Computational Kernel Algorithms for Fine-Scale, Multi-Process, Long-Time Oceanic Simulations

Alexander F. Shchepetkin and James C. McWilliams

Institute of Geophysics and Planetary Physics, University of California at Los Angeles, 405 Hilgard Avenue, Los Angeles, CA 90095-1567

e-mail: alex@atmos.ucla.edu jcm@atmos.ucla.edu

Published in: Handbook of Numerical Analysis, Vol. XIV: Computational Methods for the Ocean and the Atmosphere, P. G. Ciarlet, editor, R. Temam & J. Tribbia, guest eds., *Elsevier Science*, pp. 121-183, doi:10.1016/S1570-8659(08)01202-0, 2008.

Abstract

Progress in computer technology has made it possible to make larger calculations with finer grid-scale resolution, and physical processes that were beyond the reach of coarse-resolution models are now simulated directly. This focuses scientific interest toward more turbulent flow regimes, and applications toward more realistic modeling of specific regional configurations.

In this article we examine the numerical design of oceanic modeling codes specifically suited for modern demands. These are compared to traditional "legacy" oceanic general circulation models, as well as to computational fluid dynamics methods for modern engineering applications. Our primary subject is how the numerical algorithms for different aspects of the discretized partial differential equation system — the computational kernel — combine to yield the overall model performance, with particular focus on avoiding destructive interference among algorithmic components.

Key words: oceanic modeling, free-surface, mode-splitting, conservation properties, computational stability, numerical algorithm interference

1 Introduction: Integrated Kernel Design

Oceanic General Circulation Models (OGCMs) (Bryan & Cox, 1969; Blumberg & Mellor, 1987; Bleck & Smith, 1990; Dukowitz & Smith, 1994; McWilliams, 1996; Marshall *et al.*, 1997; Griffies *et al.*, 2000) have historically been a separate branch of computational fluid mechanics with significantly different choices for numerical methods compared to most engineering CFD (computational fluid dynamics) applications. The main motivation is the need to perform very long — even millennial — simulations over hundreds of thousands of time steps, which makes it essential to ensure conservation properties for the mean and variance of model fields (Lilly, 1965; Arakawa & Lamb, 1977). This typically has led to the choice of

discrete algorithms as a combination of basic second-order, centered spatial operators and leap-frog time-stepping, both because they can easily be made to assure desired conservation properties and because higher-order advection schemes usually do not give better solutions for coarse grids that do not adequately resolve the baroclinic deformation radii (*i.e.*, the "non-eddy-resolving" regime typical for climate studies). Instead, coarse-grid models have to rely on parameterizations of subgrid mesoscale processes to achieve physically correct results (Gent & McWilliams, 1990; Griffies *et al.*, 1998). The OGCM codes targeting large-scale circulation were ill-suited for nearshore phenomena due to inaccurate handling of complex geometry, bottom topography, free surface, and bottom boundary layer; coastal model developments took a rather independent route (Casulli & Cheng, 1992; Casulli & Cattani, 1994; Casulli & Stelling, 1998; Casulli, 1999) with more focus on achieving accurate dispersion properties for surface gravity waves, wetting and drying capabilities, *etc.*, with less emphasis on long-term conservation properties and Coriolis-force effects. These coastal codes are characterized by two-time-level time-stepping, upstream-biased, semi-Lagrangian, monotonicity-preserving advection schemes, and sometimes non-hydrostatic effects and unstructured grids (Cheng & Casulli, 2001). The combination of these features makes them more similar to CFD codes than to OGCMs.

During the 1990s all major OGCMs underwent a substantial redesign in order to take advantage of the rapidly developing computer technology, especially parallel processing. This allowed much larger computational grids, ultimately ones that can close the resolution gap between coastal and regional-global applications. A somewhat paradoxical outcome of this evolution is the use of parallel codes has become the wide-spread in oceanic modeling, while as yet there has been relatively little overhaul of the numerical methods in their hydrodynamic kernels. Most of the recent model content developments have come in physical parameterizations and peripheral modules for biogeochemical processes. Griffies *et al.* (2000) is an overview of the modern state of OGCMs in climate modeling. Some rare exceptions to the widespread use of classical time-stepping and second-order advection algorithms have been adopted to avoid spurious oscillations and negative concentrations for material tracers (Willebrand *et al.*, 2001), but only rarely are better advection schemes used for momentum (Dietrich *et al.*, 1987, 1997). Monotonic advection schemes are also used in the context of isopycnic layer models to deal with vanishing layer thickness (Bleck & Smith, 1990).

The code organizational structure in the Modular Ocean Model (MOM) became another *de facto* standard, adopted by many modelers when their code complexity matured to OGCM status. This approach is encouraged and often justified by the ease of incorporating peripheral modules. However, it led to a widespread "modular vision" of the kernel, and the interaction and, in fact, interference among the algorithmic components was often overlooked. For example, allowing a free-surface in a previously rigid-lid model, (Killworth *et al.*, 1991; Dukowitz & Smith, 1994) may result in the loss of conservation and/or constancy-preservation properties of control-volume scheme for tracer advection [1] , which was noticed, mitigated (Griffies *et al.*, 2001), and eliminated completely (Campin *et al.*, 2004; Shchepetkin & McWilliams, 2005; Marsaleix *et al.*, 2008) only a decade later. If one wants to implement a non-oscillatory advection scheme for tracers, the tracer time step has to be changed from leapfrog to a two-time level algorithm, *e.g.*, predictor-corrector. However, since this change applies to tracers only, it leads to an under-utilization of its potential benefit because the time-step size Δt of an OGCM is usually limited by the gravity-wave speed for the first baroclinic mode. The gravity wave behavior arises from an interplay between the momentum and the tracer equations and cannot be improved by refining tracers alone.

There is a common practice of two-stage code development, where a single-processor prototype code is parallelized later, only after being considered sufficiently mature. This is another reason for sub-optimal

[1] The exact cause of this loss and a remedy are considered later in Sec. 3.1.

algorithmic choices, because considerations of computational efficiency (cost) may be quite different between parallel or non-parallel cases. For example, the treatment of the Coriolis force for the barotropic (*i.e.*, depth-averaged) mode on a C-grid with an alternating-direction method (Bleck & Smith, 1990) or a fixed-point iteration procedure (Higdon, 2005) is straightforward on a single processor, but, due to the staggered placement of u- and v-points on a C-grid and the associated interpolation, it results in excessive synchronization and message passing in a parallel implementation. In contrast, for even moderately high spatial and temporal resolution, the associated stability-limiting Courant number is very small ($f\Delta t \ll 1$, where f is the Coriolis frequency). The Coriolis force can be successfully treated in parallel with virtually any explicit, conditionally stable time-stepping algorithm. Another interesting example comes from the experience of parallelization on shared-memory computers: a very efficient code can be obtained by arranging the mathematical operations in such a way that intermediate results are stored in cache-sized private arrays that are reused in as many stages as possible before a global synchronization event takes place. This experience thus stimulates the use of multi-stage, high-order accurate, wide-stencil algorithms because they naturally allow a higher computational density (*i.e.*, in this context the ratio of mathematical operations to cache-to-main-memory memory loads and stores). From this point of view, the recent tendency to develop an abstract Earth-System Modeling Framework (`www.esmf.ucar.edu`), driven primarily by computer scientists, has the danger of decoupling physical-modeling from code-infrastructure decisions, as a further commitment to modular architectures. While this approach may indeed save modelers labor by providing common code components, it also can have the effect of hiding or even impeding the resolution of the types of algorithmic interferences that are the focus of this article.

In our designs for the computational kernel in the Regional Oceanic Modeling System (ROMS) (Shchepetkin & McWilliams, 2005), we adopt an integrated approach where we try to analyze and take into account all previously known experience, but in such a way that no component from a legacy code is accepted *a priori*. Rather, we try to identify potential algorithmic interferences and conflicts and their possible reconciliations. This principle encompasses a full range of considerations, from the theoretical analysis of a linearized time-stepping scheme all the way to cross- and within-processor code optimization issues.

The advantage of using higher-order advection schemes for turbulent flows is well understood (Orszag, 1971; Leonard *et al.*, 1996; Shchepetkin & McWilliams, 1998). This approach exposes the primary criterion not as the formal order of accuracy *per se* (which is merely a Taylor series estimate of the asymptotic convergence rate for smooth functions), but rather as the spectral bandwidth (*i.e.*, the fraction of grid-resolved Fourier components that are correctly represented by the discretized operator). In practice this translates into downplaying the goal of achieving a uniformly high-order of accuracy for all terms in the governing equations — a rather unrealistic hope for a multi-scale, multi-process, nonlinear system anyway — in favor of isolating and removing specific causes of accuracy loss in particular solution regimes. Although ROMS has been successfully used for coarse-resolution climate studies (Haidvogel *et al.*, 2000), its main intended applications are medium- or high-resolution simulations with a well-resolved baroclinic deformation radius and strong advective influences. Thus, it is intended to simulate mesoscale, approximately geostrophically balanced currents and eddies, together with nonlinear gravity and inertial waves with similar spatial scales. This downplays the importance of eddy parameterization in comparison with most climate models. However, the need to avoid erroneous vertical mixing, especially across isopycnic surfaces in stably stratified regions, is a high priority for long-term simulations. For this reason the use of upstream-biased advection schemes in the vertical direction is discouraged. The Δt value is expected to be limited by the phase speeds for barotropic and baroclinic gravity waves (*i.e.*, external and internal modes, respectively), which are different from each other by at least an order of magnitude (barotropic is faster). The first-mode baroclonic speed is usually larger than the advective velocity, although comparable in its order of magnitude. The baroclinic time step is expected to be much smaller than the inertial period, so

that the Coriolis force does not impose any additional restriction on Δt. Vertical mixing is always treated implicitly since its transport rate can be much larger than the vertical advective rate.

Taking into account the specifics of this physical regime, we have been developing the kernel code in ROMS to have the features in the following list (*cf.*, Fig. 1) that foreshadows the algorithms discussed in more detail below.

- *Vertical Coordinate*: Although ROMS nominally belongs to the vertical-boundary-following model family (*i.e.*, $\sigma(x, y, t)$-coordinate), the code stores the height-coordinate transform $z = z(x, y, \sigma)$ as a special array, and, in principle, it can be used as a generalized vertical coordinate.

- *Free Surface:* ROMS is free-surface model with split-explicit time-stepping. The pressure-gradient force (PGF) for the barotropic mode is defined as a variational derivative of vertical integral of the hydrostatic PGF with respect to perturbations in the free-surface elevation $\zeta(x, y, t)$. As a result the barotropic PGF depends not only on ζ but also on the two differently averaged density fields indicated by two ascending arrows, ρ^* and $\overline{\rho}$ in Fig. 1 (Sec. 3.2) that are computed from 3D fields and held constant during barotropic time-stepping. This insures an accurate and stable split, even with a large ratio between the Δt for the baroclinic and barotropic modes.

- *Barotropic Averaging:* The barotropic variables are averaged in the fast (barotropic) time step to prevent aliasing of frequencies not resolved by the slow (3D baroclinic) time-stepping. To avoid undesirable damping of resolved frequencies, the fast-time averaging is performed using a specially designed S-shaped filter function (denoted by $\langle\, .\,\rangle$ in Fig. 1) that has second-order temporal accuracy for the averaged barotropic prognostic variables, $\langle \zeta, \overline{u}, \overline{v} \rangle$. (A strictly positive-definite averaging yields at most first-order accuracy.)

- *Tracer Conservation and Constancy-Preservation:* To assure these properties when the grid-box control volumes change due to changes ζ, one must ensure that slow-time volume fluxes are exactly consistent with the changes in ζ as computed with the barotropic mode. Hence, it is not enough to know the final state of $\langle\, .\,\rangle$-averaged barotropic variables at the new time step; one also needs to have an integral measure of the entire barotropic evolution between two consecutive baroclinic times. This is accomplished by fast-time averaging the barotropic volume flux using a second operator ($\langle\!\langle\, .\,\rangle\!\rangle$ in Fig. 1) that is derived from the primary $\langle\, .\,\rangle$ (Sec. 3.1).

- *Barotropic Time-Stepping:* Since the external mode phase speed imposes the dominant CFL restriction on Δt, the generalized forward-backward step is chosen for barotropic mode. This algorithm consists of a modified Adams-Bashforth update of free surface followed by update of momentum equations where the newly computed ζ participates in the computation of PGF. Unlike the classical forward-backward step, the new algorithm naturally combines with advection and Coriolis terms and has a dissipative leading-order truncation term.

- *Baroclinic Time-Stepping:* 3D time-stepping schemes are designed in anticipation of different Courant-number limitations corresponding to different physical processes. The internal gravity-wave speed is expected to be the most restrictive, although the other limits — advective and Coriolis CFL — are not as distant as in the barotropic mode. A modified predictor-corrector scheme with forward-backward feedback between advection of ρ (via the tracers T and S) and PGF in momentum equations is chosen. It generally maintains temporal third-order accuracy for advection and Coriolis terms to match the accuracy of spatial discretization. The use of forward-backward feedback expands the CFL stability limit for internal waves. A forward Euler step is used for horizontal viscosity and diffusion terms, and an implicit backward step is used for vertical mixing. The overall time-stepping procedure is compatible

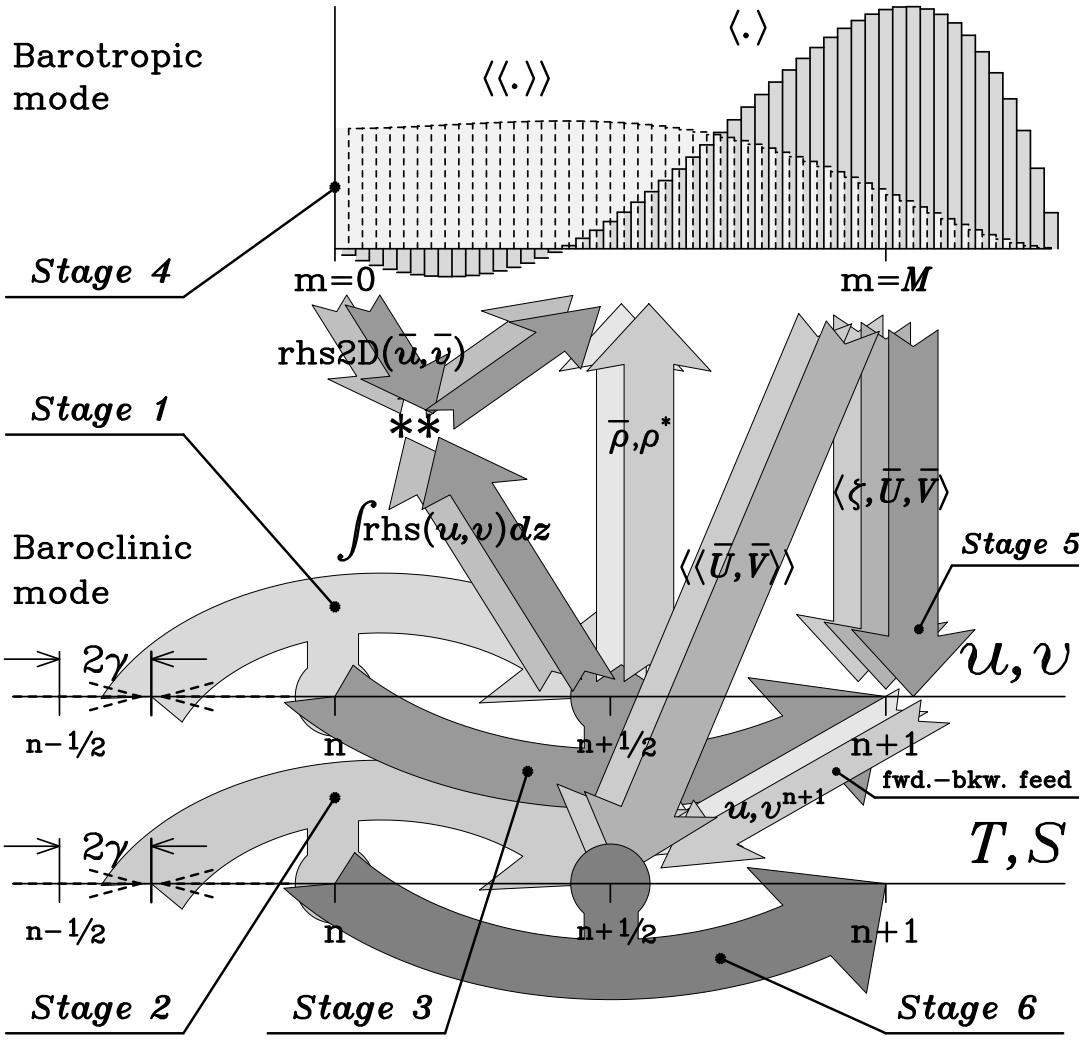


Fig. 1. Schematic diagram of main time stepping procedure of ROMS hydrodynamic kernel using Leap-Frog – third-order Adams-Moulton (LF–AM3) predictor-corrector step for the baroclinic (3D) mode with mode coupling during the corrector stage. The **arcs** (curved arrows) represent "steps", *i.e.,* updates of either momenta or tracers that involve computation of r.h.s. terms (shown as circles attached to the arcs). **Straight arrows** indicate exchange of data between the modes. Each arrow originates at the time when the corresponding variable becomes logically available, regardless of its actual temporal placement. Arcs and arrows are drawn in the sequence that matches the sequence of operations in the actual code: whenever arrows overlap, the operation occurring later corresponds to the arc or arrow on top. Note that labels *Stage 1 ... Stage 6* correspond to the actual computational stages described in Sec. 5 of Shchepetkin & McWilliams (2005). The four ascending arrows denote the vertically integrated r.h.s. terms for 3D momentum equations; and the 2-way, vertically averaged densities, $\overline{\rho}$ and ρ^* which participate in computation of pressure gradient terms for the barotropic mode (Sec. 3.2 below). The two descending arrows of smaller size on the left symbolize r.h.s. terms computed from barotropic variables. The asterisks (*** ***) where the two pairs of ascending and descending arrows meet denote the computation of baroclinic-to-barotropic forcing terms, two smaller arrows ascending diagonally to the right. The five large descending arrows symbolize 2-way fast-time-averaged barotropic variables (enclosed in $\langle . \rangle$ and ($\langle\langle . \rangle\rangle$, Sec. 3.1 below) for backward coupling; **fwd.-bkw. feed** stands for *forward-backward feedback* between momentum and tracer equations – the update of tracers is delayed until the new-time-step velocities u, v^{n+1} become available, so that they can participate in computation of r.h.s. terms for tracers; M is mode splitting ratio [number of barotropic time steps per one baroclinic. Note that barotropic time stepping goes slightly beyond ($\sim 25\%$ in the case above) the baroclinic step $n+1$]; $\gamma = 1/12$ is associated with of LF–AM3 algorithm (this is further explained in Fig. 17 in Sec. 4).

with both centered and upstream-biased advection, which is important for ROMS where we commonly use a third-order, upstream-biased advection scheme in the horizontal directions for both tracer and momentum, but a centered scheme in the vertical to avoid spurious diffusion due to "rectification" of dissipative truncation terms.

- *Temporal Stability Limits:* The time-stepping algorithms are specifically designed for use close to their limiting Courant number for computational stability yet still guarantee a numerically accurate solution. The optimal algorithms are derived by an inverse stability analysis, by writing them with arbitrary coefficients first, then deriving characteristic equations and choosing coefficients that yield the desirable characteristic roots. This makes it possible to resolve the phase propagation for both internal and external modes with an accuracy order higher than for each equation taken individually.

- *Updating:* ROMS's time-stepping utilizes a form where all temporal interpolations are applied to the primitive variables rather than their right-hand-side (r.h.s.) tendencies. This allows us to combine different time-stepping algorithms for different physical terms and reduces memory usage for a more efficient code.

- *Baroclinic PGF:* This term is discretized with a high-order, density-Jacobian scheme based on monotonized cubic polynomial fits for the vertical profiles of ρ and geopotential height z. This scheme preserves most of symmetries of the original Jacobian of Blumberg & Mellor (1987) while dramatically reducing errors in hydrostatic balance.

- *Compressible Equation of State (EOS):* Because of seawater's compressibility most of the vertical change of *in situ* ρ is due to pressure change. Monotonicity of *in situ* ρ does not guarantee the absence of spurious oscillations in the interpolated stratification profile; this degrades the accuracy of the PGF scheme and potentially leads to numerical instability. Furthermore, the combination of the Boussinesq approximation and the full EOS is a source of both inaccuracy and mode-splitting error. Therefore, the EOS (Jackett & McDougall, 1995) is modified to cancel the bulk compressibilty in *in situ* ρ to achieve a more consistent Boussinesq approximation (Dukowicz, 2001) and reformulated in terms of adiabatic ρ derivatives.

- *Advection:* ROMS commonly uses a third-order upstream-biased advection in the horizontal direction for both tracer and momentum equations and fourth-order centered advection in the vertical.

- *Coriolis and Curvilinear Metric Terms:* These are combined with advection of momentum and discretized using an energy-conserving scheme.

- *Code Architecture:* The code architecture is distinct from a modular design (*cf.*, MOM). The architectural design decisions involve optimization in multidimensional space for the model physics, numerical algorithms, and computational performance. As a rule, this results in significantly larger functional units in the code than in more traditional oceanic modeling practice. This is typically beneficial for both exploiting cache locality and minimizing the number of synchronization events in a parallel code.

