positive real axis between a numerically determined eigenvalue with its analytical counterpart, resulting in an error in the phase speed of the corresponding wave. For all choices of FB weights in Table 1, dissipation and dispersion errors for well resolved waves are very low (Fig. 1). As waves become less well resolved, each of the five



(a) (b) (c)

(d) (e) (f)

FIG. 1. An illustration of dissipation and dispersion errors for FB-RK(3,2) with each choice of FB weights from Table 1 and SSPRK3. Eigenvalues $\lambda = e^{\pm i\tilde{K}\nu}$ of the matrix in the exact solution (24) are given by orange dots. Drawing a ray from the origin to one of these dots and measuring the angle that ray makes with the positive real axis gives the corresponding value of $\tilde{K}\nu$. In other words, $\tilde{K}\nu$ is the argument of the corresponding complex number. Eigenvalues of the numerical amplification matrix $\mathbf{G}_0$ as $\tilde{K}\nu = k\Delta x\nu$ increases are given by the blue curve and the blue stars. Dashed lines connect exact eigenvalues to numerically determined counterparts, which gives a measure of dissipation and dispersion errors. Values of $\tilde{K}\nu$ close to zero correspond to well-resolved waves, while values approaching π correspond to temporal gridscale waves.

schemes exhibits different behavior. In particular, note the behavior of the eigenvalues in Fig. 1c; well resolved waves are handled well as expected, and as waves move toward grid scale, the dissipation errors become very high, i.e., the blue curve becomes closer to the origin. This means that waves that are not well-resolved are quickly damped out of the system, which may explain the strong performance of FB-RK(3,2) with the corresponding choices of FB weights. Contrast this with the behavior of SSPRK3 in Fig. 1f; as we move toward the grid scale, rather than damping out the potentially problematic motions, the eigenvalues simply leave the unit circle, signaling that the method has become unstable.

Note that the paradigm in which the plots of Fig. 1 exist is different than the paradigm for which the FB weights were optimized. That is, the matrix $\mathbf{G}_0$ comes from applying FB-RK(3,2) to a one-dimensional gravity wave system (22), whereas the FB weights presented in Table 1 are optimized for the linearized SWEs (10). We present the plots in Fig. 1 to show that the schemes optimized for stability in the linearized SWEs treat well-resolved gravity waves correctly in the best case, where the equations are not discretized in space. This also allows for a quick visual comparison between other methods presented with similar plots in the literature; in particular with the methods presented in ShMc.

3. Numerical experiments

Here, we present a number of numerical experiments that demonstrate the efficiency and accuracy of the CFL-optimized FB-RK(3,2) schemes presented in section 2. These experiments

TABLE 2. CFL performance of FB-RK(3,2) with optimal FB weights from Table 1 compared to SSPRK3 and RK(3,2) on various test cases. The recorded values for Δt represent the largest time step for which the given test case could run without becoming unstable for the given duration. Time steps were determined experimentally by running the model for the given duration, increasing the time step until the model became unstable. For practical reasons, time steps were found as multiples of 5.

			$\lvert\mathbf{U}\rvert = 0$, C_1	$\lvert\mathbf{U}\rvert = 0$, C_2	$\lvert\mathbf{U}\rvert = 0.05$, C_1	$\lvert\mathbf{U}\rvert = 0.15$, C_1	$\lvert\mathbf{U}\rvert = 0.25$, C_1
			0.500	0.516	0.531	0.359	0.656
			0.500	0.532	0.531	0.578	0.938
Test case	SSPRK3	RK(3,2)	0.344	0.331	0.313	0.234	0.188
QLW 7 days							
Max Δt (s)	515	515	1445	1435	1115	960	820
$\Delta t/\Delta t_{\text{SSPRK3}}$		1.00	2.81	2.79	2.17	1.86	1.59
BUJ 6 days							
Max Δt (s)	110	115	195	195	195	190	185
$\Delta t/\Delta t_{\text{SSPRK3}}$		1.05	1.77	1.77	1.77	1.73	1.68
WTC2 5 days							
Max Δt (s)	220	230	355	360	370	410	360
$\Delta t/\Delta t_{\text{SSPRK3}}$		1.05	1.61	1.64	1.68	1.86	1.64
WTC4 5 days							
Max Δt (s)	110	115	230	235	240	210	175
$\Delta t/\Delta t_{\text{SSPRK3}}$		1.05	2.09	2.14	2.18	1.91	1.59
WTC5 15 days							
Max Δt (s)	145	145	270	275	310	275	235
$\Delta t/\Delta t_{\text{SSPRK3}}$		1.00	1.86	1.9	2.14	1.90	1.62

compare SSPRK3 and RK(3,2) to FB-RK(3,2) in terms of time-stepping efficiency, temporal convergence, and solution accuracy. We consider a total of five different nonlinear shallow-water test cases described below. The fully nonlinear SWEs are given by (A14) in section b of the appendix.

Quasi-linear gravity wave (QLW): A simple gravity wave traveling from the north pole to the south, then returning to the north, a process which takes approximately seven days of simulated time. This is the case of an external wave mode, also called the Lamb mode in the context of atmospheric modeling. The momentum advection terms in the momentum equation of (A14) are turned off, making the momentum equation purely linear. The thickness equation (A14) is formally nonlinear, but because the fluid depth is sufficiently large compared to any perturbation in the thickness, this behaves as though it is linear. Therefore, we refer to this test case as quasi-linear. The fluid velocity is initialized to zero, while the thickness is initialized as a Gaussian bell curve centered at the north pole of the form $\exp[-100(x-\pi)^2 - 100y^2] + 500$.

Barotropically unstable jet (BUJ): Originally formulated by Galewsky et al. (2004), this case consists of a strong zonal flow at midlatitude with a small perturbation that causes the barotropic jet to become unstable. This flow is driven by strongly nonlinear momentum advection tendencies, physical processes that are of particular relevance to weather prediction and climate modeling.

Williamson test case 2 (WTC2): A steady state solution to the nonlinear SWEs, one of the standard SWE test cases developed by Williamson et al. (1992). The case consists of a geostrophically balanced zonal flow that is constant for all time.

Williamson test case 4 (WTC4): A forced nonlinear flow consisting of a translating low pressure center superimposed on a jet stream, one of the standard Williamson et al. (1992) SWE test cases.

Williamson test case 5 (WTC5): A zonal flow over an isolated mountain, one of the standard Williamson et al. (1992) SWE test cases. Here, the initial flow is exactly as in WTC2, but the seafloor topography is nonuniform, consisting of an isolated mountain at midlatitude. This perturbation alters the geostrophic balance of the flow, triggering the development of advection-driven instabilities.

All test cases take place on a rotating aquaplanet with angular velocity $\Omega = 7.292 \times 10^{-5}$ s^{-1} and a radius of 6371.22 km. The seafloor topography is uniform unless otherwise specified. Each test case is implemented using a TRiSK spatial discretization (Ringler et al. 2010), which is a finite volume-type spatial discretization made for unstructured, variable-resolution polygonal grids. Here, we slightly alter the standard TRiSK discretization by using an upwind potential vorticity flux to remove gridscale noise in the evolution of vorticity. Icosahedral meshes were used throughout, with quasi-uniform 60-km resolution employed in all test cases. Meshes were optimized according to the spherical centroidal Voronoi tessellation (SCVT) metric using the JIGSAW library (Engwirda 2017), ensuring the TRiSK discretization achieves approximately second-order accuracy in the L_2 norm. Balanced initial conditions for the BUJ and WTC test cases are generated following the procedure described in section e of the appendix. The code used to run all the experiments described in this section is open source and available on GitHub and Zenodo; see the data availability statement at the end of this document.

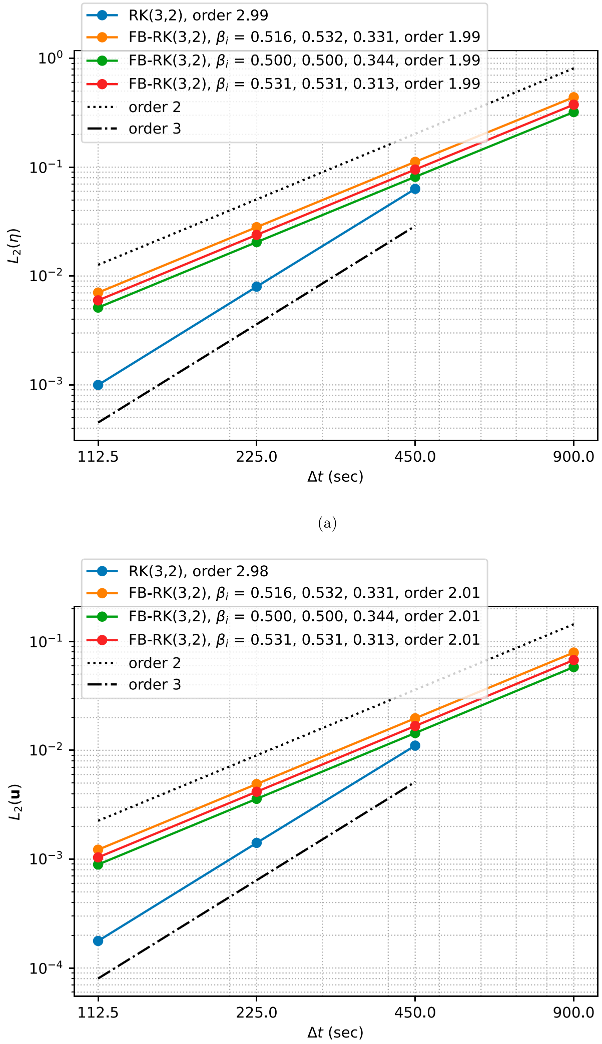
a. CFL performance

On all of the test cases described above, the optimized FB-RK(3,2) schemes from section 2d admit time steps between 1.59 and 2.81 times as large as SSPRK3 or RK(3,2). This demonstrates the ability for the FB averaging employed by FB-RK(3,2) and the subsequent optimization of the FB weights to greatly increase the stability range of an existing method across multiple problems. SSPRK3 is used as a baseline comparison here since it admits the smallest time steps for each case out of the schemes considered here, and it also consists of three Runge–Kutta stages, costing the same number of floating-point operations as RK(3,2) and FB-RK(3,2).

The largest increases in the admittable time step are in the QLW test case, with the largest increase at 2.81 times in the best case (Table 2). For practical reasons, we used a nonlinear SWE model to run the above test cases, but QLW was intentionally designed so that the model equations, while still formally nonlinear, approximate the behavior of the linear SWEs. That is, QLW provides a test case as close as possible to the paradigm for which FB-RK(3,2) was optimized. As noted, the comparison is not one-to-one. FB-RK(3,2) was optimized for the linearized SWEs on a square C-grid, while QLW uses quasi-linear SWEs and a hexagonal grid with a TRiSK discretization. Nonetheless, the improvement in CFL performance is remarkable and clearly demonstrates the efficacy of the optimization performed in section 2d.

The remaining four test cases are fully nonlinear, covering a range of standard shallow-water test cases relevant to atmospheric and oceanic flows. In these, FB-RK(3,2) admits time steps between 1.59 and 2.18 times larger than SSPRK3. As SSPRK3 consists of the same number of Runge–Kutta stages as FB-RK(3,2), this directly translates to an increase of overall efficiency of between 1.59 and 2.18. The increases in admittable time step in these nonlinear cases are slightly less than those in the QLW case in part because these problems include fully nonlinear motions that cannot be accounted for in the optimization process used here; because our optimization relies on properties of von Nuemann stability analysis, we are limited to optimizing on a linear set of equations. However, Table 1 suggests that optimizing for the linearized SWEs provides substantial benefits even for the nonlinear SWEs. Indeed, as noted above, our analysis and the subsequent optimization was performed under a number of simplifications versus the paradigm in which these numerical experiments were performed. Nonetheless, our results show that the optimized FB-RK(3,2) schemes are robust when applied to fully nonlinear problems with a full TRiSK discretization. This provides evidence that our schemes are at least partially agnostic to the details of the spatial discretization.

Additionally, one can observe that the optimized FB-RK(3,2) schemes are sensitive to the Froude number; as this increases, the corresponding values of $\nu^{\max}_{(\beta_1,\beta_2,\beta_3)}$ decrease (Table 1). As the Froude number increases, the flow becomes dominated by advection rather than gravity wave propagation. When the stability is dominated by fast waves, FB-RK(3,2) works best as evidenced by its performance in the QLW case. As the wave and advective phase speeds come closer together, the speed-up from FB is reduced, due to the flow taking on more advective nature as can be seen in the four fully nonlinear test cases.



(a)

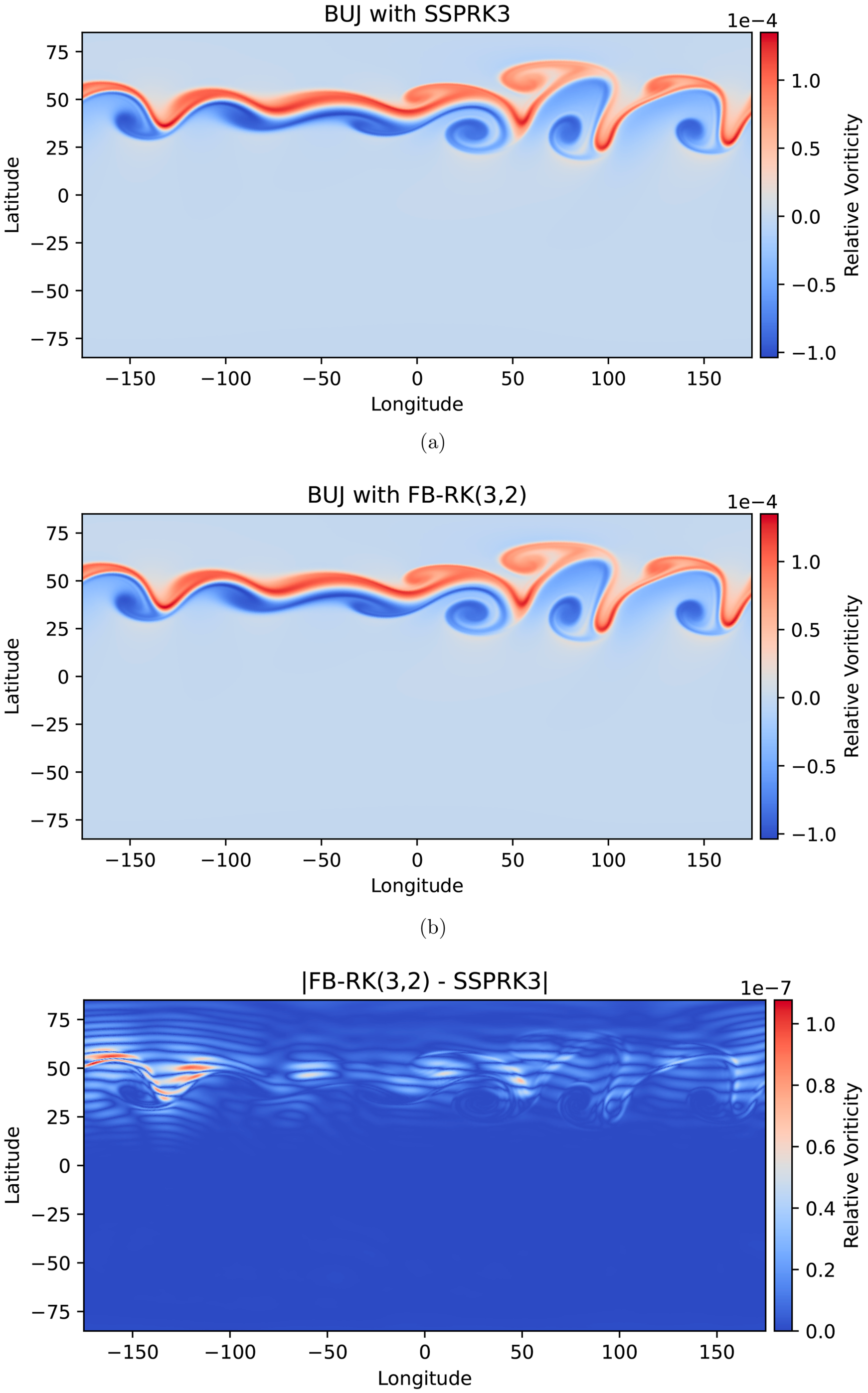
(b)

FIG. 2. Temporal convergence for FB-RK(3,2) for a selection of FB weights. Convergence tests were performed using the QLW case, run for 7 simulated days. The L_2 errors are computed against a reference solution generated by the classical four-stage, fourth-order Runge–Kutta method (RK4) with a time step of 10 s.