- *Parallelization:* ROMS is a parallel code which has both shared- (via Open MP) and distributed-memory (via MPI) capabilities, including a possibility of allowing multiple threads within each MPI process. Both Open MP and MPI options are implemented using two-dimensional subdomain decomposition in horizontal directions.

A detailed description of the components and algorithms of ROMS is outside the scope of this article. Instead, we present a comprehensive overview of the kernel algorithms, focusing on algorithm interferences that require special effort to reconcile conflicts so that multiple desired properties can coexist at the

same time. Examples of such conflicts are (*i*) the barotropic-baroclinic time-splitting scheme (as diagnosed by a linear stability analysis) can interfere with finite-volume mass conservation in slow mode, as well as cause loss of the tracer constancy-preservation property; (*ii*) linear stability analysis favors Forward-Backward time-stepping for momentum and tracers over predictor-corrector by the stability *vs.* computational cost criterion for internal waves alone, but most suitable advection algorithms are two-stage procedures that are more naturally incorporated into a predictor-corrector scheme; (*iii*) barotropic-baroclinic mode-splitting makes it impossible to satisfy the finite-volume continuity equation on slow baroclinic time during a predictor sub-step, causing loss of the constancy-preservation property for tracers; (*iv*) high-order polynomial interpolation requires monotonicity constraints to prevent spurious oscillations if the interpolated field is not smooth on the grid scale, and for ρ this leads to a monotonicity constraint for stratification that further leads to a redesign of the EOS for seawater; and (*v*) with modal time-splitting the barotropic time step requires knowledge of bottom stress related to bottom velocity that is a sum of both types of modes, yet it would be unphysical to remove more than the total momentum within the bottom-most grid box per baroclinic time step while the baroclinic velocity is held constant.

2 Time-Stepping: Accuracy and Linear Stability

Oceanic flows in a regime with high Reynolds number can usefully be viewed from the perspective of time-stepping algorithms as satisfying hyperbolic partial differential equations. We consider two simple hyperbolic test systems. One can be called an advection equation,

$$\frac{\partial q}{\partial t} + c\frac{\partial q}{\partial x} = 0\,, \tag{2.1}$$

and the other a wave system,

$$\frac{\partial \zeta}{\partial t} = -c\frac{\partial u}{\partial x}\,, \qquad\qquad \frac{\partial u}{\partial t} = -c\frac{\partial \zeta}{\partial x}\,. \tag{2.2}$$

Table 4 from Griffies *et al.* (2000) provides a comprehensive summary of time-stepping algorithms used in different oceanic models. These can be subdivided into two major classes. The first class is synchronous schemes where the r.h.s. tendencies for all prognostic variables are computed at the same time and simultaneously used to advance the variables to the next time step; examples are Leap-Frog (LF) with an Asselin Filter to suppress temporal oscillations, second-order Runge-Kutta (RK2), predictor-corrector (LF with a trapezoidal rule (LF–TR), LF with third-order Adams-Moulton (LF–AM3), second-order Adams-Bashforth with TR predictor-corrector (AB2–TR)), and third-order Adams-Bashforth (AB3) (Durran, 1991). The second class is Forward-Backward (FB) schemes, where one variable is advanced then immediately used to advance the other(s),

$$\zeta^{n+1} = \zeta^n - c\Delta t \cdot \frac{\partial u^n}{\partial x}\,, \qquad\qquad u^{n+1} = u^n - c\Delta t \cdot \frac{\partial \zeta^{n+1}}{\partial x}\,, \tag{2.3}$$

where n is a time index. A FB scheme obviously is applicable only to multi-variate systems. Almost all OGCMs currently use a synchronous method.

One can easily verify that synchronous time-stepping has identical accuracy and stability limits for the advection equation and wave system (Canuto *et al.*, 1988; Shchepetkin & McWilliams, 2005). This typically occurs for $\alpha_{\max} < 0.8$ (where $\alpha \equiv \omega\Delta t$ is the Courant number and $\omega = ck$ is the frequency

for a solution component with wavenumber k) per r.h.s. computation for the most efficient algorithms within this class. This is only half as efficient (as measured by the ratio of stability limit to the number of r.h.s. computations) as a FB scheme with $\alpha_{\max} = 2$. Thus, the commonly used synchronous time-stepping is less than optimal for oceanic modeling because the fastest process, gravity waves, occur as an interplay between momentum and mass as in the wave system. Therefore, we define our primary design goals as (*i*) to generalize the most-used synchronous algorithms (*i.e.*, RK2, LF–TR, LF–AM3, and AB3) by introducing a FB-like feedback, and (*ii*) to generalize FB to higher orders of accuracy. In both cases the time-stepping algorithm must be accurate and robust even if used close to the α limit for of numerical stability.

The methodology employed here is a von Neumann linear stability analysis (Durran, 1998) applied in an "inverse" manner to design the algorithm rather than to assess one chosen *a priori*. We insert adjustable parameters into a time-stepping algorithm, then derive the characteristic equation for the eigenvalues of the step-multiplier matrix, and then solve it as an optimization problem to find parameters that achieve the desired properties. These properties include the order of accuracy and related bandwidth of the resolved frequency spectrum that is accurately represented; the maximum stability limit; the nature of the dominant truncation error term (*n.b.*, dissipation of fastest, poorly resolved frequencies is preferred over dispersion); and sufficient damping for any computational modes. The method is applied to the spatial Fourier transform of (2.3),

$$\frac{\partial \zeta}{\partial t} = -i\omega \cdot u \,, \qquad \frac{\partial u}{\partial t} = -i\omega \cdot \zeta \,, \tag{2.4}$$

with $\omega = ck$. Although it is implicit here that the primitive system is nonlinear, the stability analysis is linear. For example, consider the evolution of a small perturbation to a nonlinear flow described by

$$\frac{\partial \zeta}{\partial t} + V\frac{\partial \zeta}{\partial x} = -c\frac{\partial u}{\partial x} \,; \qquad \frac{\partial u}{\partial t} + V\frac{\partial u}{\partial x} = -c\frac{\partial \zeta}{\partial x} \,, \tag{2.5}$$

where V is velocity of background flow. An instability of an algorithm applied to (2.5) would automatically imply an instability of the fully nonlinear system using the same algorithm. Thus, a practical time-stepping algorithm for (2.5) is always a combination of both a generalized FB step for terms involving the ζ-u interplay, as well as a synchronous algorithm for other terms where a FB step is either not applicable, or impractical. Although less critical in its CFL limitation, the synchronous step must be at least conditionally stable. A similar requirement comes from the need for stable treatment of advection and Coriolis force, and the latter is the more restrictive since robustly stable, dissipative, upstream-biased, advection schemes can be used.

2.1 A Two-time-level Scheme: RK2 with FB Feedback

Consider a discrete time-stepping algorithm for (2.4) with a predictor sub-step,

$$\begin{aligned} \zeta^{n+1,*} &= \zeta^n - i\alpha \cdot u^n \\ u^{n+1,*} &= u^n - i\alpha \cdot \left[\beta\zeta^{n+1,*} + (1-\beta)\zeta^n\right] \,, \end{aligned} \tag{2.6}$$

followed by a corrector sub-step,

$$\begin{aligned}\zeta^{n+1} &= \zeta^n - \frac{i\alpha}{2}\cdot\left(u^{n+1,*} + u^n\right)\\ u^{n+1} &= u^n - \frac{i\alpha}{2}\cdot\left[\epsilon\zeta^{n+1} + (1-\epsilon)\zeta^{n+1,*} + \zeta^n\right]\,.\end{aligned} \tag{2.7}$$

Setting $\beta = \epsilon = 0$ in the above reverts it to the standard RK2 time-stepping that is unstable for a non-dissipative system (purely real-valued α) since the eigenvalue magnitude is $|\lambda| = \sqrt{1+\alpha^4/4} \approx 1 + \alpha^4/8 > 1$, implying amplitude growth in time for any α. But, because in the limit $\alpha \to 0$ its growth rate asymptotes to unity faster than $1 + \mathcal{O}(\alpha^2)$, it is sufficient to add hyperdiffusivity, rather than normal diffusvity, to stabilize a forward-in-time, centered-in-space scheme. This behavior is called *weak* or *asymptotic* instability.

The presence of terms with β and ϵ brings FB feedback into the algorithm (2.6)-(2.7), and both the accuracy and stability can be improved by having them present. Using the r.h.s. of the predictor equations, we eliminate $\zeta^{n+1,*}$ and $u^{n+1,*}$ from the corrector and transform the algorithm into a single step written in matrix form as

$$\begin{pmatrix}\zeta\\ u\end{pmatrix}^{n+1} = \begin{pmatrix}1-\frac{\alpha^2}{2} & -i\alpha\left(1-\frac{\alpha^2\beta}{2}\right)\\ -i\alpha\left(1-\frac{\alpha^2\epsilon}{4}\right) & 1-\frac{\alpha^2}{2}+\frac{\alpha^4\beta\epsilon}{4}\end{pmatrix}\begin{pmatrix}\zeta\\ u\end{pmatrix}^{n}\,. \tag{2.8}$$

This yields the characteristic equation for $\lambda(\alpha)$,

$$\lambda^2 - \left(2 - \alpha^2 + \frac{\alpha^4\beta\epsilon}{4}\right)\lambda + 1 + \frac{\alpha^4}{4}\left(1 - 2\beta - \epsilon + \beta\epsilon\right) = 0\,. \tag{2.9}$$

Since the exact solution of (2.4) has $\lambda = e^{\pm i\alpha}$, corresponding to right- and left-traveling waves in (2.2), we substitute the desired solution into (2.9) and expand it in a Taylor series for small α, seeking to approximate the ideal step multiplier as accurately as possible by suppressing mismatch terms with successive powers of α:

$$\alpha^4\left(\frac{1}{3} - \frac{\beta}{2} - \frac{\epsilon}{4}\right) \pm i\alpha^5\left(\frac{1}{12} - \frac{\beta\epsilon}{4}\right) + \mathcal{O}\left(\alpha^6\right) = 0\,. \tag{2.10}$$

Choosing $\epsilon = 4/3 - 2\beta$ eliminates the $\mathcal{O}(\alpha^4)$ term, reducing the above to

$$\pm i\alpha^5\left[\frac{1}{36} + \frac{1}{2}\left(\beta - \frac{1}{3}\right)^2\right] + \mathcal{O}\left(\alpha^6\right) = 0\,. \tag{2.11}$$

No real-valued β can eliminate the $\mathcal{O}(\alpha^5)$ term, one can only minimize the residual by setting $\beta = 1/3$, and, correspondingly, $\epsilon = 2/3$. The position of characteristic roots relative to the unit circle (*i.e.*, the exact solution) is shown in Fig. 2.

The stability range of this algorithm is limited by one of the modes leaving the unit circle through $\lambda = -1$. Substituting $\lambda = -1$ and $\epsilon = 4/3 - 2\beta$ into (2.9) yields

$$4 - \alpha^2 + \left[\frac{1}{36} - \left(\beta - \frac{1}{3}\right)^2\right]\alpha^4 = 0\,, \tag{2.12}$$

which is to be solved for $\alpha = \alpha(\beta)$ with β playing the role of an independent parameter. A simple analysis leads to the conclusion that $\beta = 1/3$ yields the maximum α ($= 2.14093$), hence the largest

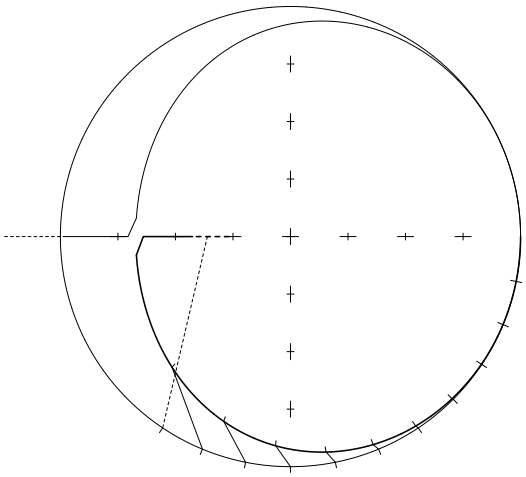


Fig. 2. Characteristic roots for the modified RK2 scheme (2.6),(2.7) with $\beta = 1/3$, $\epsilon = 2/3$ relative to the unit circle. Tickmarks on the outer side of the unit circle point to the locations of "ideal" amplification factors $e^{-i\alpha}$ for $\alpha \in \{-\pi/16, -\pi/8, -3\pi/16 ...\}$. Tickmarks on the inner side of bold solid curve indicate the actual roots corresponding to these values of α. The ideal and the actual root locations are connected by a thin straight line whose length and orientation show the magnitude and the nature (dispersive/dissipative) of numerical error. This algorithm has a third-order accurate step- multiplier $\lambda = \lambda(\alpha)$ and a stability limit $\alpha_{\max} = \sqrt{6\left(3-\sqrt{5}\right)} = 2.14093$.

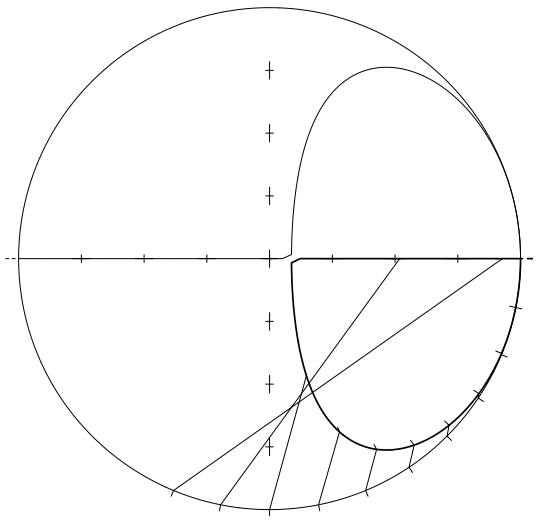


Fig. 3. Same as Fig. 2, but for $\beta = 1/2$ and $\epsilon = 1$. This setting is similar to Hallberg (1997).

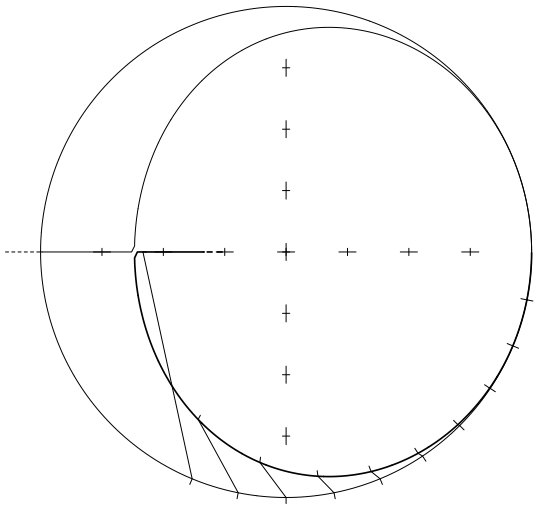


Fig. 4. Rueda *et al.* (2007) algorithm with their $\theta_{\rm p} = 1/6$, $\theta_{\rm b} = \theta = 1/2$. This is equivalent to our Eqs. (2.6),(2.13) with $\gamma = \theta = 1/2$, $\beta = 1/6$.

possible stability limit, and, as shown in (2.11) and the next paragraph, the same β value corresponds to the minimum possible truncation error among the whole subset of third-order schemes.

Overall, this modified RK2 algorithm is in line with two-time-level schemes of Hallberg (1997); Higdon (2002), except that they do not contain any counterpart for the free parameter ϵ in (2.7) by always selecting $\epsilon = 1$ (hence their algorithms cannot be reverted back to classical RK2). Setting $\beta = 0$, $\epsilon = 1$ in (2.6)-(2.7) yields a non-dissipative scheme that makes (2.9) identical to the characteristic equation for classical FB. This leads to second-order accuracy and $\alpha_{\max} = 2$. In the absence of Coriolis force, the algorithm in Eqs. (4)-(6) of Higdon (2002) has an identical characteristic equation, eigenvalues, accuracy, and stability limit. Setting $\beta = 1/2$, $\epsilon = 1$ yields another second-order algorithm which is similar to Eq. (16) of Higdon (2002) (*cf.*, Eqs. (3.9), (3.10), (3.13), and (3.14) of Hallberg (1997)). Again the stability limit is $\alpha_{\max} = 2$, but the scheme is highly dissipative, Fig. 3 [2].

Rueda *et al.* (2007) considered a family of RK2-type algorithms for the baroclinic mode of the TRIM

[2] The algorithms of Higdon (2002) and Hallberg (1997) can be viewed as the two extreme members of the β-family of second-order schemes (2.6)-(2.7) with $\epsilon = 1$ and $\beta \in [0, 1/2]$. All of them have a stability limit $\alpha_{\max} = 2$ independently of β, and they differ only by the dissipation rate that increases with β.

model [3] . They combine the predictor step (2.6) with [4]

$$\begin{aligned} \zeta^{n+1} &= \zeta^n - i\alpha \cdot [\gamma u^{n+1,*} + (1-\gamma) u^n] \\ u^{n+1} &= u^n - i\alpha \cdot [\theta \zeta^{n+1} + (1-\theta)\zeta^n] \ , \end{aligned} \tag{2.13}$$

where again there is no ϵ-mixing between predicted and corrected ζ, but an extra degree of freedom is introduced by allowing γ and θ deviate from $\gamma = \theta = 1/2$. To be second-order accurate requires $\gamma + \theta = 1$. Once this is satisfied, an additional constraint, $\beta\gamma = 1/12$, makes this algorithm third-order accurate. Rueda *et al.* (2007) restricted their analysis to a set of discrete values with $\theta = 1/2$ or 1 and β, γ are various permutations of 0, $1/2$, and 1, all of which result in either second- or first-order accuracy. They also showed that the choice of $\gamma = \theta = 1/2$, $\beta = 1/6$ results in a third-order accurate algorithm. By making an analysis similar to (2.12), one can also show that this choice yields the largest possible stability limit, $\alpha_{\max} = 2$: any deviation of γ, θ from $1/2$, while maintaining $\gamma + \theta = 1$ and $\beta\gamma = 1/12$ reduces $\alpha_{\max}$ relative to this value. Overall, it is comparable, though slightly more dissipative than, (2.6)–(2.7) with $\beta = 1/3$, $\epsilon = 2/3$(Fig. 4).

Despite the fact that two-time-level algorithms for shallow-water equations [5] are perhaps the most studied, (Hallberg, 1997; Higdon, 2002, 2005; Shchepetkin & McWilliams, 2005; Rueda *et al.*, 2007), have an extensive history, none of the previous work has produced a scheme which is competitive with the classical forward-backward step in terms of its stability limit relative to computational cost ($\alpha_{\max} = 2$ with the r.h.s. computed only once per time step for each equation). An examination the characteristic equations resulting from the two versions of a predictor–corrector algorithm — (2.6) in combination with either (2.7) or (2.13) — reveals that neither has sufficient degrees of freedom, despite the presence of three free coefficients in each. This can be remedied by combining ϵ- and θ-weightings for the corrector step so it becomes

$$\begin{aligned} \zeta^{n+1} &= \zeta^n - i\alpha \cdot [(1-\theta) u^{n+1,*} + \theta u^n] \\ u^{n+1} &= u^n - i\alpha \cdot [\theta \left(\epsilon \zeta^{n+1} + (1-\epsilon)\zeta^{n+1,*}\right) + (1-\theta)\zeta^n] \ , \end{aligned} \tag{2.14}$$

where we already replaced γ by $1-\theta$ in (2.13) to make it second-order accurate. As expected, the characteristic equation for (2.6), (2.14) is

$$\lambda^2 - \lambda \left[2 - \alpha^2 + \alpha^4 A\right] + 1 - \alpha^4 B = 0 \qquad \text{where} \quad \begin{cases} A = \beta\epsilon\theta(1-\theta) \\ B = (1-\theta)(\beta - \beta\epsilon\theta + \epsilon\theta - \theta) \end{cases} , \tag{2.15}$$

and it reverts back to (2.9) if $\theta = 1/2$ [6] . Substitution of $\lambda = e^{\pm i\alpha}$ and Taylor-series expansion leads to

[3] TRIM (tidal, residual, inter-tidal, mudflat) is an ocean model, whose emphasis is on fine-scale coastal dynamics and coastal engineering (Casulli & Cheng, 1992).

[4] Eqs. (2.6)–(2.13) can be remapped onto Rueda's Eqs. (50), (51), (52), and (25) using the following $our \rightarrow their$ substitute of variables: $\zeta \rightarrow u$; $u \rightarrow \rho, p$; $\beta \rightarrow \theta_{\rm p}$; $\gamma \rightarrow \theta_{\rm b}$; $\theta \rightarrow \theta$.

[5] In its classical sense the term *shallow-water equations* refers to a single-layer of shallow, hydrostatically balanced homogeneous fluid. After Casulli & Cheng (1992) and Casulli & Cattani (1994), it is frequently applied to hydrostatically balanced, stratified fluids, including ones admitting internal waves. Loosely, it is also applicable to governing equations for stratified, multilayer modeling in isopycnic coordinates.