Turning our attention to the four nonlinear test cases, we see that FB-RK(3,2) with the FB weights obtained by optimizing C_1 with $|\mathbf{U}| = 0.05$ admits the largest time step in each case except WTC2 where it instead admits the second largest time step. This points to $(\beta_1, \beta_2, \beta_3) = (0.531, 0.531, 0.313)$ as a robust option for a broad span of problems. However, it is almost certain that the optimal choice of coefficient will be problem dependent.

b. Temporal convergence

In section a of the appendix, we perform a simplified local truncation error analysis that shows that FB-RK(3,2) is second-order in time for any choice of FB weights on the one dimensional



(a)

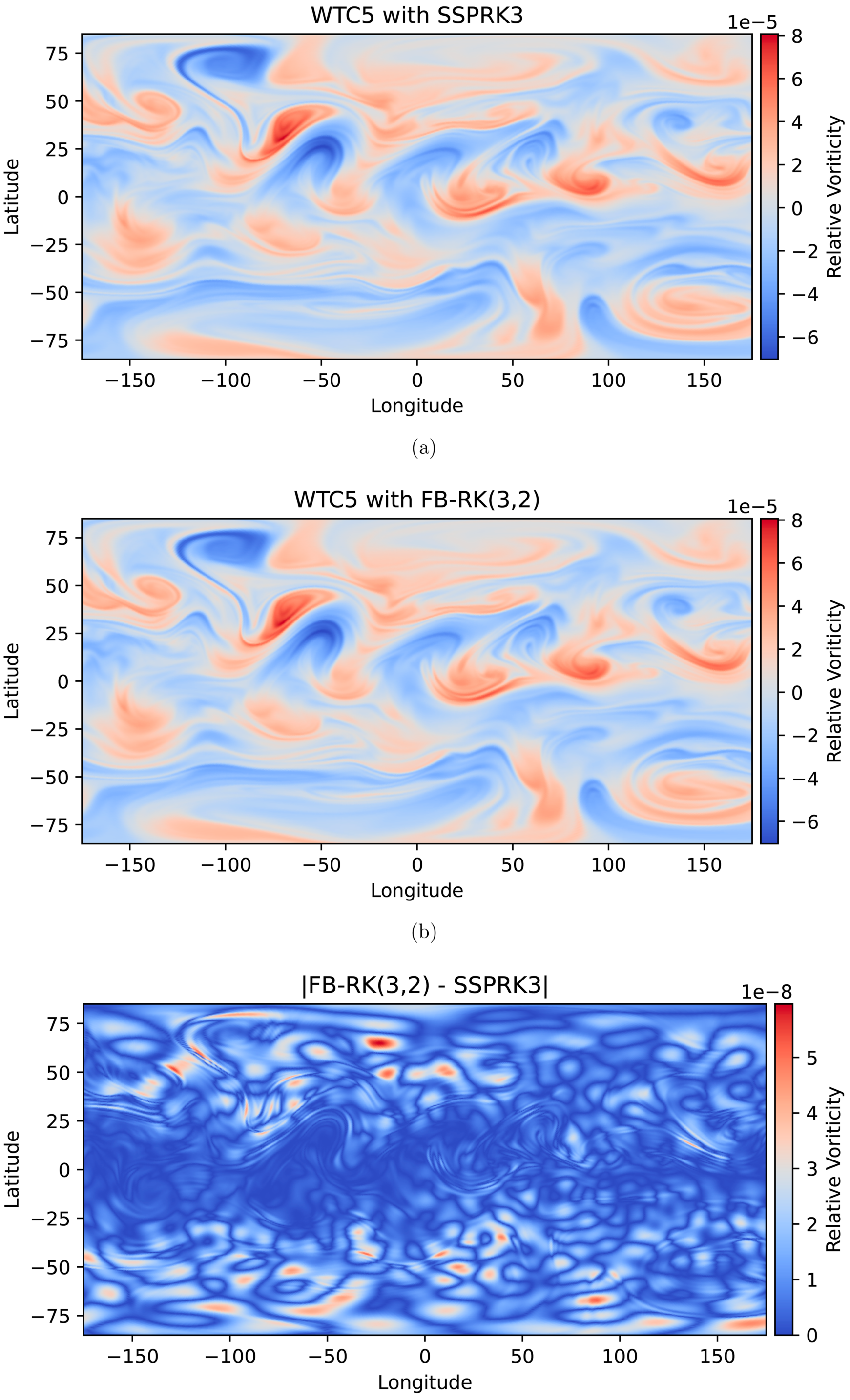
(b)

(c)

FIG. 3. Relative vorticity produced by (a) SSPRK3 and (b) FB-RK(3,2) with (β_1, β_2, β_3) = (0.531, 0.531, 0.313) on the BUJ test case after 6 days of simulated time. SSPRK3 uses a time step of 108 s and FB-RK(3,2) uses a time step of 192 s. (c) The absolute difference between the two solutions. The plots themselves are lat/lon projections of the sphere.

linearized SWEs. Here, we support the assertion that FB-RK(3,2) is second-order in time with numerical experiments.

Each tested variation of FB-RK(3,2) is second-order in time (Fig. 2). Differences in accuracy between the variations of FB-RK(3,2) are negligible, showing that the choice of optimized FB weights has little effect on accuracy in this case. RK(3,2), the scheme on which FB-RK(3,2) is based, is third-order as expected as this is a linear problem (Purser 2007).



(a)

(b)

(c)

FIG. 4. Relative vorticity produced by (a) SSPRK3 and (b) FB-RK(3,2) with (β_1, β_2, β_3) = (0.531, 0.531, 0.313) on the WTC5 test case after 50 days of simulated time. SSPRK3 uses a time step of 135 s and FB-RK(3,2) uses a time step of 288 s. (c) The absolute difference between the two solutions. The plots themselves are lat/lon projections of the sphere.

c. *Solution quality*

We evaluate the solution quality of an optimal FB-RK(3,2) to that of SSPRK3 in two test cases of interest, BUJ and WTC5, comparing the relative vorticity produced by FB-RK(3,2) with $(\beta_1, \beta_2, \beta_3) = (0.531, 0.531, 0.313)$ to that produced by SSPRK3. The strong stability preserving property of SSPRK3 makes it a good choice for a point of comparison; if FB-RK(3,2) can produce a solution of comparable quality, we can be confident that it is a good choice for the SWEs and related problems.

In both cases, shown in Figs. 3 and 4, FB-RK(3,2) produces solutions of comparable quality to SSPRK3 even with a significantly increased time step. In BUJ, the two schemes produce relative velocities that differ by at most 10^{-7}. In WTC5, this difference is on the order of 10^{-8}. We also ran both test cases using RK(3,2) which produced plots, which we omit, that are visually identical to those given here.

FB-RK(3,2) uses a time step 1.78 and 2.13 times larger that that used by SSPRK3 on BUJ and WTC5, respectively. Despite taking about half the wall-clock time and computational resources, FB-RK(3,2) produces vorticities that are qualitatively equivalent to SSPRK3. As noted in section 3a, though FB-RK(3,2) is lower order in time than SSPRK3, the produced solutions are qualitatively equivalent; likely partially due to lower-order spatial errors which dominate. In operational configurations of climate-scale models, such spatial errors are generally dominant, so one can use a more economical time-stepping scheme like FB-RK(3,2) without sacrificing model quality.

4. Conclusions

With the goal of producing computationally efficient time-stepping schemes for the SWEs, we have presented a class of three-stage second-order Runge–Kutta schemes that use forward–backward averaging to advance the momentum equation. The FB weights in these FB-RK(3,2) schemes were then optimized using a von Neumann–type analysis on the shallow-water equations linearized about a nonzero mean flow. Even though it is impossible to perform a similar analysis and optimization on the fully nonlinear SWEs, the results of our linearized analysis clearly result in significant increases in efficiency when solving nonlinear test problems.

Our optimized FB-RK(3,2) schemes compare favorably in terms of effective CFL number to an existing forward–backward scheme in Shchepetkin and McWilliams (2005). In numerical experiments, the schemes are up to 2.8 times more computationally efficient than the RK(3,2) scheme from which they were derived and a popular three-stage third-order strong stability preserving Runge–Kutta scheme (Table 2). Despite being significantly more computationally economical than other schemes, FB-RK(3,2) produces solutions that are of comparable quality to other schemes while using a time step approximately twice as long (Figs. 3 and 4). FB-RK(3,2) schemes are generally applicable to geophysical fluid flows in the ocean and atmosphere, with a range of Froude numbers, as evidenced by their performance across a number of relevant test cases. Their use provides an economical pathway to increasing the computational efficiency of models of any scope, but particularly models at the climate scale wherein time-to-solution can be restrictive.

Moving forward, we are interested in adapting FB-RK(3,2) to a local time-stepping (LTS) scheme similar to that used in Dawson et al. (2013) or Hoang et al. (2019), thereby combining the benefits of the CFL performance of FB-RK(3,2) with the obvious performance implications of LTS schemes. Another potentially valuable application of FB-RK(3,2) would be its use as a barotropic solver within a split-explicit, layered model wherein barotropic and baroclinic motions are separated and solved by different methods. The barotropic equations are essentially the SWEs and our work here clearly demonstrates that FB-RK(3,2) is a good choice for solving such a system. As the number of MPI processes increases, the communication costs with in the barotropic solver begin to dominate within a split ocean model. Taking a longer barotropic time step would help to assuage some of these costs.

Acknowledgments. JRL was supported by the U.S. Department of Energy (DOE), Office of Science, Office of Workforce Development for Teachers and Scientists, Office of Science Graduate Student Research (SCGSR) program. The SCGSR program is administered by the Oak Ridge Institute for Science and Education (ORISE) for the DOE. ORISE is managed by ORAU under Contract DE-SC0014664. JRL was also supported by the National Science Foundation (NSF) Mathematical Sciences Graduate Internship (MSGI). GC, MRP, and DE were supported by the Earth System Model Development program area of the U.S. DOE, Office of Science, Office of Biological and Environmental Research as part of the multi-program, collaborative Integrated Coastal Modeling (ICoM) project. The authors would also like to thank Pedro S. Peixoto and two anonymous reviewers for their insightful comments and suggestions.

Data availability statement. The Python code that implements the optimization problem formulated in section 2 is publicly available on both GitHub and Zenodo (Lilly 2023b). GitHub: https://github.com/jeremy-lilly/rk-fb-optimization/tree/32d2480f5ffcb66f4bea62653cb451b4373b6ace; Zenodo: https://doi.org/10.5281/zenodo.7958458. Similarly, the source code for the shallow-water model used here can be found on GitHub and Zenodo (Engwrida and Lilly 2023). GitHub: https://github.com/dengwirda/swe-python/tree/e7a7beb3ddcc9e27a17ce9eac2de196d77b18db8; Zenodo: https://doi.org/10.5281/zenodo.7958483. The data generated for this paper are also publicly available on Zenodo. This includes the model output used to generate relative vorticity plots, model output to generate convergence plots, and log files from CFL performance experiments (Lilly 2023a). Zenodo: https://doi.org/10.5281/zenodo.7958518

APPENDIX

Supplementary Mathematical Details

a. *Local truncation error analysis*

What follows is a simplified local truncation error analysis of FB-RK(3,2) that shows that the method is at least

$\mathcal{O}[(\Delta t)^2]$ on a particular problem for any choice of the FB weights β_i. To ease notation and simplify symbolic calculations, consider a simplified version of the linearized SWEs (5) with zero mean flow, in one spatial dimension, and where $f = 0$:

$$\begin{aligned}\frac{\partial u}{\partial t} &= -c\frac{\partial \eta}{\partial x},\\ \frac{\partial \eta}{\partial t} &= -c\frac{\partial u}{\partial x}.\end{aligned} \tag{A1}$$

Assume that u is sufficiently smooth that derivatives up to sixth order in both t and x exist, and that η is sufficiently smooth that derivatives up to fifth order in both t and x exist. Further, assume that both u and η are sufficiently smooth that $(\partial/\partial t)(\partial^k/\partial x^k)u = (\partial^k/\partial x^k)(\partial/\partial t)u$ and $(\partial/\partial t)(\partial^k/\partial x^k)\eta = (\partial^k/\partial x^k)(\partial/\partial t)\eta$ for $k = 1, 2$. This second assumption gives us

$$\frac{\partial^k u}{\partial t^k} = \begin{cases} -c^k\dfrac{\partial^k \eta}{\partial x^k}, & \text{if } k \text{ is odd} \\ c^k\dfrac{\partial^k u}{\partial x^k}, & \text{if } k \text{ is even} \end{cases}, \tag{A2}$$

$$\frac{\partial^k \eta}{\partial t^k} = \begin{cases} -c^k\dfrac{\partial^k u}{\partial x^k}, & \text{if } k \text{ is odd} \\ c^k\dfrac{\partial^k \eta}{\partial x^k}, & \text{if } k \text{ is even} \end{cases}, \tag{A3}$$

for $k = 1, 2$. Also, note that in the analysis that follows, the dependence of u and η on x is suppressed to simplify notation.

Now, by applying FB-RK(3,2) to (A1), we can write

$$\begin{aligned}\frac{u^{n+1} - u^n}{\Delta t} &= -c\frac{\partial \eta^n}{\partial x} + \frac{\Delta t}{2}c^2\frac{\partial^2 u^n}{\partial x^2} - \frac{(\Delta t)^2}{6}c^3(\beta_3 + 1)\frac{\partial^3 \eta^n}{\partial x^3}\\ &\quad + \frac{(\Delta t)^3}{36}c^4(-4\beta_1\beta_3 + 2\beta_1 + 9\beta_2\beta_3)\frac{\partial^4 u^n}{\partial x^4}\\ &\quad - \frac{(\Delta t)^4}{12}c^5\beta_2\beta_3\frac{\partial^5 \eta^n}{\partial x^5} + \frac{(\Delta t)^5}{36}c^6\beta_1\beta_2\beta_3\frac{\partial^6 u^n}{\partial x^6}\\ &= U(u^n, \eta^n),\end{aligned} \tag{A4}$$

$$\begin{aligned}\frac{\eta^{n+1} - \eta^n}{\Delta t} &= -c\frac{\partial u^n}{\partial x} + \frac{\Delta t}{2}c^2\frac{\partial^2 \eta^n}{\partial x^2} - \frac{(\Delta t)^2}{4}c^3\beta_2\frac{\partial^3 u^n}{\partial x^3}\\ &\quad + \frac{(\Delta t)^3}{12}c^4\beta_2\frac{\partial^4 \eta^n}{\partial x^4} - \frac{(\Delta t)^4}{36}c^5\beta_1\beta_2\frac{\partial^5 u^n}{\partial x^5}\\ &= H(u^n, \eta^n).\end{aligned} \tag{A5}$$

Notice that the right-hand side of (A4) and (A5) are equations in u^n and η^n; call them $U(u^n, \eta^n)$ and $H(u^n, \eta^n)$, respectively.