[6] After setting $\epsilon = 1$ in (2.15), this also coincides with Eq. (53) from Rueda *et al.* (2007) if $\theta_{\rm b}$ is replaced with $1-\theta$

$$\alpha^4\left(\frac{1}{12}-A-B\right)\pm i\alpha^5 B+\alpha^6\left(\frac{B}{2}-\frac{1}{360}\right)+\mathcal{O}\left(\alpha^7\right)=0\,, \tag{2.16}$$

where the absence of a α^3-term guarantees second-order accuracy for any combination of β, θ, and ϵ. Obviously, one cannot eliminate both $\mathcal{O}\left(\alpha^5\right)$ and $\mathcal{O}\left(\alpha^6\right)$ terms simultaneously. To achieve third-order accuracy one needs to satisfy $A+B=1/12$, which leads to the condition

$$\epsilon=1+\frac{1}{12\theta(1-\theta)}-\frac{\beta}{\theta}\,, \tag{2.17}$$

and turns A and B in (2.15) into

$$\left.\begin{aligned} A&=(1-\theta)\left(C^2-(\beta-C)^2\right)\\ B&=(1-\theta)\left((\beta-C)^2-C^2+\frac{1}{12(1-\theta)}\right)\end{aligned}\right\}\quad\text{where}\quad C=\frac{\theta}{2}+\frac{1}{24(1-\theta)}\,. \tag{2.18}$$

The expression for B can be made equal to zero to eliminate $\mathcal{O}\left(\alpha^5\right)$ term in 2.16) only when $\theta>0.945$ [7], resulting in a non-dissipative fourth-order algorithm; however, it has unattractive properties: a significant portion of the α-range within the limit of stability yields a wrong phase

speed without providing any damping at all, and the coefficients in (2.14) are no longer non-negative because values of (θ,β) which make $B=0$ also result in $\epsilon>1$ as follows from (2.17); *e.g.*, $\theta>0.945$ yields ($\beta=1.230$, $\epsilon=1.302$). For $0\le\epsilon\le 1$, it is only possible to minimize the dissipation by selecting

$$\beta=\frac{\theta}{2}+\frac{1}{24(1-\theta)} \tag{2.19}$$

for any θ, which is still treated as a free parameter.

Algorithms of this kind become unstable when the two modes meet at some point on the real axis, after which one of them leaves the unit circle through either $\lambda=-1$ or $\lambda=+1$, whichever occurs earlier in α. Substituting $\lambda=\pm1$ into (2.15) yields

$$\begin{aligned}\lambda=-1:&\qquad 4-\alpha^2+\alpha^4(A-B)=0\\ \lambda=+1:&\qquad \alpha^2\left[1-\alpha^2(A+B)\right]=0\,.\end{aligned} \tag{2.20}$$

The first line results in $\alpha^2_{\max}=\left(1\pm\sqrt{1+16(A-B)}\right)\Big/[2(A-B)]$ where the sign $\pm$ must be chosen to be the same as the sign of $(A-B)$. The solution exists only if $A-B<1/16$. As $A-B\to 1/16$, then $\alpha_{\max}\to\sqrt{8}$, which is the largest stability limit when this limitation applies. (Note that $\alpha_{\max}=2$ in the case of $A-B=0$, and changes continuously when $A-B$ changes sign.) The second line in (2.20) yields $\alpha^2_{\max}=1/(A+B)$, which with (2.17) leads to a less restrictive $\alpha_{\max}=\sqrt{12}$ for the entire subset of third-order algorithms. Fig. 5 summarizes this for the space of parameters θ,β,ϵ within the domain to

there.

[7] The minimum possible value of θ which makes $B=0$ in (2.18) occurs when $\beta=C$ (hence eliminating the first quadratic term in the expression for B) and $C^2=1/[24(1-\theta)]$, which, after substitution of the expression for C yields a quaric equation for θ alone. Its only solution within the range of interest, $0<\theta<1$, is $\theta=0.9452697779$. Any change in β relative to $\beta=C$ results in a larger value of θ.

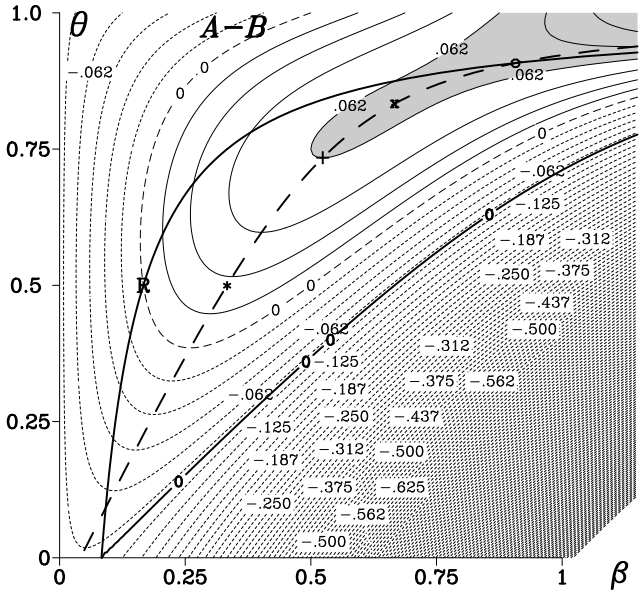


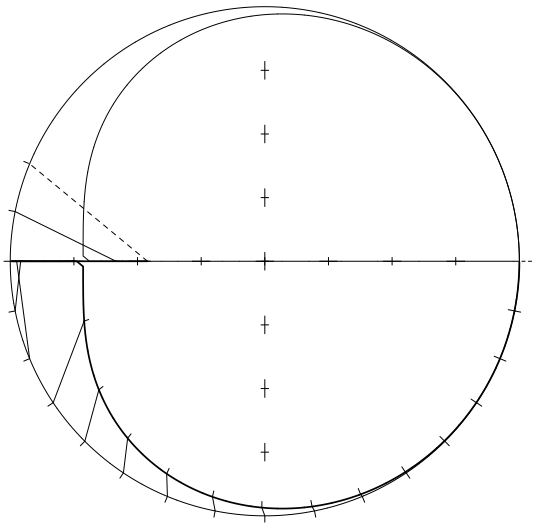
Fig. 5. Stability map for the two-parameter (β, θ)-family of third-order RK2 algorithms (2.6),(2.14) with ϵ set to satisfy (2.17). Thin contours show the difference of $A - B$ from (2.18) as a function of (β, θ), which controls the stability limit due to one of the modes leaving the unit circle at $\lambda = -1$. The shaded area corresponds to $A - B > 1/16$, where this no longer happens; hence the stability range is limited only by the mode leaving through $\lambda = +1$ resulting in $\alpha_{\max} = \sqrt{12} \approx 3.4641$ for all settings within the shaded area. Superimposed bold solid curves corresponds to $\epsilon = 0$ (lower) and $\epsilon = 1$ (upper): the values of (β, θ) must be chosen between these two curves in order for the algorithm to have all non-negative coefficients in (2.17). The bold dashed curve corresponds to a minimal dissipation subset with $\beta = \beta(\theta)$ from (2.19). Specific settings shown on this map are **R** (Rueda *et al.*, 2007) and $*$ (Shchepetkin & McWilliams (2005), also Fig. 2). The points **+**, **x**, and **o** refer to Fig. 6.

avoid negative coefficients in (2.14), $0 \le \theta, \epsilon \le 1$. In contrast, $\beta > 0$ can, in principle, exceed 1 because no coefficient like $1 - \beta$ is present in (2.6). This figure reveals the existence of an area where the stability is limited only by the lower line in (2.20); *i.e.,* none of the modes ever leaves the unit circle through $\lambda = -1$. We are therefore interested in (β, θ)-pairs from the portion of the shaded area in Fig. 5 just below the upper solid bold line that corresponds to $\epsilon = 1$. Furthermore, to minimize the $\mathcal{O}(\alpha^5)$ truncation term, we are interested in algorithms with β and θ related by (2.19), as represented by the bold dashed line on Fig. 5. Remarkably, this line follows the maximum of $A - B$ for any given θ, so that settings which minimize the truncation error are also optimal for stability.

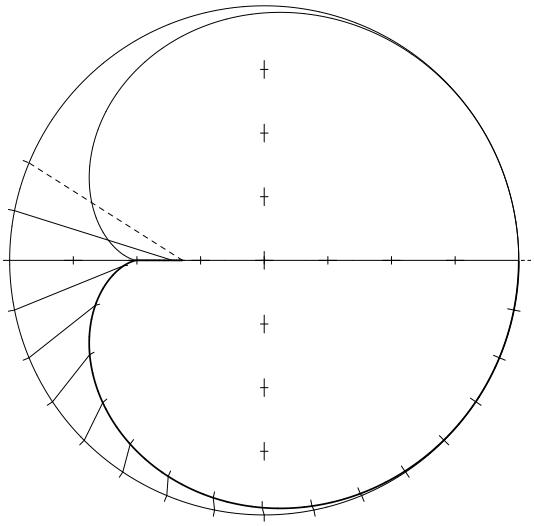
Characteristic roots for three algorithms from the shaded area are shown in Fig. 6. The one with $\theta = 0.734$ corresponds to just after entrance into the shaded area along the dashed line [8] (denoted as "+" on Fig. 5). The overall behavior of the algorithm is similar to that on Fig. 2 except that it has slightly lower dissipation. More importantly, after the two arms meet each other at $\lambda \approx -0.7$, one of them continues toward the negative real axis, but instead of exiting at $\lambda = -1$, it stops there, reverses direction, and continues toward the center. Note the existence of the stagnation point discussed in the caption. Setting θ slightly smaller that 0.734 causes this mode to exit at $\lambda = -1$.

Increasing θ beyond 0.734 while following the the dashed line on Fig. 5 moves the stagnation point toward the center of the unit circle, and subsequently it changes the behavior of the algorithm in the vicinity of the point where the two arms meet each other. For $\theta = 5/6$ (denoted as **x** on Fig. 5), they no longer approach the real axis at a 90-degree angle, but rather they bend inward and touch the real axis. The portion of the α-spectrum for which the roots λ are located on the real axis to the left of the merging point disappears when θ increases beyond $5/6$. This is beneficial for the algorithm because phase increments of α beyond π are within the aliasing range: wavenumber components corresponding to them cannot be propagated along the grid, so if the algorithm is used in this regime, these signals must be damped. Further increase of θ changes this behavior again. Instead of approaching the real axis, the arms bend inward,

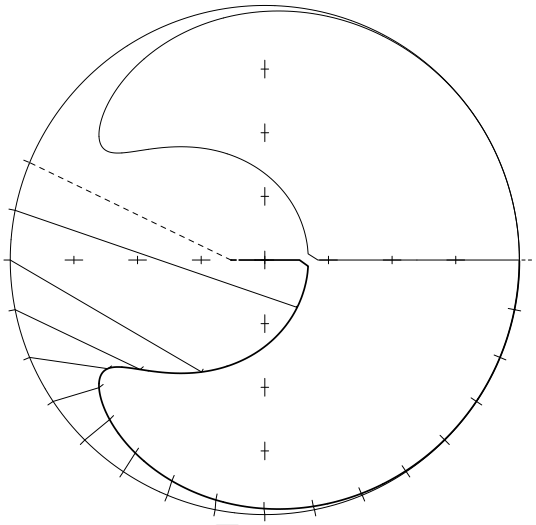
[8] The exact value of θ for the point of entry into the shaded area on Fig. 5 comes from the equation, $\left(\theta^2 - \frac{1}{6}\right)(1-\theta)^2 - \frac{1-\theta}{8} + \frac{1}{144} = 0$, derived by substituting expressions for A, B, C from (2.18) along with the condition $\beta = C$ from (2.19) into $A - B = 1/16$. This yields $\theta = 0.7332939955221$.



$\theta = 0.734$, $\beta = 0.523641604$, $\epsilon = 0.71340818$



$\theta = 5/6$, $\beta = 2/3$, $\epsilon = 4/5$



$$\theta = \beta = \frac{1}{2} + \sqrt{\frac{1}{6}} \approx 0.9082482, \quad \epsilon = 1$$

Fig. 6. Characteristic roots for the RK2 algorithm (2.6),(2.14) with coefficients chosen to yield third-order accuracy and minimal dissipation (*i.e.*, both conditions (2.17) and (2.19) are met) for three different values of θ. In the case of $\theta = 0.734$ the two arms meet each other at $\lambda \approx -0.7$, after which one of them proceeds along the real axis toward $\lambda = -1$, but stops when nearly reaching this point and reverses direction, continuing toward the center. (Note that roots corresponding to $\alpha = 7\pi/8$ and $\alpha = 15\pi/16$ are very close to each other, which indicates the existence of a stagnation point for $\lambda = \lambda(\alpha)$ in the vicinity (smaller values of θ result in one of the arms exiting the circle at $\lambda = -1$, as it occurs in Fig. 2, while larger θs move the reversal point closer to the center). Stability is limited by the other arm reaching $\lambda = +1$ at $\alpha = \alpha_{\max} = \sqrt{12}$, which is also the stability limit for the other two panels in this figure.

resulting in a highly dissipative algorithm for the upper portion of the spectrum, $13\pi/16 \le \alpha < \sqrt{12}$. Fig. 6, lower left, shows the characteristic roots for an algorithm with maximum possible β, θ along the minimal dissipation curve with all-non-negative coefficients in (2.6),(2.14) (this is the point **o** on Fig. 5, located at intersection of the bold dashed and solid $\epsilon = 1$ lines) [9] . Variation of θ within the range $0.734 \le \theta \le 0.91$ causes only minor effects on the behavior of this algorithm within the lower, physically accurate, portion of its spectrum, $|\alpha| < \pi/2$. All three examples on Fig. 6 demonstrate very small numerical dispersion and a dissipation-dominant truncation error outside this range.

This class of time-stepping algorithms is an attractive choice for isopycnic and high-resolution coastal engineering models because it is a two-time-level scheme that combines nicely with positive-definite advection algorithms as well as with wetting-and-drying schemes that also require the use of limiters. Having all non-negative coefficients in front of the r.h.s. terms in a time-stepping scheme is crucial (Stelling &

[9] Since this choice belongs to $\epsilon = 1$ – family, it can be used without modification in the TRIM code (Rueda *et al.*, 2007), except for setting coefficients $\theta, \theta_{\rm p} = 1/2 + \sqrt{1/6}$ and $\theta_{\rm b} = 1/2 - \sqrt{1/6}$ in their Eqs. (51), (52), and (25).

Duinmeijer, 2003). Its accuracy, stability, and efficiency are superior to most of the known algorithms. It is somewhat less attractive for z- or σ-coordinate models in the context of long-term, large-scale simulation because it is incompatible with centered vertical advection needed to avoid long-term drift: although this requirement is mitigated relative to forward-in-time stepping, some degree of upstream-biasing of advection schemes is required for stability if RK2-type time-stepping is used [10] . The existence of two-time level, predictor-corrector algorithms with a stability limit $\alpha_{\max}$ beyond 3 has been long overlooked, and, in fact, this makes it competitive with the FB-type algorithms considered later in this section in terms of computational efficiency (*i.e.,* the ratio of the stability limit to the number of r.h.s. computations for each equation).

2.2 LF–TR or LF–AM3 with FB Feedback

Another possibility is an algorithm comprised of a LF predictor sub-step followed by either a two-time TR or a three-time AM3 corrector:

$$\begin{aligned} \zeta^{n+1,*} &= \zeta^{n-1} - 2i\alpha \cdot u^n \\ u^{n+1,*} &= u^{n-1} - 2i\alpha \cdot \left[(1-2\beta)\,\zeta^n + \beta\,(\zeta^{n+1,*} + \zeta^{n-1}) \right] \end{aligned} \tag{2.21}$$

and

$$\begin{aligned} \zeta^{n+1} &= \zeta^n - i\alpha \cdot \left\{ \left(\frac{1}{2} - \gamma \right) u^{n+1,*} + \left(\frac{1}{2} + 2\gamma \right) u^n - \gamma u^{n-1} \right\} \\ u^{n+1} &= u^n - i\alpha \cdot \left\{ \left(\frac{1}{2} - \gamma \right) \left[\epsilon \zeta^{n+1} + (1-\epsilon)\zeta^{n+1,*} \right] + \left(\frac{1}{2} + 2\gamma \right) \zeta^n - \gamma \zeta^{n-1} \right\}, \end{aligned} \tag{2.22}$$

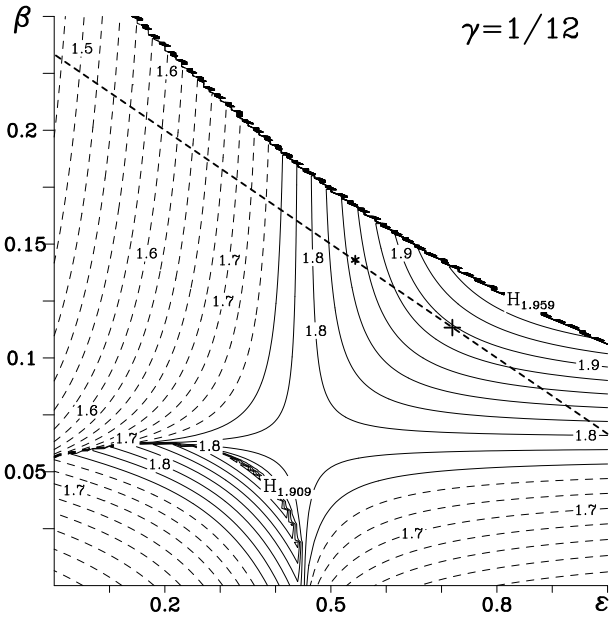
where the parameters β and ϵ introduce FB-feedback during both stages, while γ controls the type of corrector scheme. Without FB-feedback the standard algorithm is

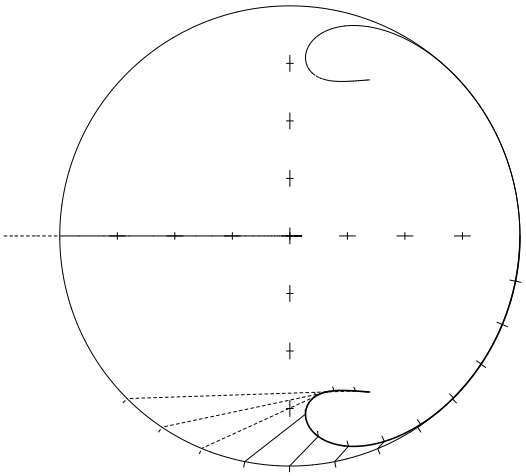
$$\beta = \epsilon = 0 \quad \Rightarrow \begin{cases} \gamma = 0 & \Rightarrow \text{LF–TR} & \alpha_{\max} = \sqrt{2} \\ \gamma = 1/12 & \Rightarrow \text{LF–AM3} & \alpha_{\max} = 1.5874 \\ \gamma = 0.0804 & \Rightarrow \text{max stability} & \alpha_{\max} = 1.5876\,, \end{cases}$$

which is one of the most efficient and attractive synchronous algorithms (*cf.,* Fig. 20 in Shchepetkin & McWilliams (2005)).

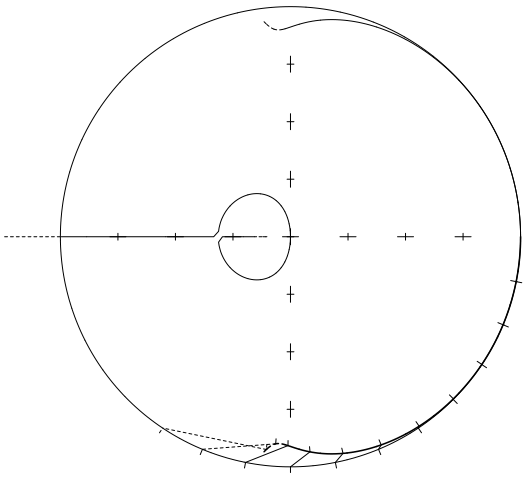
Following exactly the same path as for RK2 above, we derive a set of constraints for coefficients β, γ, and ϵ to achieve the specified orders of accuracy (Shchepetkin & McWilliams, 2005):

[10] For example, QUICK advection is asymptotically unstable in combination with forward-in-time stepping. (In contrast, QUICKEST, which explicitly contains the second-order, time-dependent terms is stable.) However, as discussed in Rueda *et al.* (2007), QUICK is stable in combination with RK2, while centered scheme are asymptotically unstable).

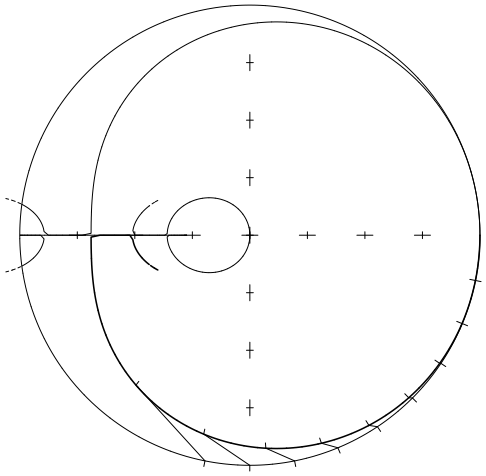




β=0.126, ϵ=0.83: maximum possible stability range ($\alpha_{\max}$=1.958537).

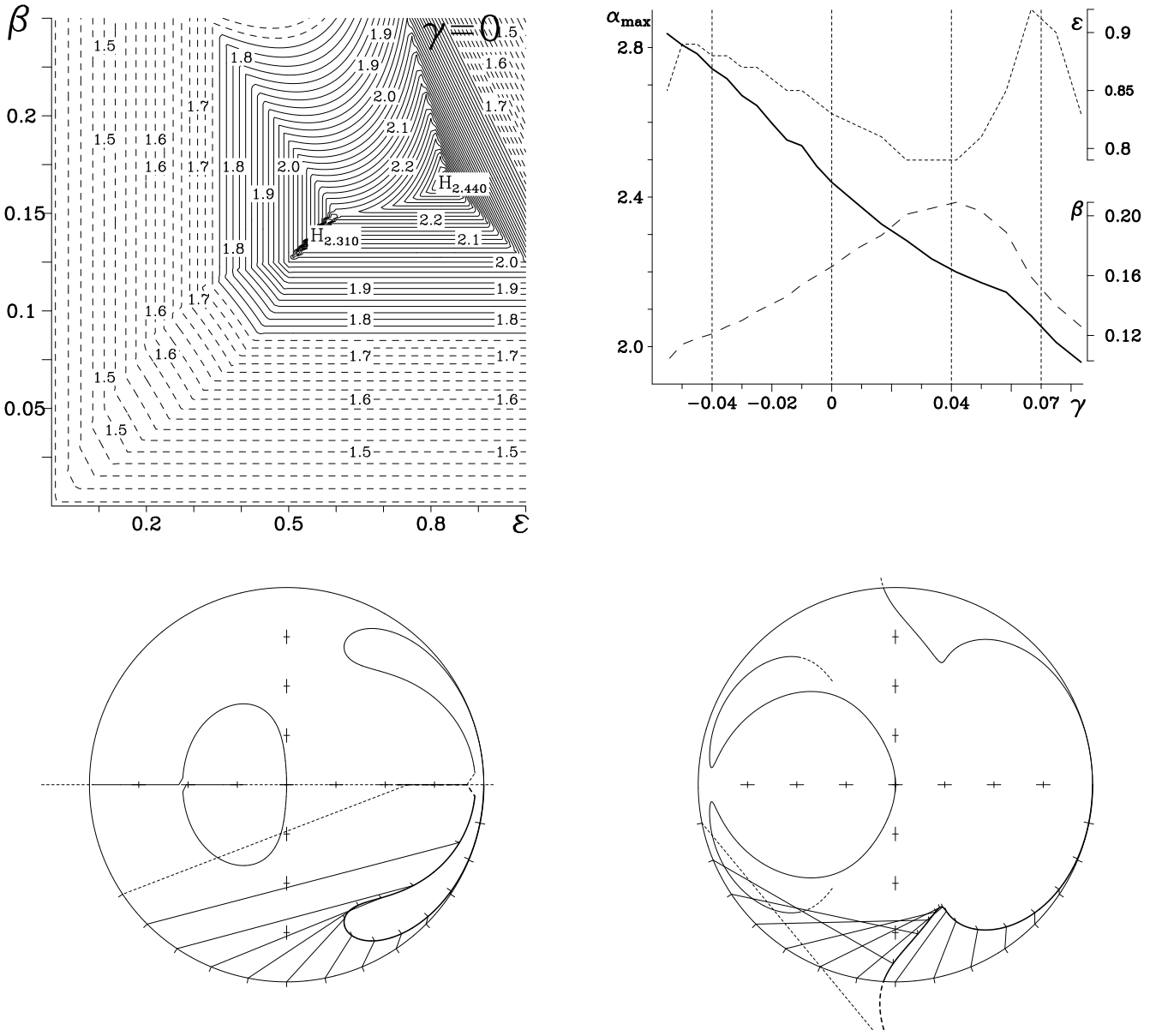


$\beta = 17/120$, $\epsilon = 11/20$: fourth-order accuracy with minimum possible truncation error ($\alpha_{\max}$=1.851640).



β=0.044, ϵ=0.39: secondary stability maximum ($\alpha_{\max}$=1.908525).

Fig. 7. *Upper left*: stability limit $\alpha_{\max}$ as a function of ϵ, β with $\gamma = 1/12$ (*i.e.,* among all third-order accurate schemes within the generalized LF–AM3 family). The empty area in the upper-right corner corresponds to schemes with an asymptotic instability for the physical modes. The straight dashed line $\beta = 7/30 - \epsilon/6$ approximately parallel to the edge corresponds to zero $\mathcal{O}\left(\alpha^5\right)$ truncation term (*i.e.,* a fourth-order accurate subset). The asterisk (*) and cross (+) on this line denote locations of the minimal possible truncation error and maximum stability limit among the fourth-order algorithms, which are not far away from each other. Note the stability maxima at (ϵ, β)=(0.83,0.126), just on the edge of asymptotic instability and (0.39,0.044). The three remaining panels show the characteristic roots for the β, ϵ choices yielding the indicated specific properties.



$\gamma = 0$, $\beta = 0.166$, $\epsilon = 0.84$: $\alpha_{\max}$=2.4114. $\gamma = -0.05$, $\beta = 0.105$, $\epsilon = 0.84$: $\alpha_{\max} = 2.8010$.

Fig. 8. *Upper left*: Map of $\alpha_{\max} = \alpha_{\max}(\epsilon, \beta)$ for γ=0. *Upper right*: $\alpha_{\max}$ and the corresponding ϵ, β as functions of γ. *Lower panels* show examples of β, ϵ choices that give the maximum stability range for a given γ.

$$\text{third-order:} \qquad \gamma = \frac{1}{12} \qquad \forall\ \beta, \epsilon \tag{2.23}$$

$$\text{fourth-order:} \qquad \text{above and } \ \beta = \frac{7}{30} - \frac{\epsilon}{6} \qquad \forall\ \epsilon \tag{2.24}$$

$$\text{fifth-order:} \qquad \text{both above and } \ -\frac{5}{6}\left(\epsilon - \frac{11}{20}\right)^2 - \frac{1603}{2400} = 0\,. \tag{2.25}$$

No set of coefficients can satisfy the condition for fifth-order accuracy, so we can only minimize the leading-order truncation term by choosing $\epsilon = 11/20$, hence $\beta = 17/120$ and $\gamma = 1/12$. This yields a fourth-order scheme with extremely small numerical dispersion and dissipation within the whole range of its numerical stability, $\alpha_{\max} = 1.851640$ (Fig. 7 lower-left panel).

Since a primary goal is to extend the stability range, we progressively give up one order of accuracy at

a time, which frees one or two parameters, ϵ, or (ϵ, β) be available for tuning. Fig. 7 (upper-left) shows a map of the stability range $\alpha_{\max}$ in an ϵ, β-plane, for all third-order accurate schemes (hence $\gamma = 1/12$ is always respected). The subset of fourth-order schemes is represented by the diagonal line, $\beta = 7/30 - \epsilon/6$, that is nearly parallel to the edge of stability. Overall, there are two stability maxima in the ϵ, β-plane, and remarkably, the choices corresponding to maximum stability are not far away from the minimal truncation error within the fourth-order subset. As a result, $\epsilon = 0.83$, $\beta = 0.126$, corresponds to the largest possible $\alpha_{\max} = 1.958537$, and it is also very accurate within the whole stability range (Fig. 7, upper-right). It has a 25% larger stability limit than the 1.5874 of the original LF–AM3 scheme with $\beta = \epsilon = 0$. The secondary maximum (lower-left) is less attractive and in fact, produces similar leading-order numerical dissipation and dispersion errors as does $\beta = \epsilon = 0$ LF–AM3, albeit with a wider stability range.

Searching for the maximum stability range in γ, β, ϵ-space while maintaining second-order accuracy (hence $\gamma \neq 1/12$ but is otherwise an adjustable parameter) requires essentially the same kind of analysis as in Fig. 7 (upper-left) but repeated for different values of γ. This is summarized in Fig. 8 (upper-right), with the upper-left panel showing a particular example of $\alpha_{\max} = \alpha_{\max}(\epsilon, \beta)$ for $\gamma = 0$. It turns out that the stability range can be expanded significantly with a decrease of γ, however, at the expense of accuracy degradation. Given that these schemes are dissipative, this is acceptable, and in fact desirable for the barotropic mode (since fast motions are fast-time-averaged anyway) and for applications where the wave propagation is not of primary interest. Thus, the introduction of FB-feedback into a LF–TR ($\gamma = 0$) scheme can achieve up to 70% gain in stability range relative to $\beta = \epsilon = 0$ (Fig. 8, lower-left). Going beyond $\gamma < 0$ is not desirable due to loss of accuracy. Still, none of these schemes can achieve an efficiency comparable to the classical FB scheme in terms of the ratio of $\alpha_{\max}$ and the number of r.h.s. computations.

2.3 Generalized FB with AB2–AM3

To approach the problem from the opposite direction — starting with a Forward-Backward scheme and attempting to construct an algorithm compatible with both advection and wave propagation — we consider an explicit algorithm comprised of an AB2-like step for ζ followed by an AM3-like step for u:

$$\begin{aligned} \zeta^{n+1} &= \zeta^n - i\alpha\left[(1+\beta)u^n - \beta u^{n-1}\right] \\ u^{n+1} &= u^n - i\alpha\left[(1-\gamma-\epsilon)\zeta^{n+1} + \gamma\zeta^n + \epsilon\zeta^{n-1}\right] . \end{aligned} \tag{2.26}$$

Obviously it reverts to the classical FB scheme if $\beta = \gamma = \epsilon = 0$. Its characteristic equation is

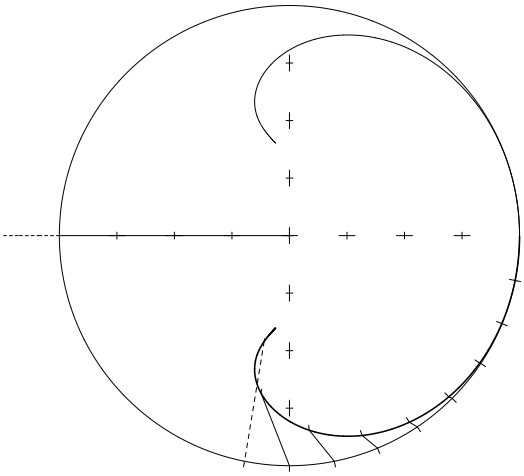
$$\begin{aligned} \lambda^2 - \left[2 - \alpha^2\left(1-\gamma-\epsilon\right)\left(1+\beta\right)\right]\lambda + 1 - \alpha^2\left(\beta - \gamma - 2\beta\gamma - \beta\epsilon\right) \\ + \alpha^2\left(\epsilon + \beta\epsilon - \beta\gamma\right)\lambda^{-1} - \alpha^2\beta\epsilon\lambda^{-2} = 0 . \end{aligned} \tag{2.27}$$

After substitution of $\lambda = e^{\pm i\alpha}$ and Taylor-series expansion in α, a set of constraints arise for achieving progressive orders of accuracy,

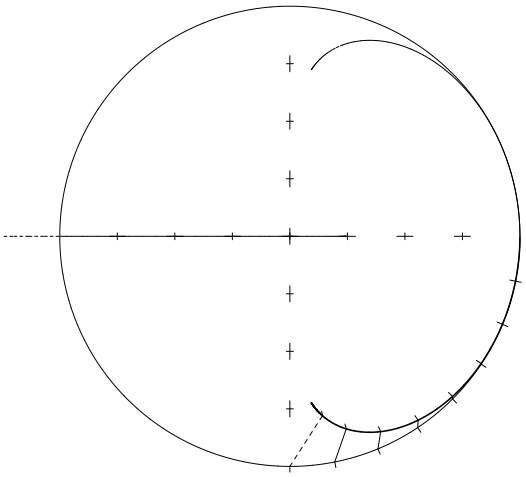
$$\text{second-order:} \qquad \gamma = \beta - 2\epsilon \qquad\qquad \forall\, \beta\,, \epsilon \tag{2.28}$$

$$\text{third-order:} \qquad \gamma = \beta - 2\beta^2 - \frac{1}{6} \quad \text{and} \quad \epsilon = \beta^2 + \frac{1}{12} \qquad \forall\, \beta \tag{2.29}$$

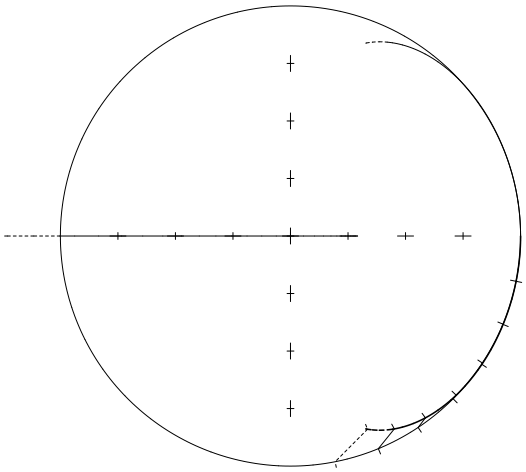
$$\text{fourth-order:} \qquad \gamma, \epsilon \quad \text{as above and} \quad \frac{1}{12} - \frac{\beta}{12} - \beta^3 = 0 \qquad \Rightarrow \quad \beta = 0.3737076\,. \tag{2.30}$$



β=0, $\gamma = -1/6$, $\epsilon = 1/12$ $\quad \alpha_{\max} = \sqrt{3}$



$\beta = 0.3737076$ (*fourth*-order) $\quad \alpha_{\max} = \sqrt{2}$



$\beta = 1/2$, $\gamma = -1/6$, $\epsilon = 1/3$ $\quad \alpha^*_{\max} = \sqrt{3/2}$

Fig. 9. Characteristic roots for the AB2–AM3 algorithm (2.26) with three different choices for β. In all three cases the remaining parameters γ and ϵ satisfy the third-order accuracy condition. The leading third-order, dissipative truncation term changes sign at $\beta = 0.3737076$, resulting in fourth-order accuracy. The scheme becomes weakly unstable beyond this point (note that physical modes on the left-lower panel are slightly outside the unit circle, reaching $|\lambda| \approx 1.01$ for $\alpha \approx \pi/3$). The stability range decreases with increasing β, and for $\beta =< 1/2$ the AB2-type time step is unconditionally unstable for an advection equation with centered spatial discretization.

The second-order accuracy condition can be interpreted as a time-centering balance rule: once the r.h.s. for ζ is placed at $t_n + (1/2 - \delta)\,\Delta t$, the r.h.s. for u is centered at $t_n + (1/2 + \delta)\,\Delta t$ with the same offset $\delta \equiv 1/2 - \beta$ from the midway time $t_n + \Delta t/2$. The classical FB scheme obeys this rule, and it is also respected by the third- and fourth-order constraints. The third-order condition introduces a single-parameter family of schemes with a useful range of $0 < \beta \leq 1/2$ (Fig. 9).

The leading-order truncation term has a dissipative character, and it decreases with increasing β. It vanishes at $\beta = 0.3737076$ when the scheme becomes fourth-order, and it changes sign thereafter; this means that the physical modes become asymptotically unstable beyond this β value. Leaving the weak asymptotic instability aside, the overall stability range is limited by one of the computational modes that leaves the unit circle at $\lambda = -1$, hence

$$\alpha_{\max} = \sqrt{3} \Big/ \sqrt{1 + \frac{\beta}{2} + 6\beta^3}\,, \tag{2.31}$$

which decreases with β. Although potentially attractive and simple, this algorithm does not combine nat-

urally with the other hyperbolic terms (advection, Coriolis) because there is no overlap in its β range: the AB2-like time step is asymptotically unstable for the advection equation when $\beta \leq 1/2$, while the algorithm (2.26) for the wave system needs $\beta \leq 0.3737076$, and in fact $\beta = 0$ is desirable to achieve the widest possible stability range.

2.4 Generalized FB with an AB3–AM4 Step

To overcome the limitation of (2.26), we explore the possibility of using a three-time, AB3-like step for ζ-equation followed by a four-time AM4-like step for u,

$$\begin{aligned} \zeta^{n+1} &= \zeta^n - i\alpha \left[\left(\tfrac{3}{2} + \beta\right) u^n - \left(\tfrac{1}{2} + 2\beta\right) u^{n-1} + \beta u^{n-2} \right] \\ u^{n+1} &= u^n - i\alpha \left[\left(\tfrac{1}{2} + \gamma + 2\epsilon\right) \zeta^{n+1} + \left(\tfrac{1}{2} - 2\gamma - 3\epsilon\right) \zeta^n + \gamma \zeta^{n-1} + \epsilon \zeta^{n-2} \right] , \end{aligned} \tag{2.32}$$

where the r.h.s. for both equations are already time-centered at $t_n + \Delta t/2$ regardless of the values for β, γ, and ϵ, (***i.e.***, the r.h.s. time-centering rule (2.28) for the AB2–AM3 scheme is already respected). As a result, second-order accuracy is always guaranteed. Overall, the AB3-type (β-family) time step for the advection equation is stable as long as $\beta > 1/6$ (otherwise it is subject to an asymptotic instability of an AB2-type), and it is third-order accurate if $\beta = 5/12$, while a smaller value of $\beta = 0.281105$ yields the largest stability range. This time step naturally combines with the Coriolis and advection terms (both centered and upstream-biased).

A viable choice would be a straightforward combination of third-order accurate AB3 (hence $\beta = 5/12$) with either a TR or a third-order accurate Adams-Moulton scheme ($\gamma = -1/12$, $\epsilon = 0$), resulting respectively in second- and third-order accuracy with a stability range $\alpha_{\max}$ slightly exceeding unity (Fig. 10). This about 50% more efficient than a synchronous third-order AB3 scheme for both equations ($\alpha_{\max} = 0.71$), but has only half the efficiency of the classical FB scheme. In the remaining part of this section we will show that the stability range of algorithm (2.32) can be significantly expanded by relaxing the condition $\beta = 5/12$, which is in fact the key to utilizing its full potential.

The analysis of the algorithm (2.32) follows the same path as for AB2–AM3 above. It again leads to a collection of conditions to achieve progressive orders of accuracy,

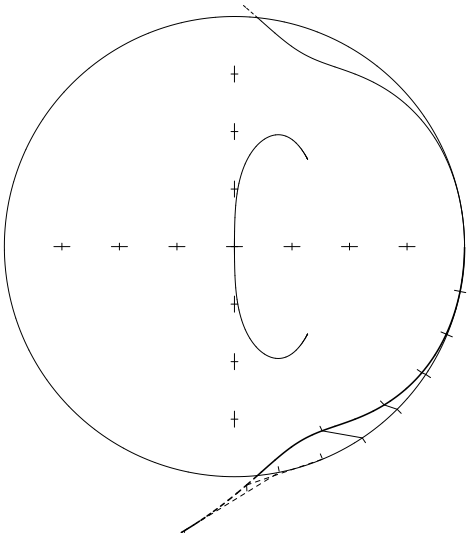
$$\text{third-order:} \quad \gamma = \frac{1}{3} - \beta - 3\epsilon \qquad\qquad \forall \beta, \epsilon \tag{2.33}$$

$$\text{fourth-order:} \quad \beta = \frac{1}{12} - \epsilon \quad \text{and} \quad \gamma = \frac{1}{4} - 2\epsilon \qquad\qquad \forall \epsilon \tag{2.34}$$

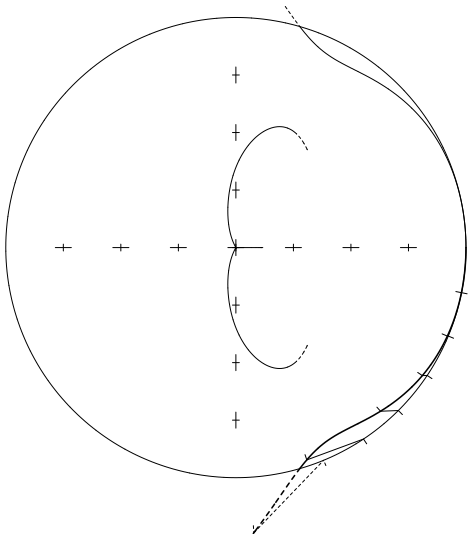
$$\text{fifth-order:} \quad \frac{7}{120} + \frac{2}{3}\epsilon + \epsilon^2 = 0 \qquad \Rightarrow \quad \epsilon = -\frac{1}{3} \pm \frac{\sqrt{190}}{60} ; \qquad \beta, \gamma \quad \text{from above} . \tag{2.35}$$

The fifth-order algorithm is asymptotically unstable and has $\alpha_{\max} = 1.0145$ limited by one of the computational modes leaving the unit circle at $\lambda = -1$ (Fig. 11, left). Overall this is not an attractive choice, due to both its modest stability range and the asymptotically instability of its physical modes.

Giving up one order of accuracy allows us to treat ϵ as an adjustable parameter that can be tuned to achieve the maximum stability range. This search yields $\epsilon = 1/12$ and a stability limit of $\alpha_{\max} = \sqrt{3}$ (Fig.11, right). (Here one can substitute $\beta = 0$, $\gamma = \epsilon = 1/12$ into the characteristic equation for (2.32) and verify that $\lambda = -1$ is a double root if $\alpha^2 = 3$.) An obvious drawback for this algorithm is that $\beta = 0$ means the time-stepping for the ζ-equation is only AB2, which is asymptotically unstable for advection and Coriolis force.