The local truncation error for a given time-stepping scheme can be thought of as the error introduced by said scheme during one time step, assuming exact data at the beginning of the time step. For this problem, the local truncation errors in u and η are defined to be the quantities τ_u^n and τ_η^n such that

$$\frac{u(t^{n+1}) - u(t^n)}{\Delta t} = U[u(t^n), \eta(t^n)] + \tau_u^n, \tag{A6}$$

$$\frac{\eta(t^{n+1}) - \eta(t^n)}{\Delta t} = H[u(t^n), \eta(t^n)] + \tau_\eta^n. \tag{A7}$$

To calculate τ_u^n and τ_η^n, rearrange (A6) and (A7), then expand $u(t^{n+1})$ and $\eta(t^{n+1})$ in t as Taylor polynomials centered at $t = t^n$ to get

$$\begin{aligned}\Delta t\,\tau_u^n &= u(t^{n+1}) - u(t^n) - \Delta t U[u(t^n), \eta(t^n)]\\ &= \left\{\sum_{k=0}^{6}\frac{(\Delta t)^k}{k!}\frac{\partial^k u}{\partial t^k}(t^n) + \mathcal{O}[(\Delta t)^7]\right\}\\ &\quad - u(t^n) - \Delta t\,U[u(t^n), \eta(t^n)],\end{aligned} \tag{A8}$$

$$\begin{aligned}\Delta t\,\tau_\eta^n &= \eta(t^{n+1}) - \eta(t^n) - \Delta t\,H[u(t^n), \eta(t^n)]\\ &= \left\{\sum_{k=0}^{5}\frac{(\Delta t)^k}{k!}\frac{\partial^k \eta}{\partial t^k}(t^n) + \mathcal{O}[(\Delta t)^6]\right\}\\ &\quad - \eta(t^n) - \Delta t H[u(t^n), \eta(t^n)].\end{aligned} \tag{A9}$$

Now, simplify (A8) and (A9). Starting with (A8), we get

$$\begin{aligned}\Delta t\,\tau_u^n &= \left\{\sum_{k=0}^{3}\frac{(\Delta t)^k}{k!}\frac{\partial^k u}{\partial t^k}(t^n) + \mathcal{O}[(\Delta t)^4]\right\} - u(t^n) - \Delta t U[u(t^n), \eta(t^n)]\\ &= \left\{u(t^n) + \Delta t\,\underbrace{\frac{\partial u}{\partial t}(t^n)}_{=-c\frac{\partial \eta}{\partial x}(t^n)} + \frac{(\Delta t)^2}{2}\underbrace{\frac{\partial^2 u}{\partial t^2}(t^n)}_{=c^2\frac{\partial^2 u}{\partial x^2}(t^n)} + \frac{(\Delta t)^3}{6}\frac{\partial^3 u}{\partial t^3}(t^n) + \mathcal{O}[(\Delta t)^4]\right\}\\ &\quad + \left\{-u(t^n) + \Delta t\,c\frac{\partial \eta^n}{\partial x}(t^n) - \frac{(\Delta t)^2}{2}c^2\frac{\partial^2 u^n}{\partial x^2}(t^n) + \frac{(\Delta t)^3}{6}c^3(\beta_3 + 1)\frac{\partial^3 \eta^n}{\partial x^3}(t^n) + \mathcal{O}[(\Delta t)^4]\right\}\\ &= \left\{\frac{(\Delta t)^3}{6}\frac{\partial^3 u}{\partial t^3}(t^n) + \frac{(\Delta t)^3}{6}c^3(\beta_3 + 1)\frac{\partial^3 \eta}{\partial x^3}(t^n) + \mathcal{O}[(\Delta t)^4]\right\}.\end{aligned} \tag{A10}$$

From (A10), we see that $\Delta t\, \tau_u^n = \mathcal{O}[(\Delta t)^3]$ no matter the values of the FB weights, so

$$\tau_u^n = \mathcal{O}[(\Delta t)^2]. \tag{A11}$$

Next simplify (A9) in a similar way:

$$\begin{aligned}
\Delta t\, \tau_\eta^n &= \left\{\sum_{k=0}^{3} \frac{(\Delta t)^k}{k!} \frac{\partial^k \eta}{\partial t^k}(t^n) + \mathcal{O}[(\Delta t)^4]\right\} - \eta(t^n) - \Delta t H[u(t^n),\, \eta(t^n)] \\
&= \left\{\eta(t^n) + \Delta t \underbrace{\frac{\partial \eta}{\partial t}(t^n)}_{=-c\frac{\partial u}{\partial x}(t^n)} + \frac{(\Delta t)^2}{2} \underbrace{\frac{\partial^2 \eta}{\partial t^2}(t^n)}_{=c^2\frac{\partial^2 \eta}{\partial x^2}(t^n)} + \frac{(\Delta t)^3}{6} \frac{\partial^3 \eta}{\partial t^3}(t^n) + \mathcal{O}[(\Delta t)^4]\right\} \\
&\quad + \left\{-\eta(t^n) + \Delta t\, c \frac{\partial u^n}{\partial x}(t^n) - \frac{(\Delta t)^2}{2} c^2 \frac{\partial^2 \eta^n}{\partial x^2}(t^n) + \frac{(\Delta t)^3}{4} c^3 \beta_2 \frac{\partial^3 u^n}{\partial x^3}(t^n) + \mathcal{O}[(\Delta t)^4]\right\} \\
&= \left\{\frac{(\Delta t)^3}{6} \frac{\partial^3 \eta}{\partial t^3}(t^n) + \frac{(\Delta t)^3}{4} c^3 \beta_2 \frac{\partial^3 u}{\partial x^3}(t^n) + \mathcal{O}[(\Delta t)^4]\right\}.
\end{aligned} \tag{A12}$$

From (A12), we see that with no assumptions on the FB weights, we have $\Delta t \tau_\eta^n = \mathcal{O}[(\Delta t)^3]$, so

$$\tau_\eta^n = \mathcal{O}[(\Delta t)^2]. \tag{A13}$$

b. Linearization of the shallow-water equations

Here, we present the derivation of (4), the linearized SWEs centered about a nonzero mean flow. The nonlinear SWEs on a rotating sphere are given by

$$\frac{\partial \breve{\mathbf{u}}}{\partial t} + (\nabla \times \breve{\mathbf{u}} + f\mathbf{k}) \times \breve{\mathbf{u}} = -\nabla \frac{|\breve{\mathbf{u}}|^2}{2} - g\nabla \breve{h},$$

$$\frac{\partial \breve{h}}{\partial t} + \nabla \cdot (\breve{h}\, \breve{\mathbf{u}}) = 0, \tag{A14}$$

where $\breve{\mathbf{u}}(x, y, t) = [\breve{u}(x, y, t), \breve{v}(x, y, t)]$ is the horizontal fluid velocity, x and y are the spatial coordinates, t is the time coordinate, f is the Coriolis parameter, g is the gravitational constant, and $\breve{h}$ is the fluid thickness.

To linearize (A14) about a nonzero mean flow, assume that

$$\breve{\mathbf{u}}(x, y, t) = \tilde{\mathbf{U}} + \tilde{\mathbf{u}}(x, y, t), \tag{A15}$$

$$\breve{h}(x, y, t) = H + \tilde{h}(x, y, t), \tag{A16}$$

where $\tilde{\mathbf{U}} = (\tilde{U}, \tilde{V})$ is the constant (with respect to time and space) mean horizontal fluid velocity, $\tilde{\mathbf{u}} = [\tilde{u}(x, y, t), \tilde{v}(x, y, t)]$ is a small perturbation of $\tilde{\mathbf{U}}$, H is the constant (with respect to time and space) fluid thickness when at rest, and $\tilde{h} = \tilde{h}(x, y, t)$ is a small perturbation of H. In both cases, small means that the quantities themselves, as well as their derivatives, are negligible when multiplied together or squared.