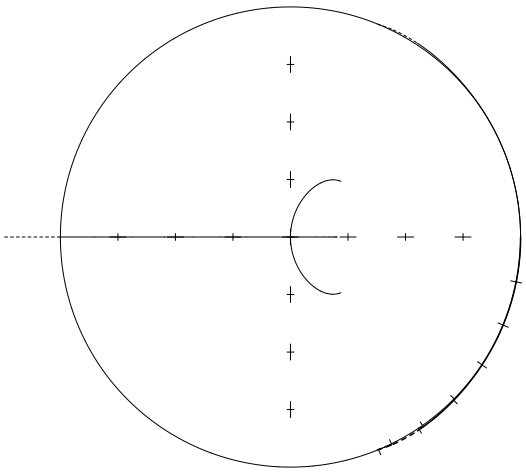


AB3–TR: β=5/12, γ=ϵ=0
$\alpha_{\max}$=1.1441551

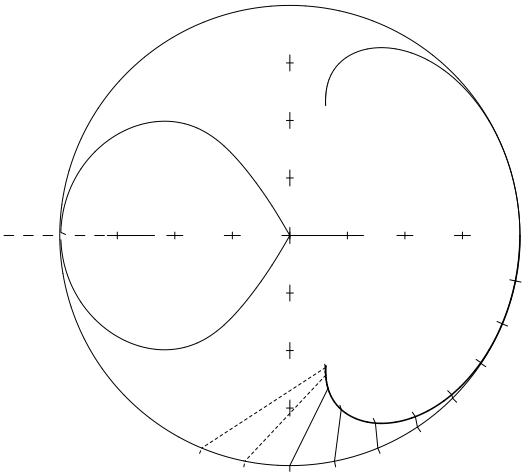


AB3–AM3: β=5/12, $\gamma = -1/12$, ϵ=0
$\alpha_{\max}$=1.003859

Fig. 10. The algorithm (2.32) with a third-order accurate AB3 (hence $\beta = 5/12$) first step. These "naive" settings result in a stability limit of order of unity. The algorithm on the left was the original version for the main time step in the ROMS family of codes, and it is still widely used.



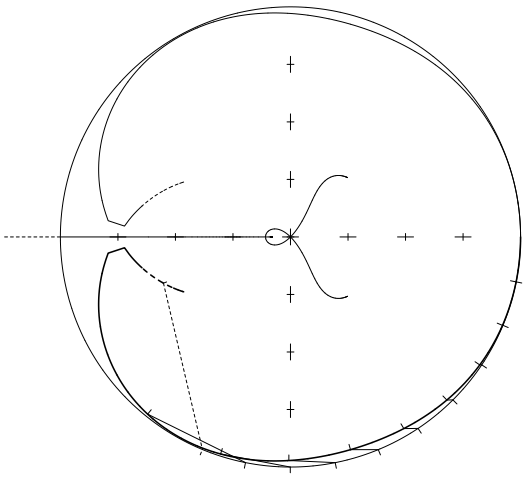
fifth-order accuracy with $\alpha_{\max}$=1.0145.



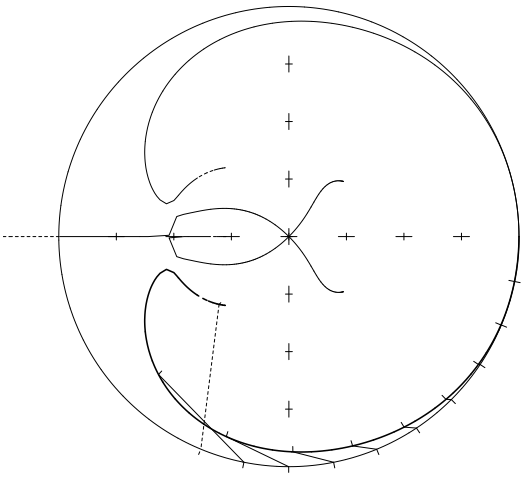
$\beta = 0 \quad \gamma = \epsilon = 1/12$: $\alpha_{\max} = \sqrt{3}$.

Fig. 11. Characteristic roots for the AB3–AM4 algorithm (2.32) with β, γ, ϵ set to achieve either fifth-order accuracy (*left panel*) or the maximum possible stability limit while maintaining fourth-order accuracy (*right panel*). Note that at the optimum ϵ, two computational modes meet at $\lambda = -1$, after which one of them continues out of the unit circle along the negative real axis. If $\epsilon > 1/12$ the meeting occurs outside the circle (*i.e.,* the computational modes leave the circle before they meet), while a smaller ϵ moves the meeting point inside, resulting in an earlier escape of one of the modes along the negative imaginary axis. Either way, $\alpha_{\max}$ ends up being smaller than $\sqrt{3}$ if $\epsilon \neq 1/12$.

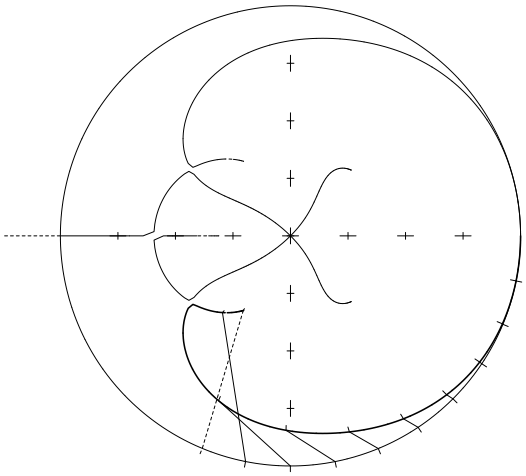
The third-order, 2-parameter (β,ϵ) family can reach up to $\alpha_{\max} = 1.939$ (Fig. 12, upper-left) that is now very close to that of the classical FB scheme. Its β value lies within the desirable range of $1/6 < \beta < 5/12$ (*i.e.,* the range of stable choices for the advection equation with spatially centered schemes, as well as for Coriolis force). The only undesirable property of this algorithm is its nearly purely dispersive truncation error, resulting in weak damping of frequency components that are not accurately represented. This issue can be addressed by a slight bias of β away from the maximum stability (Fig. 12, upper-right), which leads



β=0.232, ϵ=0.00525: maximum possible $\alpha_{\max}$ $\forall\beta,\epsilon$ ($\alpha_{\max}$=1.939).



β=0.21, ϵ=0.0115: monotonic dissipation ($\alpha_{\max} = 1.874$).



β=0.281105, ϵ=0.013, γ=0.0880: $\alpha_{\max}$=1.7802.

Fig. 12. *Upper left*: AB3–AM4 scheme (2.32) with the maximum possible stability range and third-order accuracy for $\lambda = \lambda(\alpha)$. The physical modes touch the unit circle at $\alpha \approx \pm 2\pi/3$. Larger values of β result in the physical modes going outside the circle near these α values (as in Fig. 10). Smaller β values cause an earlier escape of one of the computational modes along the negative real axis. *Upper right*: a third-order schene with parameters slightly deviating from optimum stability to ensure that numerical dissipation increases monotonically with α. *Lower left*: a multi-purpose compromise with β set to maximize the stability range for the advection equation, while ϵ and γ are set to yield a good stability range for the wave system while maintaining monotonically increasing dissipation.

to an insignificant decrease in $\alpha_{\max}$. Since this is achieved with a smaller value of β, the stability range for advection and Coriolis force is also decreased.

From a practical point of view, it is attractive to chose $\beta = 0.281105$, corresponding to the largest possible stability range for advection and Coriolis force within the β-family for AB3-like schemes, accompanied by $\gamma = 0.088$ and $\epsilon = 0.013$ that yield a sufficiently large stability range for waves (Fig. 12, lower-left) and a dissipation-dominant truncation error. This compromise gives second-order accuracy, and our experience is that it is robust even applied to the full nonlinear system (Sec. 4). Thus, it is the method of choice for the barotropic mode.

To summarize for (2.32), we note that the crucial step to obtain an algorithm with a stability limit comparable to that of FB ($\alpha_{\max} = 2$) is to reduce the curvature parameter of the AB3-like step for ζ by setting $\beta < 5/12$. This brings a tension between the need to keep β relatively large to avoid an asymptotic instability for centered advection and Coriolis force, and the desire to expand the maximum stability range for the wave system that favors bringing β closer to zero. A simultaneous optimization of both stability

ranges yields a useful range of $0.21 < \beta < 0.281105$. Remarkably, in this range the AM4-like coefficients in the second equation in (2.32) end up quite different from that in the classical AM4 weights, and one can verify that terms with ζ^{n+1}, ζ^n, ζ^{n-1}, and ζ^{n-2} in the r.h.s. all have positive coefficients for all of the cases shown in Fig. 12.

2.5 Summary for Time-stepping Algorithms

We have analyzed four different classes of algorithms for a wave system that use a degree of forward-backward feedback to achieve better accuracy for modeling the phase speed for wave motions and/or extend the stability limit $\alpha_{\max}$ over their previously known prototypes. Although in most oceanic applications external and internal waves are not the major interest, the wave properties of the discretized system is always a primary concern from a numerical viewpoint because they are likely to impose the most restrictive limit on the time-step size in flows with small Froude number. Unlike for a simple advection equation, where one can construct a stable algorithm using a simple forward-in-time stepping and upstream-biased, semi-Lagrangian discretization in space, the stability of a wave system cannot rely entirely on a specially designed spatial operator [11] . Thus, stability of an algorithm with respect to wave motions is always in consideration, even thought the final selection of the time-stepping scheme among the ones described above depends on the choice of algorithms for spatial discretization that, in turn, depends on the physical application. The algorithms in Sec. 2.2 are fully compatible with centered advection for the tracer and momentum equations, and they naturally incorporate the treatment of Coriolis force using a synchronous LF–AM3 predictor-corrector step. The same remark applies to the algorithms in Sec. 2.4; however, compatibility with centered advection imposes some restriction on the choice of coefficients (formally $\beta > 1/6$ in (2.32, but in practice we use a greater value of β), which lead to a compromise in the stability limit $\alpha_{\max}$ for wave motions. In contrast, algorithms in Sec. 2.3 are incompatible with centered advection because of weak instability of second-order Adams-Bashforth step (one needs at least small viscosity to mitigate this, or introduce "a forward bias" into AB2 extrapolation coefficients, Campin *et al.* (2004), which is in essense setting $\beta > 1/2$ in (2.26) in it would be a single advection equation). Similarly, the RK2-type algorithms in Sec. 2.1 always require some degree of upstream bias in the advection scheme for stability because RK2 is asymptotically unstable in combination with centered advection. Monotonicity-preserving advection schemes typically require two-level time-stepping and have built-in compatibility with forward-in-time stepping but are incompatible with algorithms that have negative coefficients in their temporal interpolation. This makes RK2 preferable, if monotonicity is desired (*e.g.*, in modeling estuaries characterized by sharp fronts in temperature and salinity). The time-stepping algorithms described here are just linearizations of more general algorithms for the full nonlinear system (Sec. 4) which involve other considerations in their design (*e.g.*, conservation properties), resulting in additional selection criteria.

3 Vertical Mode-Splitting

Although vertical mode-splitting has been used in oceanic modeling since the very beginning (Bryan & Cox, 1969; Berntsen *et al.*, 1981; Blumberg & Mellor, 1987; Bleck & Smith, 1990; Killworth *et al.*,

[11] In principle, one can separate signals propagating in different directions and construct an approximate Riemann solver (Roe, 1981), which essentially relies on upstream-biased algorithms for stability. However, this is not a viable option for oceanic modeling because of complexity (due to implied normal mode decomposition in vertical direction in the case of 3D mode (Shulman *et al.*, 1999)), computational cost, large numerical dissipation, and the implied directional splitting that is not desirable.

1991), a mature theoretical understanding of its stability and accuracy is relatively recent (Skamarock & Klemp, 1992; Higdon & Bennett, 1996; Higdon & de Szoeke, 1997; Hallberg, 1997; Nadiga *et al.*, 1997; Higdon, 1999, 2002; Shchepetkin & McWilliams, 2005). The major issues to be resolved in this approach are (*i*) an inaccurate separation of fast- and slow-time (*i.e.,* barotropic and baroclinic) components in the PGF that may cause "leakage" of fast-time signals into the slow evolution and numerical instability even for linearized systems (Higdon & Bennett, 1996); (*ii*) the time delay in calculating the vertically integrated r.h.s. terms of the slow component can, in effect, be a forward-in-time treatment of the barotropic mode, with associated loss of accuracy and numerical instability; (*iii*) an aliasing of fast barotropic signals due to sub-sampling in the baroclinic time-stepping; (*iv*) a loss of conservation and constancy preservation properties for tracers in both split-explicit (Griffies *et al.*, 2001) and implicit free-surface models (Adcroft & Cadmin, 2004); (*v*) the compressibility effect in EOS complicates the definition of the barotropic PGF with the Boussinesq approximation; and (*vi*) the bottom stress must be known before the barotropic mode starts at every baroclinic time step.

3.1 Tracer Conservation and Constancy-Preservation

In an incompressible fluid the equation for material tracers q can be written in two forms, respectively emphasizing the Lagrangian-parcel and volume-integral conservation properties:

$$\text{advection form:} \qquad \frac{\partial q}{\partial t} + \Big(\mathbf{u}\cdot\nabla\Big)q = 0 \tag{3.1}$$

$$\text{conservation form:} \qquad \frac{\partial q}{\partial t} + \nabla\cdot(\mathbf{u}q) = 0\,. \tag{3.2}$$

The continuity (nondivergence) equation $\nabla\cdot\mathbf{u} = 0$ plays the role of a compatibility condition making these two forms equivalent. If q is initially uniform in space, parcel conservation implies that it remains so: the property of constancy-preservation.

Oceanic models always use the conservation form as the prototype for discrete equations,

$$\begin{aligned}\Delta\mathscr{V}^{n+1}_{i,j,k}\,q^{n+1}_{i,j,k} = \Delta\mathscr{V}^{n}_{i,j,k}\,q^{n}_{i,j,k} - \Delta t\,\Big[&\widetilde{q}_{i+\frac12,j,k}U_{i+\frac12,j,k} - \widetilde{q}_{i-\frac12,j,k}U_{i-\frac12,j,k} + \widetilde{q}_{i,j+\frac12,k}V_{i,j+\frac12,k}\\ &-\widetilde{q}_{i,j-\frac12,k}V_{i,j-\frac12,k} + \widetilde{q}_{i,j,k+\frac12}W_{i,j,k+\frac12} - \widetilde{q}_{i,j,k-\frac12}W_{i,j,k-\frac12}\Big]\,,\end{aligned} \tag{3.3}$$

where discrete concentration values $q_{i,j,k}$ are understood as averages over the local control-volumes $\Delta\mathscr{V}_{i,j,k}$; *i.e.,* $q_{i,j,k} = \dfrac{1}{\Delta\mathscr{V}_{i,j,k}}\displaystyle\int\limits_{\Delta\mathscr{V}^n_{i,j,k}} q(x,y,z)\;\mathrm{d}^3\mathscr{V}$. The tilde operator $\widetilde{q}_{i+\frac12,j,k}$ denotes an appropriate translation algorithm from grid-box averages to interface values, either as a simple spatial interpolation or as one involving both space and time in a semi-Lagrangian approach. $U_{i+\frac12,j,k}$, $V_{i,j+\frac12,k}$, $W_{i,j,k+\frac12}$ are volume fluxes across grid-box interfaces. The discretized continuity equation,

$$\Delta\mathscr{V}^{n+1}_{i,j,k} = \Delta\mathscr{V}^{n}_{i,j,k} - \Delta t\cdot\Big[U_{i+\frac12,j,k} - U_{i-\frac12,j,k} + V_{i,j+\frac12,k} - V_{i,j-\frac12,k} + W_{i,j,k+\frac12} - W_{i,j,k-\frac12}\Big]\,, \tag{3.4}$$

is formally consistent with (3.3) for $q_{i,j,k}\equiv 1$; therefore, that as long as (3.4) holds, this time-stepping scheme has both conservation and constancy-preservation.

The control volumes $\Delta\mathscr{V}_{i,j,k} = H_{i,j,k}\Delta\mathscr{A}_{i,j}$ in (3.3) and (3.4) are time-dependent because grid-box heights $H_{i,j,k}$ depend on $\zeta(x,y,t)$,

$$H_{i,j,k} = z_{i,j,k+\frac{1}{2}} - z_{i,j,k-\frac{1}{2}} \quad \text{where} \quad \begin{cases} z_{i,j,k+\frac{1}{2}} = z^{(0)}_{i,j,k+\frac{1}{2}} + \zeta_{i,j} \left(1 + \dfrac{z^{(0)}_{i,j,k+\frac{1}{2}}}{h_{i,j}} \right) \\ z^{(0)}_{i,j,k+\frac{1}{2}} \equiv z^{(0)} \left(\xi_{i,j}, \eta_{i,j}, s_{k+\frac{1}{2}} \right) , \qquad k = 0, 1, \, ... \, N. \end{cases} \tag{3.5}$$

The $z^{(0)}$ comprise a set of unperturbed (*i.e.,* corresponding to $\zeta \equiv 0$) isosurfaces of a terrain-following vertical coordinate, $s_{k+\frac{1}{2}} \in [-1, 0]$. The lowest surface, $z_{i,j,\frac{1}{2}} \equiv z^{(0)}_{i,j,\frac{1}{2}} \equiv -h_{i,j}$, follows the bottom topography. Since $z^{(0)}_{i,j,N+\frac{1}{2}} \equiv 0$, the highest surface $z_{i,j,N+\frac{1}{2}} \equiv \zeta_{i,j}$ follows the oceanic top. Otherwise the vertical coordinate transformation is a general one. In (3.5) the grid-box heights are proportionally stretched relative to their unperturbed values, $H^{(0)}_{i,j,k}$; *i.e.,*

$$H_{i,j,k} = H^{(0)}_{i,j,k} \cdot \left(1 + \frac{\zeta_{i,j}}{h_{i,j}} \right) . \tag{3.6}$$

The loss of constancy-preservation in (3.3) can occur if $\Delta\mathscr{V}^{n+1}_{i,j,k}$ does not come from (3.4), but rather is computed with a barotropic mode that uses a different time step and time-stepping algorithm and, furthermore, is averaged in fast-time, replacing $\zeta \rightarrow \langle\zeta\rangle^{n+1} = \sum_{m=1}^{M^*} a_m \zeta^m$, to prevent aliasing of the barotropic frequencies unresolved by the baroclinic time-stepping. A vertical summation of (3.4) yields

$$\zeta^{n+1}_{i,j} = \zeta^{n}_{i,j} - \frac{\Delta t}{\Delta\mathscr{A}_{i,j}} \cdot \sum_{k=1}^{N} \left[U_{i+\frac{1}{2},j,k} - U_{i-\frac{1}{2},j,k} + V_{i,j+\frac{1}{2},k} - V_{i,j-\frac{1}{2},k} \right] . \tag{3.7}$$

This is not necessarily consistent with the fast-time-averaged free surface computed by the barotropic mode, implying that

$$\langle\zeta\rangle^{n+1} \neq \langle\zeta\rangle^{n} - \Delta t \cdot \mathrm{div} \langle\overline{\mathbf{U}}\rangle \, , \tag{3.8}$$

where indices n and $n+1$ correspond to the slow (baroclinic) time step, and the overbar in $\overline{\mathbf{U}}$ means a vertically integrated volume flux.

Conversely, (3.4) is used for the computation of vertical velocity: start with $W_{i,j,\frac{1}{2}} = 0$ at the bottom and recursively proceed with

$$W_{i,j,k+\frac{1}{2}} = - \sum_{k'=1}^{k} \left\{ \frac{\Delta\mathscr{V}^{n+1}_{i,j,k'} - \Delta\mathscr{V}^{n}_{i,j,k'}}{\Delta t} + U_{i+\frac{1}{2},j,k'} - U_{i-\frac{1}{2},j,k'} + V_{i,j+\frac{1}{2},k'} - V_{i,j-\frac{1}{2},k'} \right\} \tag{3.9}$$
$$\text{for all} \qquad k = 1, 2, ..., N \, .$$

This essentially defines $W_{i,j,k+\frac{1}{2}}$ as the finite-volume, finite-time-interval volume flux across the moving interface between vertically adjacent grid boxes, $\Delta\mathscr{V}_{i,j,k}$ and $\Delta\mathscr{V}_{i,j,k+1}$. This procedure does not automatically guarantee that the surface kinematic boundary condition,

$$W_{i,j,N+\frac{1}{2}} = 0 \, , \tag{3.10}$$

is satisfied if $\Delta\mathscr{V}^{n+1}_{i,j,k}$ comes from the barotropic mode with a different time-stepping.