Now, we make the substitutions $\breve{\mathbf{u}} = \tilde{\mathbf{U}} + \tilde{\mathbf{u}}$ and $\breve{h} = H + \tilde{h}$ and simplify term by term. The kinetic energy term becomes

$$\begin{aligned}
-\nabla \frac{\breve{\mathbf{u}}^2}{2} &= -\nabla \frac{1}{2}(\breve{u}^2 + \breve{v}^2) \\
&= -\nabla \frac{1}{2}(\tilde{U}^2 + 2\tilde{U}\tilde{u} + \tilde{u}^2 + \tilde{V}^2 + 2\tilde{V}\tilde{v} + \tilde{v}^2) \\
&= -\tilde{U}\nabla\tilde{u} - \tilde{V}\nabla\tilde{v} - \frac{1}{2}(\nabla\tilde{u}^2 + \nabla\tilde{v}^2) \\
&\approx -\tilde{U}\nabla\tilde{u} - \tilde{V}\nabla\tilde{v} \\
&= \tilde{\mathbf{U}} \cdot \nabla\tilde{\mathbf{u}},
\end{aligned} \tag{A17}$$

where $\nabla\tilde{\mathbf{u}}$ is the Jacobian matrix,

$$\begin{pmatrix} \partial_x \tilde{u} & \partial_x \tilde{v} \\ \partial_y \tilde{u} & \partial_y \tilde{v} \end{pmatrix},$$

and $\tilde{\mathbf{U}} \cdot \nabla\tilde{\mathbf{u}}$ denotes a 2 × 1 column vector wherein the ith entry is the dot product of $\tilde{\mathbf{U}}$ with the ith row of $\nabla\tilde{\mathbf{u}}$. The potential vorticity and Coriolis term becomes

$$\begin{aligned}
(\nabla \times \breve{\mathbf{u}} + f\mathbf{k}) \times \breve{\mathbf{u}} &= \left[\left(\frac{\partial \breve{v}}{\partial x} - \frac{\partial \breve{u}}{\partial y}\right)\mathbf{k} + f\mathbf{k}\right] \times \breve{\mathbf{u}} \\
&= \mathbf{i}\left[-\breve{v}\left(\frac{\partial \breve{v}}{\partial x} - \frac{\partial \breve{u}}{\partial y} + f\right)\right] + \mathbf{j}\left[\breve{u}\left(\frac{\partial \breve{v}}{\partial x} - \frac{\partial \breve{u}}{\partial y} + f\right)\right] \\
&= \mathbf{i}\left[-f(\tilde{V} + \tilde{v}) - \tilde{V}\left(\frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right) - \tilde{v}\left(\frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right)\right] \\
&\quad + \mathbf{j}\left[f(\tilde{U} + \tilde{u}) + \tilde{U}\left(\frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right) + \tilde{u}\left(\frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right)\right] \\
&\approx \mathbf{i}\left[-f(\tilde{V} + \tilde{v}) - \tilde{V}\left(\frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right)\right] + \mathbf{j}\left[f(\tilde{U} + \tilde{u}) + \tilde{U}\left(\frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right)\right] \\
&= f(\tilde{\mathbf{U}} + \tilde{\mathbf{u}})^{\perp} + \tilde{\zeta}\tilde{\mathbf{U}}^{\perp},
\end{aligned} \tag{A18}$$

where $\tilde{\zeta} = (\partial\tilde{v}/\partial x) - (\partial\tilde{u}/\partial y)$ is the vertical component of the vorticity $\nabla \times \tilde{\mathbf{u}}$. The thickness gradient becomes

$$\begin{aligned}
-g\nabla\breve{h} &= -g\nabla(H + \tilde{h}) \\
&= -g\nabla\tilde{h}\, \cdot
\end{aligned} \tag{A19}$$

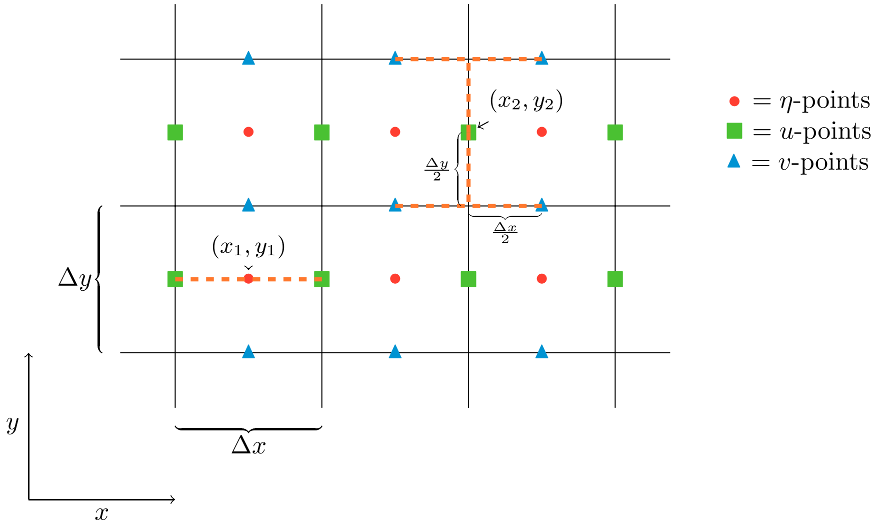


FIG. A1. An example Arakawa C-grid.

Finally, the mass advection term becomes

$$\begin{aligned}\nabla\cdot(\breve{h}\breve{\mathbf{u}}) &= \nabla\cdot(H\tilde{\mathbf{U}} + H\tilde{\mathbf{u}} + \tilde{h}\tilde{\mathbf{U}} + \tilde{h}\tilde{\mathbf{u}})\\ &= \nabla\cdot(H\tilde{\mathbf{u}} + \tilde{h}\tilde{\mathbf{U}} + \tilde{h}\tilde{\mathbf{u}})\\ &\approx H(\nabla\cdot\tilde{\mathbf{u}}) + \nabla\cdot(\tilde{h}\tilde{\mathbf{U}})\cdot \qquad \text{(A20)}\end{aligned}$$

So, linearized SWEs with a nonzero mean flow are given by

$$\begin{aligned}&\frac{\partial\tilde{\mathbf{u}}}{\partial t} + f(\tilde{\mathbf{U}} + \tilde{\mathbf{u}})^{\perp} + \tilde{\zeta}\tilde{\mathbf{U}}^{\perp} = -\tilde{\mathbf{U}}\cdot\boldsymbol{\nabla}\tilde{\mathbf{u}} - g\nabla\tilde{h}\\ &\frac{\partial\tilde{h}}{\partial t} + H(\nabla\cdot\tilde{\mathbf{u}}) + \nabla\cdot(\tilde{h}\tilde{\mathbf{U}}) = 0. \qquad \text{(A21)}\end{aligned}$$

c. Arakawa C-grid discretization

Figure A1 shows an example of a rectangular, staggered Arakawa C-grid Arakawa and Lamb (1977). Here, the mass variable η is computed at cell centers and the normal components of velocity u and v are computed at the center of cell edges.

To see how to Eqs. (6) and (7) are obtained, consider the following example wherein we compute a derivative in x at the point (x_1, y_1) in Fig. A1. On this grid, the derivative $\partial/\partial x$ of some quantity can be approximated by taking a centered difference of values at the two nearest points:

$$\begin{aligned}\frac{\partial}{\partial x}[\exp(ikx_1 + i\ell y_1)] &= \frac{\exp\left[ik\left(x_1 + \frac{\Delta x}{2}\right) + i\ell y_1\right] - \exp\left[ik\left(x_1 + \frac{\Delta x}{2}\right) + i\ell y_1\right]}{\Delta x}\\ &= \frac{e^{ik(\Delta x/2)} - e^{-ik(\Delta x/2)}}{\Delta x}\exp(ikx_1 + i\ell y_1)\\ &= \frac{i2\sin\left(k\frac{\Delta x}{2}\right)}{\Delta x}\exp(ikx_1 + i\ell y_1)\cdot\end{aligned}$$

An identical calculation hold for derivatives in y.