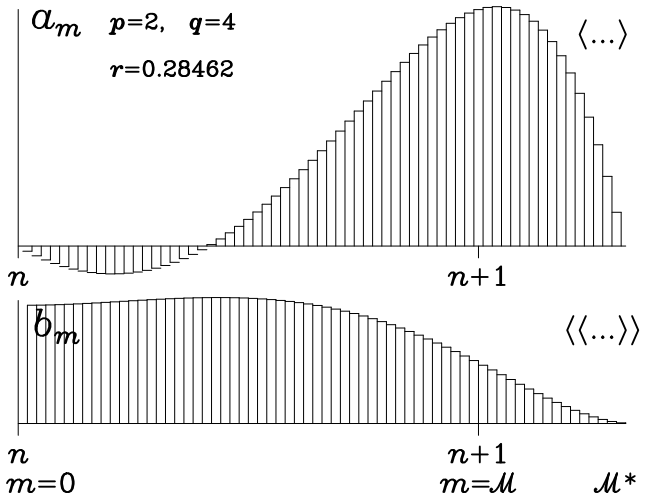


Fig. 13. Relationship between the primary, $\{a_m\}$, and secondary, $\{b_m\}$, fast-time-averaging weights. By definition, $\langle\zeta\rangle^{n+1} \equiv \sum_{m=1}^{M^*} a_m\zeta^m$ and $\langle\!\langle\overline{\mathbf{U}}\rangle\!\rangle^{n+\frac{1}{2}} \equiv \sum_{m=1}^{M^*} b_m\overline{\mathbf{U}}^m$. In order to satisfy normalization and centroid conditions (3.16), the integration of the barotropic mode must go beyond the $n+1$th baroclinic step, hence $M^* > M$. In this example the a_m are negative at the beginning of their sequence (*i.e.*, they have a S-shape). The value of this negative lobe and the meaning of parameters p, q, r are explained in Sec. 3.3.

To ensure that slow-time continuity equation (3.4) is consistent with the barotropic mode we must impose a constraint on the vertical integrals of the volume fluxes, $U_{i+\frac{1}{2},j,k}$ and $V_{i,j+\frac{1}{2},k}$,

$$\sum_{k=1}^{N} U_{i+\frac{1}{2},j,k} = \left\langle\!\left\langle \overline{U} \right\rangle\!\right\rangle^{n+\frac{1}{2}}_{i+\frac{1}{2},j} \qquad \text{and} \qquad \sum_{k=1}^{N} V_{i,j+\frac{1}{2},k} = \left\langle\!\left\langle \overline{V} \right\rangle\!\right\rangle^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} , \tag{3.11}$$

so that

$$\langle\zeta\rangle^{n+1}_{i,j} = \langle\zeta\rangle^{n}_{i,j} - \frac{\Delta t}{\Delta\mathscr{A}_{i,j}} \left[\left\langle\!\left\langle \overline{U} \right\rangle\!\right\rangle^{n+\frac{1}{2}}_{i+\frac{1}{2},j} - \left\langle\!\left\langle \overline{U} \right\rangle\!\right\rangle^{n+\frac{1}{2}}_{i-\frac{1}{2},j} + \left\langle\!\left\langle \overline{V} \right\rangle\!\right\rangle^{n+\frac{1}{2}}_{i,j+\frac{1}{2}} - \left\langle\!\left\langle \overline{V} \right\rangle\!\right\rangle^{n+\frac{1}{2}}_{i,j-\frac{1}{2}} \right] \tag{3.12}$$

is consistent with the change in $\langle\zeta\rangle$ between two consecutive baroclinic time steps. To define the second averaging operator $\langle\!\langle ... \rangle\!\rangle$, we note that a summation of consecutive barotropic time steps yields

$$\zeta^{m+1} = \zeta^m - \frac{\Delta t}{M} \cdot \mathrm{div}\overline{\mathbf{U}}^{m+\frac{1}{2}} , \qquad \text{hence} \qquad \zeta^m = \zeta^0 - \frac{\Delta t}{M} \sum_{m'=0}^{m-1} \mathrm{div}\overline{\mathbf{U}}^{m'+\frac{1}{2}} , \tag{3.13}$$

where m is the fast-time index and M is the integer mode-splitting ratio (*i.e.*, ratio of commensurate baroclinic and barotropic time-step sizes). The $m = 0$ starting field ζ^0 corresponds to the baroclinic step n, and the barotropic mode restarts at the end of every baroclinic time step, $\langle\zeta, \overline{U}, \overline{V}\rangle^{n+1} \rightarrow \zeta^0, \overline{U}^0, \overline{V}^0$. After applying fast-time averaging $\langle ... \rangle$ to both sides of (3.13),

$$\langle\zeta\rangle^{n+1} \equiv \sum_{m=1}^{M^*} a_m\zeta^m = \zeta^0 - \frac{\Delta t}{M} \cdot \mathrm{div} \sum_{m=1}^{M^*} \left[a_m \sum_{m'=1}^{m} \overline{\mathbf{U}}^{m'-\frac{1}{2}} \right] . \tag{3.14}$$

This translates into

$$\langle\zeta\rangle^{n+1} = \langle\zeta\rangle^{n} - \Delta t \cdot \mathrm{div} \sum_{m'=1}^{M^*} b_{m'}\overline{\mathbf{U}}^{m'-\frac{1}{2}} \qquad \text{where} \quad b_{m'} = \frac{1}{M} \sum_{m=m'}^{M^*} a_m \tag{3.15}$$

$\forall\, m' = 1, .., M^*$. The coefficients $\{a_m, m = 1, ..., M^*\}$ are the primary averaging weights (Fig. 13) that satisfy normalization and centroid conditions,

$$\sum_{m=1}^{M^*} a_m \equiv 1 \qquad \text{and} \qquad \sum_{m=1}^{M^*} \frac{m}{M}\, a_m \equiv 1 . \tag{3.16}$$

but they are otherwise arbitrary thus far. $M^* \geq M$ is the fast-time index of the last non-zero a_m. We define

$$\left\langle\!\left\langle \overline{\mathbf{U}} \right\rangle\!\right\rangle^{n+\frac{1}{2}} \equiv \sum_{m=1}^{M^*} b_m \overline{\mathbf{U}}^{m-\frac{1}{2}} \,. \tag{3.17}$$

Using this in the integral constraint (3.11) with (3.15) guarantees that (3.4) holds exactly between baroclinic steps n and $n+1$ and, therefore, guarantees both conservation and constancy-preservation properties in (3.3). In practice, after completion of the barotropic time-stepping at every baroclinic time step, five fields ($\langle\zeta\rangle^{n+1}$, $\langle\overline{U}\rangle^{n+1}$, $\langle\overline{V}\rangle^{n+1}$, $\left\langle\!\left\langle \overline{U} \right\rangle\!\right\rangle^{n+\frac{1}{2}}$, and $\left\langle\!\left\langle \overline{V} \right\rangle\!\right\rangle^{n+\frac{1}{2}}$) must be available for the baroclinic integration since $\langle\!\langle ... \rangle\!\rangle$ fields cannot be expressed directly in terms of $\langle ... \rangle$ fields.

3.2 Mode-Splitting Error in the Pressure-Gradient Force

Vertical mode-splitting separates the vertically integrated, hydrostatic, horizontal PGF,

$$\mathscr{F} \equiv \mathscr{F}\left[\nabla_x\zeta,\ \zeta,\ \ \nabla_x\rho(z),\ \rho(z)\right] = -\frac{1}{\rho_0}\int_{-h}^{\zeta} \nabla_x P\, dz = -\frac{g}{\rho_0}\int_{-h}^{\zeta}\left[\int_{z}^{\zeta} \nabla_x\rho\left(z'\right)\, dz'\right] dz\,, \tag{3.18}$$

into a "fast" term , $-gD\nabla_x\zeta$, and the remaining "slow" $\left\{....\right\}$ terms (these are also known as "coupling" or baroclinic-to-barotropic forcing terms),

$$\frac{\partial\left(D\overline{u}\right)}{\partial t} + ... = -gD\nabla_x\zeta + \left\{gD\nabla_x\zeta + \mathscr{F}\right\}. \tag{3.19}$$

The fast terms are recomputed at every barotropic step, while the slow terms are held constant since they change only once per baroclinic time step. $D = h + \zeta$ is total depth of a vertical column. If the functional $\mathscr{F}$ contains nonlinear combinations of ζ and ρ (*i.e.,* $\partial^2\mathscr{F}/\partial\zeta\,\partial\rho \neq 0$), freezing the slow terms can cause a mode-splitting error,

$$-gD\nabla_x\zeta' + \left\{gD\nabla_x\zeta + \mathscr{F}\left[\nabla_x\zeta,\ \zeta,\ \ \nabla_x\rho(z),\ \rho(z)\right]\right\} \neq \mathscr{F}\left[\nabla_x\zeta',\ \zeta',\ \nabla_x\rho(z),\ \rho(z)\right]\,; \tag{3.20}$$

i.e., at the end of barotropic time-stepping when $\zeta \rightarrow \zeta'$, the PGF seen by the barotropic mode no longer matches the vertical integral of the total PGF from the same ρ and the new ζ. Consequently, at the beginning of the new time step when the full PGF is recomputed, its vertical integral is no longer in equilibrium with the state of the barotropic mode PGF even in the case when there is no change of the σ-level ρ values between consecutive baroclinic steps. The mismatch between the two contaminates the forcing terms computed and the new time step, and subsequently affects the state of barotropic mode one step later, thereby closing the feedback loop.

In isopycnic coordinates Higdon & Bennett (1996); Higdon & de Szoeke (1997); Hallberg (1997) found an instability of the linearized mode-split system with non-dissipative time-stepping schemes (FB, LF). Their diagnosis and proposed remedies were that (*i*) mode-splitting can cause artificial mode-coupling; (*ii*) for some time-stepping schemes the mode-coupling may cause a phase lag that induces a numerical instability similar to that of a forward time step for a hyperbolic system; (*iii*) a perturbation analysis of weakly coupled linear system shows that the instability is a resonance of an aliased barotropic mode subsampled at the baroclinic steps; (*iv*) the remedy is to redefine barotropic mode PGF to make it be equal to

the vertical integral of the 3D PGF; and (*v*) a dissipative time-stepping scheme that filters the barotropic mode to prevent aliasing or a dissipative predictor–corrector scheme (Hallberg, 1997) can be a useful way to achieve stability.

The common justification for (3.19) is $\zeta \ll D$ and $\rho' \equiv \rho - \rho_0 \ll \rho_0$, hence the magnitude of the mismatch in (3.20) is $\mathcal{O}\left(\max\left\{(\rho'/\rho_0)\nabla_x\zeta\ ,\ \zeta\nabla_x\rho'/\rho_0\right\}\right)$ relative to $\mathcal{O}\left(\nabla_x\zeta\right)$. Among other restrictions this implies that ρ_0 must be chosen sufficiently close to the actual density ρ to avoid a "leakage" of barotropic signals into the baroclinic mode (Higdon, 2002). Suppose that both modes are time-stepped within but close to their CFL limits of stability taken individually. This implies a choice of M in (3.13) as the ratio of the barotropic and first-baroclinic gravity-wave phase speeds adjusted by the ratio of the stability limits of their respective time-stepping algorithms. The coupled system may still be unstable if $M > \mathcal{O}\left(\rho_0/\left|\rho-\rho_0\right|\right)$. This is because in a Boussinesq model using splitting (3.19) the barotropic pressure gradient term arising from free-surface gradient $\nabla_x\zeta$ creates an acceleration equal to $-g\nabla_x\zeta$ independently of the choice of ρ_0. On the other hand the net vertically integrated PGF computed by full (baroclinic + barotropic) 3D scheme from a given density field and given state of free-surface has slightly different sensitivity to $\nabla_x\zeta$, in creates acceleration more similar to $-g\nabla_x\left[\left(1+\rho^{*\prime}/\rho_0\right)\zeta\right]$ where $\rho^{*\prime}$ depends on the deviation of local density from ρ_0 in a manner quantified later in this section. This leads to the fact that phase speed of surface gravity waves as seen by the 3D part of the code is different from that seen by the barotropic mode. To avoid numerical instability, the difference in phase increment per one baroclinic time step Δt between the two must be smaller that allowed by CFL criterion for the time stepping scheme for the baroclinic mode. Since the density variation due to baroclinic effects can be estimated as large as 3% (*i.e.,* comparable, and in some situations larger that the ratio phase speeds of barotropic and the first baroclinic modes) this potentially may force to chose a smaller Δt than required for stability of the baroclinic mode taken alone.

Furthermore, even if the mismatch in (3.20) is small in most cases, the primary concern here is that it still may cause a numerical instability even if ρ variations are small and ρ_0 is chosen so that the preceding M-criterion is respected. This is due to phase delays in computing the mismatch term associated with the organization of the coupled time-stepping algorithm. Another remedy to mitigate the consequences of this type of error is the use of a dissipative time filter for the barotropic mode (Sec. 3.3): however, this unavoidably degrades the numerical accuracy. Either way, it is always desirable to remove or minimize the mismatch.

Eq. (3.20) suggests a general guideline for eliminating the mode-splitting PGF error by replacing $-gD\nabla_x\zeta$ in (3.19) with the variational derivative of $\mathscr{F} = \mathscr{F}\left[\nabla_x\zeta\ ,\ \zeta\ ,\ ...\right]$,

$$\delta\mathscr{F} = \frac{\partial\mathscr{F}}{\partial\left(\nabla_x\zeta\right)}\delta\left(\nabla_x\zeta\right) + \frac{\partial\mathscr{F}}{\partial\zeta}\delta\zeta\ . \tag{3.21}$$

ζ and $\nabla_x\zeta$ are treated as independent variables for the functional partial differentiation. In the discretized version this corresponds to having ζ_i and ζ_{i+1} as independent degrees of freedom that are alternatively expressible as their difference $\zeta_{i+1}-\zeta_i$ and average $(\zeta_{i+1}+\zeta_i)/2$. Substitution of (3.21) into (3.20) makes it into a Taylor-series expansion,

$$\mathscr{F}\left[\nabla_x\zeta,\ \zeta,\ ...\right] + \frac{\partial\mathscr{F}}{\partial\left(\nabla_x\zeta\right)}\nabla_x\left(\zeta'-\zeta\right) + \frac{\partial\mathscr{F}}{\partial\zeta}\left(\zeta'-\zeta\right) \approx \mathscr{F}\left[\nabla_x\zeta',\ \zeta',\ ...\right]\ , \tag{3.22}$$

resulting in a cancellation of the dominant part of the mode-splitting error: recall that the mismatch between l.h.s. and r.h.s. of (3.22) can be estimated as $\mathcal{O}\left(\left(\nabla_x\left(\zeta'-\zeta\right)\right)^2\right) + \mathcal{O}\left(\left(\zeta'-\zeta\right)^2\right)$.

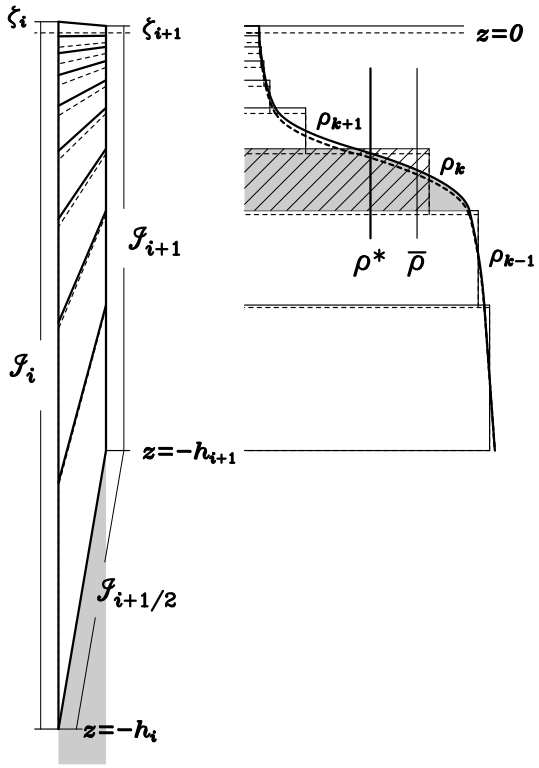


Fig. 14. *Left*: A segment of the vertical grid used in derivation of total vertically-integrated PGF (3.23). Dashed lines correspond to the unperturbed ($\zeta = 0$) vertical coordinate and solid lines to the coordinate perturbed according to (3.6). *Right*: Computation of 2-way vertically averaged densities (3.25) for a stratified water column. The ρ_k are interpreted as control-volume averages, hence the area of hatched rectangle is equal to the shaded area left from the continuous profile. Note that for a stably-stratified profile ρ^* is systematically smaller that $\overline{\rho}$, as illustrated here.

Note that the net horizontal force applied to fluid element in Fig. 14 can be calculated as

$$\begin{aligned}\mathscr{F}_{i+\frac{1}{2}} = \int_{-h_i}^{\zeta_i} P(x_i, z)\, dz - \int_{-h_{i+1}}^{\zeta_{i+1}} P(x_{i+1}, z)\, dz + \int_{x_i}^{x_{i+1}} P\Big(x, -h(x)\Big) \frac{\partial h(x)}{\partial x} dx \\ = \mathcal{I}_i - \mathcal{I}_{i+1} + \mathcal{I}_{i+\frac{1}{2}}\,,\end{aligned} \tag{3.23}$$

where $P_i(z)$ is hydrostatic the pressure calculated separately in each vertical column,

$$P_i(z) = g \int_{z'_i}^{\zeta_i} \rho_i(z')\, dz'\,. \tag{3.24}$$

By introducing

$$\overline{\rho}(x) = \frac{1}{D_i} \int_{-h_i}^{\zeta_i} \rho_i(z') dz' \qquad \text{and} \qquad \rho_i^* = \frac{1}{\frac{1}{2} D_i^2} \int_{-h_i}^{\zeta_i} \left\{ \int_{z_i}^{\zeta_i} \rho_i(z') dz' \right\} dz \tag{3.25}$$

where $D_i = \zeta_i + h_i$, the net force (3.23) be expressed as

$$\mathscr{F}_{i+\frac{1}{2}} = g \left[\frac{\rho_i^* D_i^2}{2} - \frac{\rho_{i+1}^* D_{i+1}^2}{2} + \int_{x_i}^{x_{i+1}} \overline{\rho} D \, \frac{\partial h}{\partial x} dx \right]\,. \tag{3.26}$$

This corresponds to the continuous form,

$$\frac{\partial}{\partial t}\left(D\overline{u}\right)+...=-\frac{g}{\rho_0}\left[\nabla_x\left(\frac{\rho^* D^2}{2}\right)-\overline{\rho}D\nabla_x h\right]=-\frac{g}{\rho_0}\Bigg[\underbrace{\frac{h^2}{2}\nabla_x\rho^*+\left(\rho^*-\overline{\rho}\right)\nabla_x\frac{h^2}{2}}_{\mathscr{F}^{(0)}\ (\zeta=0\text{ part})} \tag{3.27}$$

$$+\underbrace{h\nabla_x\left(\rho^*\zeta\right)+\nabla_x\left(\frac{\rho^*\zeta^2}{2}\right)+\left(\rho^*-\overline{\rho}\right)\zeta\nabla_x h}_{\mathscr{F}'\ (\text{perturbation due to }\zeta\neq 0)}\Bigg],$$

where we have separated $\mathscr{F}^{(0)}$ which is independent of ζ and the remainder, $\mathscr{F}'$. Since the mode-coupling algorithm already performs a vertical integration of the momentum r.h.s. terms, including the full PGF, $\mathscr{F}^{(0)}$ is not further required. However, $\mathscr{F}'$ satisfies (3.21) and is therefore a valid replacement for $-gD\nabla_x\zeta$ in (3.19) (as expected, one can easily verify that $\mathscr{F}'$ reverts back to $-gD\nabla_x\zeta$ if ρ is uniform, $\rho^*=\overline{\rho}=\rho_0$).

The accuracy of mode splitting using the decomposition of $\mathscr{F}=\mathscr{F}^{(0)}+\mathscr{F}'$ fundamentally comes from the fact that changes in ζ from one time step to the next do not the modify grid-box values of density $\rho_{i,j,k}$. In a purely barotropic motion fluid parcels move up and down following changing free surface, and the grid-box locations move together with the parcels (3.5), resulting in no change in $\rho_{i,k}$. Hence, ρ^* and $\overline{\rho}$ in (3.25) are also nearly independent of ζ, which justifies keeping them constant during the fast-time stepping of barotropic mode. To ensure numerical stability and at least second-order accuracy, $\rho_{i,k}$, ρ^*, and $\overline{\rho}$ must be time-centered at $n+1/2$ in baroclinic time.