To calculate the Coriolis terms, we need data for u at v points, and for v at u points, called u_v and v_u, respectively. To see how Eqs. (8) and (9) are obtained, consider the following example wherein we calculate v_u at (x_2, y_2) in Fig. A1 by taking the average of v data at the four nearest v points:

$$
\begin{aligned}
v_u(x_2, y_2, t) &= \frac{1}{4}\hat{v}(k, \ell, t)\left\{\exp\left[ik\left(x_2 + \frac{\Delta x}{2}\right) + i\ell\left(y_2 + \frac{\Delta y}{2}\right)\right] + \exp\left[ik\left(x_2 + \frac{\Delta x}{2}\right) - i\ell\left(y_2 + \frac{\Delta y}{2}\right)\right]\right.\\
&\quad \left. + \exp\left[ik\left(x_2 - \frac{\Delta x}{2}\right) + i\ell\left(y_2 + \frac{\Delta y}{2}\right)\right] + \exp\left[ik\left(x_2 - \frac{\Delta x}{2}\right) + i\ell\left(y_2 - \frac{\Delta y}{2}\right)\right]\right\}\\
&= \frac{1}{4}\left[\exp\left(ik\frac{\Delta x}{2} + i\ell\frac{\Delta y}{2}\right) + \exp\left(ik\frac{\Delta x}{2} - i\ell\frac{\Delta y}{2}\right) + \exp\left(-ik\frac{\Delta x}{2} + i\ell\frac{\Delta y}{2}\right) + \exp\left(-ik\frac{\Delta x}{2} - i\ell\frac{\Delta y}{2}\right)\right]\hat{v}(k, \ell, t)\exp(ikx_2 + i\ell y_2)\\
&= \frac{1}{4}\left[4\cos\left(k\frac{\Delta x}{2}\right)\cos\left(\ell\frac{\Delta y}{2}\right)\right]\hat{v}(k, \ell, t)\exp(ikx_2 + i\ell y_2)\\
&= \cos\left(k\frac{\Delta x}{2}\right)\cos\left(\ell\frac{\Delta y}{2}\right)\hat{v}(k, \ell, t)\exp(ikx_2 + i\ell y_2).
\end{aligned}
$$

An identical calculation holds for u_v.

d. Formulation of $\tilde{\mathbf{G}}$

Assume that the mean flow **U** in the linearized SWEs (5) is taken to be zero, and apply a Fourier transform in space. The resulting equations can be written as

$$
\frac{\partial}{\partial t}\begin{pmatrix}\hat{u}\\ \hat{v}\\ \hat{\eta}\end{pmatrix} = \begin{pmatrix}0 & f & -ick\\ -f & 0 & -ic\ell\\ -ick & -ic\ell & 0\end{pmatrix}\begin{pmatrix}\hat{u}\\ \hat{v}\\ \hat{\eta}\end{pmatrix}. \tag{A22}
$$

These equations are spatially continuous counterparts to (10).

The general solution to this system is given by

$$
\hat{\mathbf{w}}(t) = e^{\mathbf{A}t}\hat{\mathbf{w}}(0), \tag{A23}
$$

where $\hat{\mathbf{w}}(t) = [\hat{u}(t),\hat{v}(t),\hat{\eta}(t)]^{\mathrm{T}}$ and $\mathbf{A}$ is as given in (18). Now, consider the general solution of the system at time $t = \Delta t$. Then,

$$
\hat{\mathbf{w}}(\Delta t) = e^{\mathbf{A}\Delta t}\hat{\mathbf{w}}(0). \tag{A24}
$$

Multiplying initial data $\hat{\mathbf{w}}(0) = \hat{\mathbf{w}}^0$ by the numerical amplification matrix $\mathbf{G}$ defined in (12) returns the numerical solution after one time step. Assume that the FB-RK(3,2) scheme produces an exact solution, that is, assume $\mathbf{G}\hat{\mathbf{w}}^0 = \hat{\mathbf{w}}(\Delta t)$. Then

$$
\mathbf{G}\hat{\mathbf{w}}^0 = e^{\mathbf{A}\Delta t}\hat{\mathbf{w}}^0. \tag{A25}
$$

This holds for any choice of initial data $\hat{\mathbf{w}}^0$. Take $\hat{\mathbf{w}}^0 = \mathbf{e}_i$, where $\mathbf{e}_i$ is the ith standard basis vector for $i = 1, 2, 3$ to get

$$
\mathbf{G}\mathbf{e}_i = e^{\mathbf{A}\Delta t}\mathbf{e}_i \qquad \text{for } i = 1, 2, 3. \tag{A26}
$$

Because $\mathbf{G}\mathbf{e}_i$ and $e^{A\Delta t}\mathbf{e}_i$ are just the ith columns of $\mathbf{G}$ and $e^{\mathbf{A}\Delta t}$, respectively, it follows that $\mathbf{G} = e^{\mathbf{A}\Delta t}$. The reverse implication is trivial. To simplify notation, we write

$$
\tilde{\mathbf{G}} := e^{\mathbf{A}\Delta t}. \tag{A27}
$$

e. Constructing balanced initial conditions

While the shallow-water cases described by Galewsky et al. (2004) and Williamson et al. (1992) have been successfully implemented for various structured grid models, we briefly describe an initialization procedure that is used to produce geostrophically balanced initial conditions for the unstructured TRiSK-like model used in this study. Introducing a matrix–vector form of the nonlinear SWE system presented in (A14)

$$
\begin{aligned}
\frac{\partial u}{\partial t} + (\mathbf{C}u + f)u^{\perp} &= -\mathbf{G}[g(h + z_b) + K],\\
\frac{\partial h}{\partial t} + \mathbf{D}(uh) &= 0,\\
K &= \frac{1}{2}|u|^2,
\end{aligned} \tag{A28}
$$

where $\mathbf{D}$, $\mathbf{G}$, $\mathbf{C}$ are sparse linear operators encoding the action of divergence, gradient and curl on a given discrete vector, as per Ringler et al. (2010). A balanced thickness distribution h^0 can be found for a given velocity field u^0 by setting $\partial u/\partial t = 0$ and taking the divergence of momentum tendencies:

$$
\begin{aligned}
K^0 &= \frac{1}{2}|u^0|^2,\\
\mathbf{D}[(\mathbf{C}u^0 + f)(u^0)^{\perp} + \mathbf{G}K^0] &= -g\mathbf{D}\mathbf{G}h^0.
\end{aligned} \tag{A29}
$$

Recognizing the product $\mathbf{D}\mathbf{G} = \mathbf{L}$ is an approximation of the continuous Laplacian ∇^2, (A29) can be seen as an elliptic problem that can be solved by inverting the system of linear equations:

$$
h^0 = -\frac{1}{g}\mathbf{L}^{-1}\mathbf{D}[(\mathbf{C}u^0 + f)(u^0)^{\perp} + \mathbf{G}K^0]\cdot \tag{A30}
$$

In this study we solve (A30) using SciPy's (Virtanen et al. 2020) gcrotmk iterative solver. Compared to the evaluation of analytical profiles, the construction of initial conditions through the solution of (A30) is agnostic to details of the computational mesh used and ensures the discrete state is balanced with respect to the spatial operators used to advance the flow. This approach has proved robust for the unstructured model configurations considered in this study.

REFERENCES

Arakawa, A., and V. R. Lamb, Eds., 1977: Computational design of the basic dynamical processes of the UCLA general circulation model. *Methods in Computational Physics: Advances in Research and Applications*, J. Chang, Ed., General Circulation Models of the Atmosphere, Vol. 17, Elsevier, 173–265, https://doi.org/10.1016/B978-0-12-460817-7.50009-4.

Azerad, P., J.-L. Guermond, and B. Popov, 2017: Well-balanced second-order approximation of the shallow water equation with continuous finite elements. *SIAM J. Numer. Anal.*, **55**, 3203–3224, https://doi.org/10.1137/17M1122463.

Bao, L., R. D. Nair, and H. M. Tufo, 2014: A mass and momentum flux-form high-order discontinuous Galerkin shallow water model on the cubed-sphere. *J. Comput. Phys.*, **271**, 224–243, https://doi.org/10.1016/j.jcp.2013.11.033.

Capodaglio, G., and M. Petersen, 2022: Local time stepping for the shallow water equations in MPAS. *J. Comput. Phys.*, **449**, 110818, https://doi.org/10.1016/j.jcp.2021.110818.

Chassignet, E. P., H. E. Hurlburt, O. M. Smedstad, G. R. Halliwell, P. J. Hogan, A. J. Wallcraft, R. Baraille, and R. Bleck, 2007: The HYCOM (HYbrid Coordinate Ocean Model) data assimilative system. *J. Mar. Syst.*, **65**, 60–83, https://doi.org/10.1016/j.jmarsys.2005.09.016.