An analogous discrete derivation (Shchepetkin & McWilliams, 2005) yields

$$\begin{aligned}\mathscr{F}'_{i+\frac{1}{2}}=-\frac{g}{\rho_0}\Bigg\{&\frac{h_{i+1}+h_i}{2}\left(\rho^*_{i+1}\zeta_{i+1}-\rho^*_i\zeta_i\right)+\frac{\rho^*_{i+1}\zeta^2_{i+1}}{2}-\frac{\rho^*_i\zeta^2_i}{2}\\&+\left(h_{i+1}-h_i\right)\left[\frac{\left(\rho^*_{i+1}-\overline{\rho}_{i+1}\right)\zeta_{i+1}+\left(\rho^*_i-\overline{\rho}_i\right)\zeta_i}{2}+\frac{\left(\overline{\rho}_{i+1}-\overline{\rho}_i\right)\left(\zeta_{i+1}-\zeta_i\right)}{6}\right]\Bigg\}.\end{aligned} \tag{3.28}$$

The particular form of (3.28) depends on the discrete scheme for 3D PGF (Sec. 5). In principle, the splitting error can be eliminated entirely rather than just the leading-order term cancellation in (3.22). However, doing so imposes severe restrictions on the discretization choice for the 3D PGF that basically would then be limited to pressure-Jacobian schemes (Shchepetkin & McWilliams, 2003). This is undesirable because it raises the overall error in the PGF. For example, the scheme in Lin (1997) results in (3.28) without the last term inside $[...]$ on the second line. Although this term is formally $\mathcal{O}\left(\Delta x^3\right)$-small (*i.e.,* two orders higher than the preceding term), it is desirable to keep it since it makes (3.28) exact if ρ is a linear function of depth and horizontal coordinate, unlike the scheme in Lin (1997). A density-Jacobian scheme (as in Blumberg & Mellor (1987)) does not allow separating ρ values belonging to different horizontal indices, so that the vertical integral of $\mathscr{F}$ cannot be expressed in terms of ρ^* and $\overline{\rho}$ computed independently within each vertical column. The standard PGF scheme in ROMS (Shchepetkin & McWilliams, 2003) uses a 4-point stencil in the horizontal and nonlinear interpolation of density to avoid spurious oscillations; both attributes make it impractical to derive an exactly consistent PGF scheme for the barotropic mode. Nevertheless, practical experience with (3.28) indicates that it is sufficiently accurate and stable.

For flat topography ρ^* is the only relevant density for the barotropic mode. This choice is similar to (3.2) in Higdon (1999), but is differs from Bleck & Smith (1990) which uses the vertically averaged density (analogous here to $\overline{\rho}$) and from Griffies *et al.* (2001) that uses the local density at the top-most grid cell instead of ρ^*. All other split-explicit models just use ρ_0. The terms proportional to $\nabla_x h$ in (3.27) and (3.28) reflect the dynamical coupling between barotropic and baroclinic motions; it depends on the density difference, $\rho^*-\overline{\rho}$, and thus it is part of what is sometimes referred to as the JEBAR effect, (Holland, 1973).

3.3 Design of the Fast-time Averaging Filter

Averaging of the barotropic mode in a split-explicit model (***i.e.,*** choosing a_m in (3.14) distinct from a delta-function $\delta_{mM} = \{1,\, m = M; \quad 0,\, m \neq M\}$ is sometimes viewed as a "necessary evil" (Hallberg, 1997; Higdon, 1999; Griffies ***et al.***, 2001): while it yields a stable and robust numerical code, it undesirably degrades the temporal accuracy of the resolved barotropic motions and often introduces a numerical dissipation comparable to that of implicit backward-Euler time-stepping. We identify three reasons for averaging. First, although the effort is made to remove mode splitting error in PGF (Sec. 3.2), the split is never perfect in practice. If both the barotropic and baroclinic time-stepping algorithms are non-dissipative, barotropic aliasing may introduce numerical instability, Higdon & Bennett (1996), whereas fast-time averaging excludes the possibility of a coincidence of characteristic roots λ by placing the barotropic roots from the aliased range deep inside the unit circle, (Sec. 3.2). Second, depending on the stage when the time-stepping algorithm computes the vertically-integrated momentum advection terms that are kept constant during the barotropic time-stepping, they may incur a delay effectively like a forward time step for these terms. This leads to numerical instability of the same type as for a forward-in-time, centered-in-space advection equation. Fast-time-averaging provides a mechanism to control this instability. This aspect puts an emphasis on damping at the low-frequency end, which is a very different requirement for the filter design compared to its anti-aliasing role. Third, depending on the algorithm for taking the first time step (typically forward-in-time), the recurrent restart of the barotropic mode at each baroclinic time step may introduce yet another numerical instability. Net dissipation in the barotropic time-stepping scheme and fast-time averaging can suppress this instability.

We now examine the design principles for the barotropic time filters. For simplicity of analysis, we assume $M \gg 1$, neglect the truncation error in the barotropic time-stepping, and replace the discrete summation over fast-time indices with a continuous time integration. $A(\tau)$ is defined as the continuous analog of $\{a_m |\, m = 1, ..., M^*\}$ with $\tau \sim m/M$ and $\tau_* \sim M^*/M$. A barotropic Fourier component ω_k gets a phase increment $\alpha = \omega_k \Delta t$ in one baroclinic time step Δt. After fast-time averaging, its step multiplier becomes

$$\lambda(\alpha) = \int_0^{\tau_*} \mathrm{e}^{-i\alpha\cdot\tau} A(\tau)\, d\tau = \mathfrak{R}(\alpha)\mathrm{e}^{-i\alpha}\,, \tag{3.29}$$

where $\mathfrak{R}(\alpha)$ is the response function. Ideally $\mathfrak{R}(\alpha) \approx 1$ for $\alpha \leq \alpha_0 \sim 1$ and $\mathfrak{R}(\alpha) \to 0$ rapidly in α once $\alpha > \alpha_0$. In the vicinity of $\alpha = 0$, $1 - \mathfrak{R}(\alpha) = \mathcal{O}\left(\alpha^n\right)$, where n is the temporal order of accuracy. Substitution of a Taylor-series expansion $\mathrm{e}^{-i\alpha\tau} = 1 - i\alpha\tau - \dfrac{\alpha^2\tau^2}{2} + \dfrac{i\alpha^3\tau^3}{6} + ...$ for $|\alpha| \ll 1$ in (3.29) leads to

$$\lambda(\alpha) = 1 - i\alpha - \frac{\alpha^2}{2}\mathfrak{I}_2 + \frac{i\alpha^3}{6}\mathfrak{I}_3 + \frac{\alpha^4}{24}\mathfrak{I}_4 + ... \quad \text{where} \quad \mathfrak{I}_n = \int_0^{\tau_*} \tau^n A(\tau)\, d\tau\,, \tag{3.30}$$

with $\mathfrak{I}_0 \equiv \mathfrak{I}_1 \equiv 1$ due to the normalization and consistency conditions analogous to (3.16). Using the identity, $\tau^2 \equiv (\tau - 1)^2 + 2\tau - 1$, and the relation, $2\mathfrak{I}_1 - \mathfrak{I}_0 \equiv 1$, we find that

$$\mathfrak{I}_2 \equiv \int_0^{\tau_*} \tau^2 A(\tau)\, d\tau = 1 + \int_0^{\tau_*} (\tau - 1)^2 A(\tau)\, d\tau \equiv 1 + \epsilon\,. \tag{3.31}$$

If $A(\tau)$ is non-negative, the integrands are non-negative too; hence, $\epsilon \geq 0$, with equality reached only if $A(\tau)$ is a delta-function, $\delta(\tau-1)$. Substitution of $\mathfrak{I}_2$ into (3.30) leads to the appearance of ϵ as a coefficient in the leading-order truncation term at second order. $\epsilon > 0$ corresponds to numerical dissipation. Therefore any choice of a positive-definite $A(\tau)$, results in at most first-order accuracy for the fast-time-averaged barotropic mode (*i.e.,* $\lambda(\alpha)$ agrees with $\mathrm{e}^{-i\alpha}$ only up to $\mathcal{O}(\alpha^2)$).

To achieve second-order accuracy, we introduce a shape function that allows some of the primary weights to be negative,

$$A(\tau) = A_0 \left\{ \left(\frac{\tau}{\tau_0}\right)^p \left[1 - \left(\frac{\tau}{\tau_0}\right)^q\right] - r\frac{\tau}{\tau_0} \right\} \tag{3.32}$$

where p and q are independent parameters. A_0, τ_0, and r are then chosen to satisfy normalization, centroid, and second-order accuracy conditions in (3.30), *viz.,* $\mathfrak{I}_n = 1$ for $n = 0, 1, 2$. In practice we initially specify

$$r = 0 \quad \text{and} \quad \tau_0 = \frac{(p+2)\,(p+q+2)}{(p+1)\,(p+q+1)}, \tag{3.33}$$

(this choice of τ_0 centers $A(\tau)$ at $\tau = 1$; *i.e.,* $\mathfrak{I}_1/\mathfrak{I}_0 = 1$), then compute A_0 from the normalization condition. Using this initial $A(\tau)$ we adjust r, A_0, and τ_0 with an iterative procedure — adjust r to minimize $\epsilon = \mathfrak{I}_2 - 1$; recompute A_0 and τ_0 to restore $\mathfrak{I}_0 = \mathfrak{I}_1 = 1$; and repeat until $\epsilon \to 0$ — to satisfy the $\mathfrak{I}_n$ conditions. This yields a family $r = r(p, q)$ of second-order filters such as the following tabulated p, q, r-triplets.

$p=2$	$q=1$	$r=0.1696907$
2	2	0.2346283
2	3	0.2664452

$p=2$	$q=4$	$r=0.2846158$
2	6	0.2961888
3	8	0.1369941

The alternative choices, $p, q = 2, 4$ or $2, 2$, are the settings in ROMS for most applications; Fig. 13 is one of the corresponding shape functions.

Fig. 15 compares the step multipliers for some fast-time-averaging algorithms with an S-shaped filter designed as described in this section. Ideally $\lambda(\alpha) \approx 1$ for $\alpha \leq 2$ (the baroclinic time-stepping stability range), and $\lambda(\alpha) \ll 1$ thereafter. As expected, a flat averaging over $2\Delta t$ (left panel) results in very strong damping of the resolved frequencies (Griffies *et al.,* 2001). A Hamming window (Oppenheim & Schafer, 1989) (middle panel) has much smaller dissipation for resolved frequencies and provides an efficient damping for the aliasing range. The $p, q = 2, 4$ filter (right panel) has virtually no damping for $|\alpha| \leq \pi/4$, and it as efficient as the Hamming window in its anti-aliasing role. Another effect of having a negative lobe is that $A(\tau)$ makes the model more efficient by reducing the duration of the barotropic integration beyond t_{n+1} (*i.e.,* , $M^* - M$): the $p, q = 2, 4$ filter takes only 30% of the extra Δt step, while the Hamming window needs 50% and flat averaging needs 100%.

3.4 Comparison with an Implicit Free-Surface Model

An implicit free-surface models entirely eliminates aliasing by simply restricting the phase increment of the barotropic mode. A particular scheme from the CFD community, the theta-method (Casulli & Cattani, 1994), is

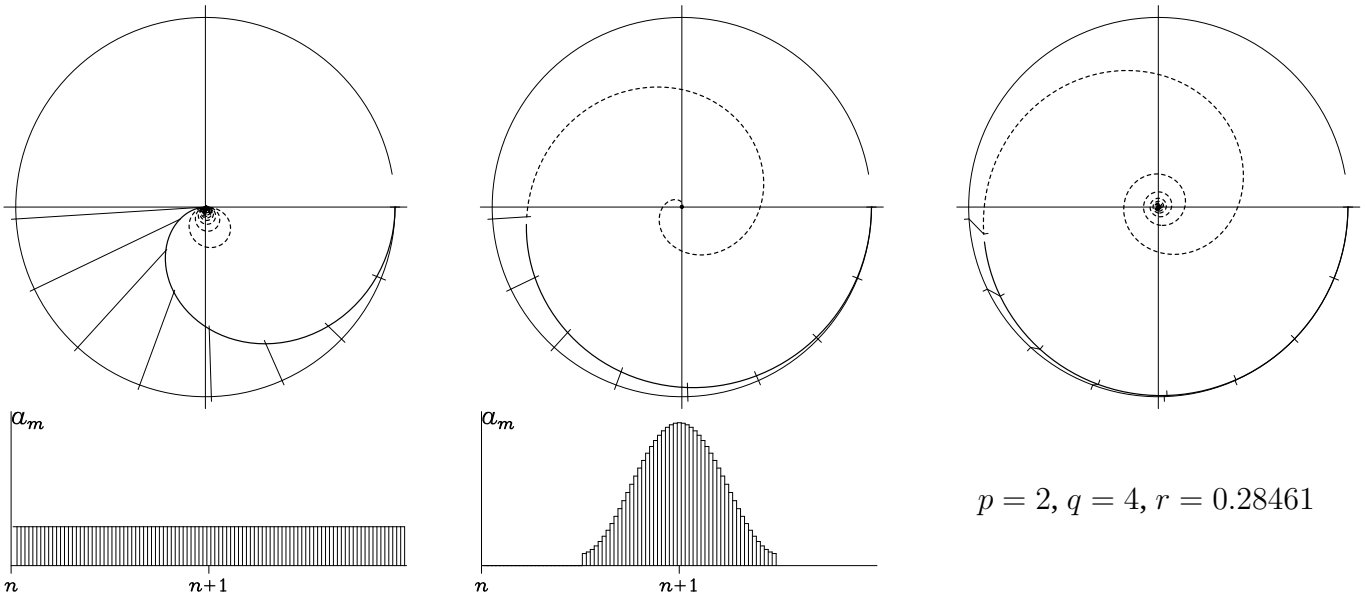


Fig. 15. Step multiplier $\lambda(\alpha)$ for three different choices of the of fast-time-averaging weights. *Left*: flat averaging over $2\Delta t$; *middle*: Hamming window; *right*: S-shaped weights from Fig. 13. The bold solid line on the diagrams turns dashed when entering the aliasing range.

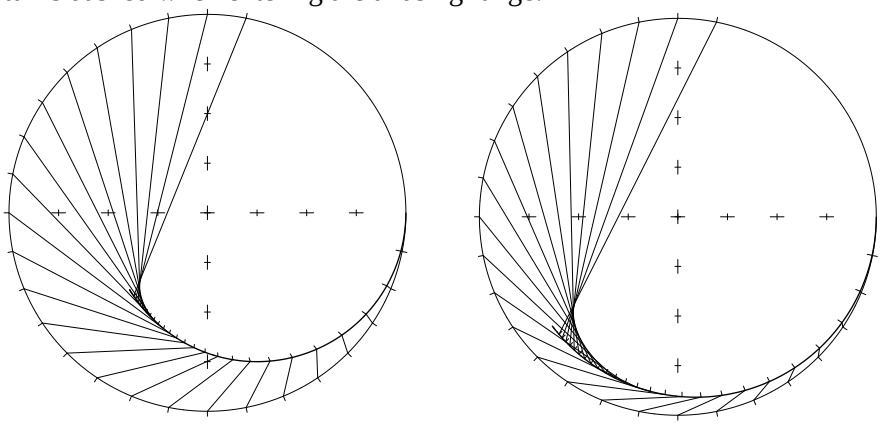


Fig. 16. $\lambda(\alpha)$ for the theta-method with two different θ values in the same format as Fig. 15. Comparing left and right panels shows that, while the dissipation increases with $\theta - 1/2|$, the phase error changes little with θ. The phase of $\lambda(\alpha)$ asymptotes to $-\pi$ when $\alpha \to \infty$; so $\lambda(\alpha)$ never enters the aliasing range.

$$\left.\begin{aligned} \zeta^{n+1} + i\alpha\theta u^{n+1} &= \zeta^n - i\alpha(1-\theta)u^n \\ u^{n+1} + i\alpha\theta\zeta^{n+1} &= u^n - i\alpha(1-\theta)\zeta^n \end{aligned}\right\} \;\Rightarrow\quad \lambda(\alpha) = \frac{1 - \alpha^2\theta(1-\theta) \pm i\alpha}{1 + \alpha^2\theta^2}\,. \tag{3.34}$$

It is unconditionally stable if $1/2 \le \theta \le 1$ and is second-order accurate for $\theta = 1/2$. However, if used with $\alpha > 1$, the $\theta = 1/2$-scheme is prone to $2\Delta t$ oscillations, usually addressed by slightly biasing θ above 1/2, which makes it first-order accurate and dissipative. Setting $\theta = 2/3$ (Fig. 16, left) has a dissipation comparable to flat averaging over $2\Delta t$ (Fig. 15, left). A standard CFD practice is to use $\theta = 0.55$ (Fig. 16, right). Its damping is comparable (about twice as much) to the Hamming window. Since no third- or higher-order, unconditionally stable, implicit algorithm exists (*n.b.*, an implicit AM3 scheme is asymptotically unstable for a purely hyperbolic problem), the theta-method is the only possibility for an implicit free-surface model, which constrains its accuracy to asymptotically approach second order when $\theta \to 1/2$. Therefore, a split-explicit model can be made inherently more accurate in representing even the relatively slow barotropic motions resolved by the baroclinic time step (*e.g.*, tides and topographic Rossby waves) than an implicit model.

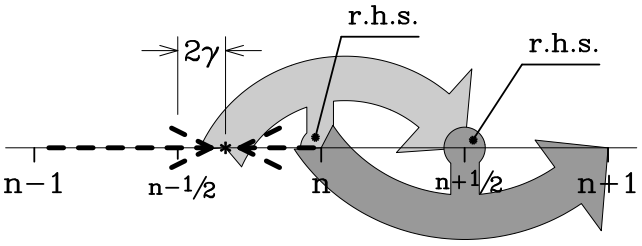


Fig. 17. Schematic diagram explaining the alternative LF–AM3 step: At first, $(n-1)$th and nth-step variables are interpolated linearly to $n-1/2+2\gamma$, which is used as the initial condition. It is advanced to $n+1/2$ using r.h.s. terms computed at nth step (predictor; $\gamma = 1/12$). Subsequently, the nth field is advanced to $n+1$ using the r.h.s. at $n+1/2$ (corrector).

4 Time-Stepping the Nonlinear System

4.1 Implementation of LF–AM3

The time-stepping algorithms in Sec. 2 are multi-time-level methods, relying on temporal interpolation or extrapolation of the r.h.s. terms computed at several consecutive steps to achieve the desired accuracy. This in principle can be applied to nonlinear systems as well (Canuto *et al.*, 1988): compute and store the entire nonlinear r.h.s. at discrete time levels and interpolate it using these fields. On the other hand, mode-splitting in Sec. 3 restricts the choice of time-stepping algorithms to logically forward-in-time, two-time-level methods where the only available degree of freedom is the time placement of the tracer flux variables in (3.3) and similar quantities in momentum equations. Since tracer fluxes are products of volume fluxes and tracer values and volume fluxes are constrained by (3.11) to satisfy the finite-volume continuity equation (3.4), it is no longer possible to compute the complete tracer r.h.s. tendency terms at several consecutive time steps and interpolate the result. Therefore, the algorithms from Sec. 2 must be adjusted for compatibility with mode-splitting.

The LF–AM3 scheme (2.21)-(2.22) is rewritten as

$$\begin{aligned}
\zeta^{n+\frac{1}{2}} &= \left(\frac{1}{2}-2\gamma\right)\zeta^{n-1} + \left(\frac{1}{2}+2\gamma\right)\zeta^{n} - i\alpha\,(1-2\gamma)\,u^{n} \\
u^{n+\frac{1}{2}} &= \left(\frac{1}{2}-2\gamma\right)u^{n-1} + \left(\frac{1}{2}+2\gamma\right)u^{n} - i\alpha\left[(1-2\gamma)\,\zeta^{n} + \beta\left(2\zeta^{n+\frac{1}{2}} - 3\zeta^{n} + \zeta^{n-1}\right)\right],
\end{aligned} \tag{4.1}$$

followed by

$$\begin{aligned}
\zeta^{n+1} &= \zeta^{n} - i\alpha\cdot u^{n+\frac{1}{2}} \\
u^{n+1} &= u^{n} - i\alpha\cdot\left\{(1-\epsilon)\,\zeta^{n+\frac{1}{2}} + \epsilon\left[\left(\frac{1}{2}-\gamma\right)\zeta^{n+1} + \left(\frac{1}{2}+2\gamma\right)\zeta^{n} - \gamma\zeta^{n-1}\right]\right\},
\end{aligned} \tag{4.2}$$

after which the provisional values $\zeta^{n+\frac{1}{2}}$ and $u^{n+\frac{1}{2}}$ are discarded. This alternative algorithm has a simple geometrical interpretation as a combination of interpolation and two consecutive LF-like steps (Fig. 17). It eliminates the need to store the full r.h.s. terms from one time step to another, making the code more efficient. It is completely equivalent to the original algorithm if applied to a linear system (***n.b.***, for the actual problem the symbolic operator $i\alpha[...]$ here translates into a r.h.s. computation), while for a nonlinear system it differs by computing r.h.s. terms from the time-interpolated prognostic variables rather than an interpolation of the complete r.h.s. fields.

A comparison with LF–TR stepping, *i.e.*, (4.1)-(4.2) with $\gamma = 0$, offers another interpretation of Fig. 17: the 2γ bias relatively to $n-1/2$ in setting the initial condition introduces a pre-distortion that cancels the second-order truncation errors of the subsequent "logically-LF" corrector stage, yielding an overall third-order accuracy of the algorithm as a whole.