Collins, W., and Coauthors, 2004: Description of the NCAR Community Atmosphere Model (CAM 3.0). NCAR Tech. Note NCAR/TN-464+STR, 214 pp., https://doi.org/10.5065/D63N21CH.

Dawson, C., C. J. Trahan, E. J. Kubatko, and J. J. Westerink, 2013: A parallel local timestepping Runge–Kutta discontinuous Galerkin method with applications to coastal ocean modeling. *Comput. Methods Appl. Mech. Eng.*, **259**, 154–165, https://doi.org/10.1016/j.cma.2013.03.015.

Endres, S. C., C. Sandrock, and W. W. Focke, 2018: A simplicial homology algorithm for Lipschitz optimisation. *J. Global Optim.*, **72**, 181–217, https://doi.org/10.1007/s10898-018-0645-y.

Engwirda, D., 2017: JIGSAW-GEO (1.0): Locally orthogonal staggered unstructured grid generation for general circulation modelling on the sphere. *Geosci. Model Dev.*, **10**, 2117–2140, https://doi.org/10.5194/gmd-10-2117-2017.

Engwrida, D., and J. R. Lilly, 2023: Source code for swe-python, version 1. Zenodo, accessed 22 May 2023, https://doi.org/10.5281/zenodo.7958483.

Fu, G., 2022: A high-order velocity-based discontinuous Galerkin scheme for the shallow water equations: Local conservation, entropy stability, well-balanced property, and positivity preservation. *J. Sci. Comput.*, **92**, 86, https://doi.org/10.1007/s10915-022-01902-y.

Galewsky, J., R. K. Scott, and L. M. Polvani, 2004: An initial-value problem for testing numerical models of the global shallow-water equations. *Tellus*, **56A**, 429–440, https://doi.org/10.3402/tellusa.v56i5.14436.

Golaz, J.-C., and Coauthors, 2022: The DOE E3SM model version 2: Overview of the physical model and initial model evaluation. *J. Adv. Model. Earth Syst.*, **14**, e2022MS003156, https://doi.org/10.1029/2022MS003156.

Gottlieb, S., and C.-W. Shu, 1998: Total variation diminishing Runge-Kutta schemes. *Math. Comput.*, **67**, 73–85, https://doi.org/10.1090/S0025-5718-98-00913-2.

Hoang, T.-T.-P., W. Leng, L. Ju, Z. Wang, and K. Pieper, 2019: Conservative explicit local time-stepping schemes for the shallow water equations. *J. Comput. Phys.*, **382**, 152–176, https://doi.org/10.1016/j.jcp.2019.01.006.

Kärnä, T., S. C. Kramer, L. Mitchell, D. A. Ham, M. D. Piggott, and A. M. Baptista, 2018: Thetis coastal ocean model: Discontinuous Galerkin discretization for the three-dimensional hydrostatic equations. *Geosci. Model Dev.*, **11**, 4359–4382, https://doi.org/10.5194/gmd-11-4359-2018.

Katta, K. K., R. D. Nair, and V. Kumar, 2015: High-order finite volume shallow water model on the cubed-sphere: 1D reconstruction scheme. *Appl. Math. Comput.*, **266**, 316–327, https://doi.org/10.1016/j.amc.2015.04.053.

Kerbyson, D. J., and P. W. Jones, 2005: A performance model of the parallel ocean program. *Int. J. High Perform. Comput. Appl.*, **19**, 261–276, https://doi.org/10.1177/1094342005056114.

Lan, R., L. Ju, Z. Wang, M. Gunzburger, and P. Jones, 2022: High-order multirate explicit time-stepping schemes for the baroclinic-barotropic split dynamics in primitive equations. *J. Comput. Phys.*, **457**, 111050, https://doi.org/10.1016/j.jcp.2022.111050.

LeVeque, R. J., Ed., 2007: Diffusion equations and parabolic problems. *Finite Difference Methods for Ordinary and Partial Differential Equations*, Other Titles in Applied Mathematics, Society for Industrial and Applied Mathematics, 181–200, https://doi.org/10.1137/1.9780898717839.ch9.

Lilly, J. R., 2023a: Data for ‘CFL optimized forward-backward Runge-Kutta schemes for the shallow water equations,’ version 1. Zenodo, accessed 22 May 2023, https://doi.org/10.5281/zenodo.7958518.

——, 2023b: Source code for rk-fb-optimization, version 1. Zenodo, accessed 22 May 2023, https://doi.org/10.5281/zenodo.7958458.

——, G. Capodaglio, M. R. Petersen, S. R. Brus, D. Engwirda, and R. L. Higdon, 2023: Storm surge modeling as an application of local time-stepping in MPAS-ocean. *J. Adv. Model. Earth Syst.*, **15**, e2022MS003327, https://doi.org/10.1029/2022MS003327.

Madec, G., and Coauthors, 2017: NEMO ocean engine, version 3.6. Notes du Pôle de modélisation de l’Institut Pierre-Simon Laplace (IPSL) 27, 412 pp., https://doi.org/10.5281/zenodo.3248739.

Mesinger, F., 1977: Forward-backward scheme, and its use in a limited area model. *Contrib. Atmos. Phys.*, **50**, 200–210.

Petersen, M. R., and Coauthors, 2019: An evaluation of the ocean and sea ice climate of E3SM using MPAS and interannual CORE-II forcing. *J. Adv. Model. Earth Syst.*, **11**, 1438–1458, https://doi.org/10.1029/2018MS001373.

Purser, R. J., 2007: Accuracy considerations of time-splitting methods for models using two-time-level schemes. *Mon. Wea. Rev.*, **135**, 1158–1164, https://doi.org/10.1175/MWR3339.1.

Ringler, T. D., J. Thuburn, J. B. Klemp, and W. C. Skamarock, 2010: A unified approach to energy conservation and potential vorticity dynamics for arbitrarily-structured C-grids. *J. Comput. Phys.*, **229**, 3065–3090, https://doi.org/10.1016/j.jcp.2009.12.007.

——, M. Petersen, R. L. Higdon, D. Jacobsen, P. W. Jones, and M. Maltrud, 2013: A multi-resolution approach to global ocean modeling. *Ocean Modell.*, **69**, 211–232, https://doi.org/10.1016/j.ocemod.2013.04.010.

Roullet, G., and T. Gaillard, 2022: A fast monotone discretization of the rotating shallow water equations. *J. Adv. Model. Earth Syst.*, **14**, e2021MS002663, https://doi.org/10.1029/2021MS002663.

Shchepetkin, A. F., and J. C. McWilliams, 2005: The Regional Oceanic Modeling System (ROMS): A split-explicit, free-surface, topography-following-coordinate oceanic model. *Ocean Modell.*, **9**, 347–404, https://doi.org/10.1016/j.ocemod.2004.08.002.

——, and ——, 2009: Computational kernel algorithms for fine-scale, multiprocess, longtime oceanic simulations. *Handbook of Numerical Analysis*, R. M. Temam and J. J. Tribbia, Eds., Computational Methods for the Atmosphere and the Oceans, Vol. 14, Elsevier, 121–183, https://doi.org/10.1016/S1570-8659(08)01202-0.

Skamarock, W. C., J. B. Klemp, M. G. Duda, L. D. Fowler, S.-H. Park, and T. D. Ringler, 2012: A multiscale nonhydrostatic atmospheric model using centroidal Voronoi tesselations and C-grid staggering. *Mon. Wea. Rev.*, **140**, 3090–3105, https://doi.org/10.1175/MWR-D-11-00215.1.

Ullrich, P. A., C. Jablonowski, and B. van Leer, 2010: High-order finite-volume methods for the shallow-water equations on the sphere. *J. Comput. Phys.*, **229**, 6104–6134, https://doi.org/10.1016/j.jcp.2010.04.044.

Virtanen, P., and Coauthors, 2020: SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nat. Methods*, **17**, 261–272, https://doi.org/10.1038/s41592-019-0686-2.

Wicker, L. J., and W. C. Skamarock, 2002: Time-splitting methods for elastic models using forward time schemes. *Mon. Wea. Rev.*, **130**, 2088–2097, https://doi.org/10.1175/1520-0493(2002)130<2088:TSMFEM>2.0.CO;2.

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