Another difficulty with LF–AM3 is that the fluxes satisfying the discrete continuity equation (3.4) are available only during the corrector time step, not predictor step. Hence, it is impossible to achieve simultaneous conservation and constancy preservation for tracers during a predictor sub-step. Since the predicted values of the prognostic variables are used only to compute advective fluxes during the subsequent corrector step, the predictor sub-step does not necessarily have to be a conservative algorithm for the complete step to be conservative. A non-conservative, pseudo-compressible, predictor sub-step for tracers is

$$q^{n+\frac{1}{2}}_{i,j,k} = \frac{1}{\Delta\mathscr{V}^{+}_{i,j,k}}\Bigg\{\Delta\mathscr{V}^{-}_{i,j,k}\left[\left(\frac{1}{2}+2\gamma\right)q^{n}_{i,j,k}+\left(\frac{1}{2}-2\gamma\right)q^{n-1}_{i,j,k}\right]-(1-2\gamma)\,\Delta t\,\Big[\widetilde{q}^{n}_{i+\frac{1}{2},j,k}U^{n}_{i+\frac{1}{2},j,k} \\ -\widetilde{q}^{n}_{i-\frac{1}{2},j,k}U^{n}_{i-\frac{1}{2},j,k}+\widetilde{q}^{n}_{i,j+\frac{1}{2},k}V^{n}_{i,j+\frac{1}{2},k}-\widetilde{q}^{n}_{i,j-\frac{1}{2},k}V^{n}_{i,j-\frac{1}{2},k} \\ +\widetilde{q}^{n}_{i,j,k+\frac{1}{2}}W^{n}_{i,j,k+\frac{1}{2}}-\widetilde{q}^{n}_{i,j,k-\frac{1}{2}}W^{n}_{i,j,k-\frac{1}{2}}\Big]\Bigg\}, \tag{4.3}$$

where $\Delta\mathscr{V}^{\pm}_{i,j,k}$ is obtained from an artificial continuity equation,

$$\Delta\mathscr{V}^{\pm}_{i,j,k} = \Delta\mathscr{V}^{n}_{i,j,k} \mp \left(\tfrac{1}{2}-\gamma\right)\Delta t\,\Big[U^{n}_{i+\frac{1}{2},j,k}-U^{n}_{i-\frac{1}{2},j,k}+V^{n}_{i,j+\frac{1}{2},k}-V^{n}_{i,j-\frac{1}{2},k} \\ +W^{n}_{i,j,k+\frac{1}{2}}-W^{n}_{i,j,k-\frac{1}{2}}\Big]\,. \tag{4.4}$$

The latter "absorbs" incompressibility errors in $U^{n}_{i+\frac{1}{2},j,k}$, $V^{n}_{i,j+\frac{1}{2},k}$, and $W^{n}_{i,j,k+\frac{1}{2}}$. The result is a conservative, constancy-preserving algorithm for $q^{n+\frac{1}{2}}_{i,j,k}$. Once the computation for $q^{n+\frac{1}{2}}_{i,j,k}$ is completed, $\Delta\mathscr{V}^{\pm}_{i,j,k}$ is discarded and recomputed during the next time step. Because there is no guarantee that $\Delta\mathscr{V}^{+}_{i,j,k}$ is the same as $\Delta\mathscr{V}^{-}_{i,j,k}$ during the next time step, (4.3) does not maintain the volume, $\sum_{i,j,k}\Delta\mathscr{V}_{i,j,k}q^{n+\frac{1}{2}}_{i,j,k}$. However, the complete algorithm — (4.3) in combination with corrector step via (3.3) — does.

4.2 Implementation of AB3–AM4

The AB3–AM4 forward-backward scheme (2.32) is the method of choice for the barotropic mode because the time-step restriction imposed by the phase speed of barotropic waves dominates all other limitations (*i.e.*, advection velocity and Coriolis frequency) by such a large degree, that the other terms receive no consideration except for avoiding unconditionally unstable schemes. Its practical version consists of an AB3-extrapolation of prognostic variables,

$$\begin{pmatrix}\zeta\\ \overline{\mathbf{u}}\end{pmatrix}^{m+\frac{1}{2}} = \left(\frac{3}{2}+\beta\right)\begin{pmatrix}\zeta\\ \overline{\mathbf{u}}\end{pmatrix}^{m} - \left(\frac{1}{2}+2\beta\right)\begin{pmatrix}\zeta\\ \overline{\mathbf{u}}\end{pmatrix}^{m-1} + \beta\begin{pmatrix}\zeta\\ \overline{\mathbf{u}}\end{pmatrix}^{m-2}, \tag{4.5}$$

computation of finite-volume fluxes,

$$D^{m+\frac{1}{2}} = h+\zeta^{m+\frac{1}{2}}\,, \qquad \overline{U}^{m+\frac{1}{2}} = D^{m+\frac{1}{2}}\Delta\eta\,\overline{u}^{m+\frac{1}{2}}\,, \qquad \overline{V}^{m+\frac{1}{2}} = D^{m+\frac{1}{2}}\Delta\xi\,\overline{v}^{m+\frac{1}{2}}\,, \tag{4.6}$$

free-surface update,

$$\zeta^{m+1} = \zeta^{m} - \Delta t_{*}\cdot\mathrm{div}\overline{\mathbf{U}}^{m+\frac{1}{2}}\,; \tag{4.7}$$

computation of provisional ζ for the PGF,

$$\zeta' = \left(\frac{1}{2} + \gamma + 2\epsilon\right)\zeta^{m+1} + \left(\frac{1}{2} - 2\gamma - 3\epsilon\right)\zeta^{m} + \gamma\zeta^{m-1} + \epsilon\zeta^{m-2}\,; \tag{4.8}$$

and the momentum step,

$$\overline{\mathbf{u}}^{m+1} = \frac{1}{D^{m+1}}\left\{D^{m}\overline{\mathbf{u}}^{m} + \Delta t_{*}\cdot\left[\mathscr{F}\left(\zeta'\right) - D^{m+\frac{1}{2}}f\mathbf{k}\times\overline{\mathbf{u}}^{m+\frac{1}{2}} + ...\right]\right\}. \tag{4.9}$$

In the last step, the PGF $\mathscr{F}\left(\zeta'\right)$ is from (3.28), and the dots denote the other r.h.s. terms (advection, viscous diffusion, *etc.*). This algorithm naturally accommodates advection (centered or upstream-biased) and the Coriolis force; it is stable without the need for viscosity or upstream-bias for the $\overline{\mathbf{U}}$ terms in the ζ equation; and it eliminates the need to store r.h.s. terms from one time step to another.

A similar algorithm is applied for 3D mode in Kanarska *et al.* (2007), except that unlike (4.5)–(4.9), it starts with the update of momentum equation followed by the update of tracers. In that approach the tracer fields were actually extrapolated toward $(n+1/2)$th step twice using two different sets of AB3-like coefficients: the first time to compute density and then baroclinic pressure gradient (using coefficients optimized for stability of forward-backward step), and the second time to compute advection terms for tracer equations (using coefficients chosen more close to the conventional AB3 set). This dual extrapolation removes the competitive requirements in setting of β in (2.32) discussed in Sec. 2.4.

5 Pressure-Gradient Force

The discrete PGF error for a hydrostatic model in generalized vertical coordinates (including the σ family, *e.g.*, ROMS) is widely recognized as a significant algorithmic problem (Mesinger, 1982; Arakawa & Suarez, 1983; Mesinger & Janjic, 1985; Blumberg & Mellor, 1987; Haney, 1991; Mellor *et al.*, 1994; Stelling & van Kester, 1994; Lin, 1997; Slordal, 1997; Song, 1998; Song & Wright, 1998; Kliem & Pietrzak, 1999; Shchepetkin & McWilliams, 2003; Chu & Fan, 2003). It is often attributed to so-called hydrostatic inconsistency, *i.e.*, a failure of the discretized PGF to vanish when isopycnic surfaces are horizontal. Because of deviation of quasi-horizontal coordinates from either geopotential-height (z) or isopycnic (ρ) surfaces, the PGF in the horizontal momentum equations appears in the form of two large terms that tend to cancel each other,

$$-\frac{1}{\rho_0}\left.\frac{\partial P}{\partial x}\right|_{z} = -\frac{1}{\rho_0}\left[\left.\frac{\partial P}{\partial x}\right|_{s} - \frac{\partial P}{\partial z}\cdot\left.\frac{\partial z}{\partial x}\right|_{s}\right]. \tag{5.1}$$

In the usual way the partial-derivative subscript z means that it is computed with respect to a constant z surface, and the subscript s means that the differentiation is performed along the isosurface of the transformed vertical coordinate, $s = \mathrm{const}$.

The most common focus has been on achieving accurate cancellation of the two terms in (5.1) in the special case of a horizontally uniform (*i.e.*, flat) stratification, $\rho = \rho(z)$, where the correct answer is zero velocity (a state of rest). In this context Mellor *et al.* (1998) points out a Sigma-coordinate Error of the Second Kind (SESK), which is the growth in time of a mainly barotropic flow with no mechanism of advective self-compensation (in contrast to a baroclinic tendency to redistribute horizontal ρ surfaces by a flow generated by the PGF error to partially cancel the artificial flow). A small initial error does not guarantee that the error remain small at a later time. This experience brought attention to the integral

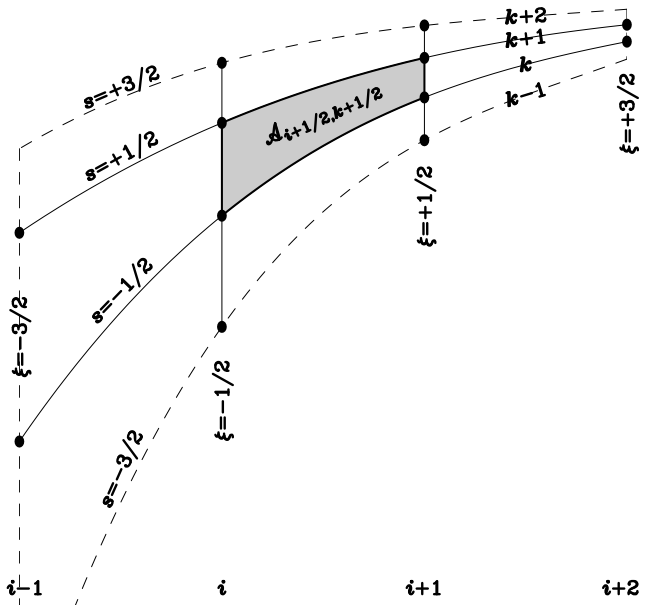


Fig. 18. Stencil in the x-z plane for computing the baroclinic PGF in the density-Jacobian scheme. The Jacobian is approximated as a contour integral around the shaded area $\mathscr{A}_{i+1/2,k+1/2}$, $-\Delta x\,\Delta z\cdot\mathscr{J}_{x,s}(\rho,z)=\oint\rho(x,z)\,\mathbf{dz}$. The contour integral is approximated using one-dimensional cubic polynomial fits for both $\rho(x,z)$ and z as functions of the coordinates x and s along each of the four curvilinear facets bounding $\mathscr{A}_{i+1/2,k+1/2}$. Since a cubic fit requires a 4-point one-dimensional stencil, the whole Jacobian is evaluated using 12-points: a 4×4 grid without corners. Each of the line integrals FX, FC, (5.4) participates in the computation of the density Jacobians for the two cell adjacent in either horizontally or vertically. The Jacobians are then integrated (via a simple summation) to compute PGF.

properties such as material conservation and consistent conversion between potential and kinetic energy. Despite the vast published experience there is not yet a consensus approach nor resolution of the problem. The approaches tend to fall into four major categories: (*i*) increase the order of accuracy in all coordinate directions(Beckmann & Haidvogel, 1993; Chu & Fan, 2003); while this can be quite successful in idealized test cases, it has earned a reputation of being useless for realistic oceanic modeling (Kliem & Pietrzak, 1999); (*ii*) compute the PGF in z-coordinate space (Kliem & Pietrzak (1999) and its references); (*iii*) use a finite-volume, flux-form, pressure-Jacobian formulation Lin (1997); Chu & Fan (2003); or (*iv*) use a density-Jacobian discretization of an alternative form for PGF that computes the horizontal ρ gradient first then integrates it vertically (Blumberg & Mellor, 1987; Song, 1998). The last two approaches rely on symmetric discretizations, mimicking the symmetries of the Jacobian operator, to reduce PGF error.

We found a successful technique to reduce PGF error. It is a generalization of the density-Jacobian approach going to higher-order accuracy while retaining most of the symmetries of its original schemes (Shchepetkin & McWilliams, 2003). It can also be viewed as a polynomial reconstruction of the ρ field with subsequent analytical contour integration. A similar approach was applied to construct a high-order analog of the pressure-Jacobian in Lin (1997); however, the generalized density-Jacobian is more attractive because of smaller truncation error and, more importantly, slower error growth in time for the flat stratification test cases. In this method the PGF is formulated (similar to Blumberg & Mellor (1987)) as

$$-\frac{1}{\rho_0}\frac{\partial P}{\partial x}\bigg|_z=-\frac{1}{\rho_0}\frac{\partial P}{\partial x}\bigg|_{z=\zeta}-\frac{g}{\rho_0}\int_z^\zeta\frac{\partial\rho}{\partial x}\bigg|_z dz'=-\frac{g}{\rho_0}\rho\bigg|_{z=\zeta}\frac{\partial\zeta}{\partial x}-\frac{g}{\rho_0}\int_z^\zeta\left[\frac{\partial\rho}{\partial x}\bigg|_s-\frac{\partial\rho}{\partial z'}\,\frac{\partial z'}{\partial x}\bigg|_s\right]dz'\,,$$

where the last term can be rewritten as

$$-\frac{g}{\rho_0}\int_s^0\left[\frac{\partial\rho}{\partial x}\bigg|_s\frac{\partial z}{\partial s}-\frac{\partial\rho}{\partial s}\,\frac{\partial z}{\partial x}\bigg|_s\right]ds'=-\frac{g}{\rho_0}\int_s^0\mathscr{J}_{x,s}(\rho,z)\,ds'\,,\tag{5.2}$$

to justify the classification of the scheme as a density-Jacobian type. The transformed vertical coordinate $s\in[-1,0]$ is assumed in (3.5) to be neither isopycnic nor geopotential so that both terms inside the left-side integral in (5.2) are nontrivial. To discretize this we introduce a control element $\mathscr{A}_{i+1/2,k+1/2}$ (the shaded area in Fig. 18) and apply Green's theorem,

$$-\Delta x\,\Delta z\cdot\mathscr{J}_{x,s}(\rho,z)\Big|_{i+\frac{1}{2},k+\frac{1}{2}}\approx\iint\limits_{\mathscr{A}}\mathscr{J}_{x,s}(\rho,z)\,dz\,dx=\oint\limits_{\partial\mathscr{A}}\rho\,\mathbf{dz}=FX_{i,k}-FX_{i+1,k}\\ +FC_{i+\frac{1}{2},k+1}-FC_{i+\frac{1}{2},k}\,. \tag{5.3}$$

FX and FC are the line integrals along the vertical and quasi-horizontal curvilinear segments bounding $\mathscr{A}_{i+1/2,k+1/2}$,

$$FX_{i,k+\frac{1}{2}}=\int\limits_{z_{i,k}}^{z_{i,k+1}}\rho\,dz\,,\qquad\qquad FC_{i+\frac{1}{2},k}=\int\limits_{(x,s)_{i,k}}^{(x,s)_{i+1,k}}\rho\,\frac{\partial z}{\partial x}\bigg|_s\,dx\,. \tag{5.4}$$

The problem thus reduces to interpolations for $\rho=\rho(x,s)$ and $z=z(x,s)$ along the integration contours. If linear interpolation is used for both ρ and z, the resultant scheme is equivalent to Blumberg & Mellor (1987). The natural extension is to use a cubic polynomial interpolation,

$$\rho(\xi)=\frac{\rho_j+\rho_{j+1}}{2}-\frac{d_{j+1}-d_j}{8}+\left[\frac{3}{2}\left(\rho_{j+1}-\rho_j\right)-\frac{d_j+d_{j+1}}{4}\right]\xi+\frac{d_{j+1}-d_j}{8}\xi^2\\ +\left[d_j+d_{j+1}-2\left(\rho_{j+1}-\rho_j\right)\right]\xi^3\,, \tag{5.5}$$

where ξ defined for $-\frac{1}{2}\le\xi\le+\frac{1}{2}$ is either x or s; the index j is either i or k (see Fig. 18), and by construction

$$\rho\left(\xi\right)\Big|_{\xi=-\frac{1}{2}}\equiv\rho_j\qquad\rho\left(\xi\right)\Big|_{\xi=+\frac{1}{2}}\equiv\rho_{j+1}\qquad\frac{\partial\rho}{\partial\xi}\bigg|_{\xi=-\frac{1}{2}}\equiv d_j\qquad\frac{\partial\rho}{\partial\xi}\bigg|_{\xi=+\frac{1}{2}}\equiv d_{j+1} \tag{5.6}$$

yields ρ values and derivatives at the side boundary of $\mathscr{A}_{i+1/2,k+1/2}$. Once (5.5) interpolates both ρ and z, the segment integrals (5.4) are evaluated analytically in terms of $z_{i,k}$, $\rho_{i,k}$, and their first spatial derivatives at the same location (see Shchepetkin & McWilliams (2003) for full formulas).

The most important issue is the estimator for the derivative d_j, especially if ρ is not smooth on the grid. Using an algebraically-averaged slope, the formula,

$$d_j=\frac{\Delta\rho_{j+\frac{1}{2}}+\Delta\rho_{j-\frac{1}{2}}}{2}\qquad\text{where}\qquad\Delta\rho_{j+\frac{1}{2}}\equiv\rho_{j+1}-\rho_j\qquad\forall j\,, \tag{5.7}$$

is sufficient to achieve the desired order of accuracy with a smooth ρ field and nearly uniform grid spacing. However, if ρ is not smooth, it admits spurious oscillations of the interpolant (5.5) that contaminate the PGF scheme as negative stratification patches, even when grid-point stratification values are positive everywhere, and this may result in numerical instability. In addition, (5.7) loses second-order accuracy if the grid spacing is not uniform. In contrast, a harmonic average,

$$d_j=\begin{cases}\dfrac{2\Delta\rho_{j+\frac{1}{2}}\Delta\rho_{j-\frac{1}{2}}}{\Delta\rho_{j+\frac{1}{2}}+\Delta\rho_{j-\frac{1}{2}}} & \text{if}\quad\Delta\rho_{j+\frac{1}{2}}\Delta\rho_{j-\frac{1}{2}}>0\\ 0 & \text{otherwise}\,,\end{cases} \tag{5.8}$$

has the property that if $\Delta\rho_{j+\frac{1}{2}}$ and $\Delta\rho_{j-\frac{1}{2}}$ have the same sign, d_j is no greater than twice the smaller of the two in magnitude; *i.e.*,

$$|d_j| < 2\left|\text{minmod}\left(\Delta\rho_{j+\frac{1}{2}}\,,\Delta\rho_{j-\frac{1}{2}}\right)\right|\,. \tag{5.9}$$

This guarantees that $\rho(\xi)$ in (5.5) is a monotonic, continuous function over the whole area of its definition.

Harmonic averaging (5.8) also escapes the loss of accuracy associated with non-uniformity of the vertical grid. This is extremely valuable for oceanic modeling since it is a common practice to choose only a moderate number of vertical levels with a grid spacing Δz that may change by as much as two orders of magnitude over the vertical column. Suppose that discretized values ρ_k are defined at locations z_k, such that $\Delta z_{k+\frac{1}{2}} \equiv z_{k+1} - z_k \neq \Delta z_{k-\frac{1}{2}} \equiv z_k - z_{k-1}$. A Taylor series expansion around z_k gives

$$\rho_{k\pm1} = \rho_k \pm \rho'\Delta z_{k\pm\frac{1}{2}} + \frac{1}{2}\rho''\Delta z^2_{k\pm\frac{1}{2}} \pm \frac{1}{6}\rho'''\Delta z^3_{k\pm\frac{1}{2}} + \ldots\,, \tag{5.10}$$

$$\Rightarrow \quad \begin{cases} \partial\rho_{k+\frac{1}{2}} \equiv \dfrac{\rho_{k+1}-\rho_k}{\Delta z_{k+\frac{1}{2}}} = \rho' + \dfrac{1}{2}\rho''\Delta z_{k+\frac{1}{2}} + \dfrac{1}{6}\rho'''\Delta z^2_{k+\frac{1}{2}} + \ldots \\ \partial\rho_{k-\frac{1}{2}} \equiv \dfrac{\rho_k-\rho_{k-1}}{\Delta z_{k-\frac{1}{2}}} = \rho' - \dfrac{1}{2}\rho''\Delta z_{k-\frac{1}{2}} + \dfrac{1}{6}\rho'''\Delta z^2_{k-\frac{1}{2}} + \ldots \end{cases}\,. \tag{5.11}$$

This leads to a second-order accurate approximation for $\partial\rho/\partial z$ at the location z_k,

$$\left.\frac{\partial\rho}{\partial z}\right|_{z=z_k} = \frac{\Delta z_{k-\frac{1}{2}}\partial\rho_{k+\frac{1}{2}} + \Delta z_{k+\frac{1}{2}}\partial\rho_{k-\frac{1}{2}}}{\Delta z_{k+\frac{1}{2}} + \Delta z_{k-\frac{1}{2}}} = \rho' + \frac{1}{6}\rho'''\Delta z_{k+\frac{1}{2}}\Delta z_{k-\frac{1}{2}} + \mathcal{O}\left(\Delta z^3\right)\,, \tag{5.12}$$

which is just a linear interpolation of $\partial\rho$ on a non-uniform grid. On the other hand, since

$$\left.\frac{\partial\rho}{\partial z}\right|_{z=z_k} = \frac{\partial\rho/\partial s|_{s=s_k}}{\partial z/\partial s|_{s=s_k}}\,, \tag{5.13}$$

the use of (5.7) makes the estimator into

$$\left.\frac{\partial\rho}{\partial z}\right|_{z=z_k} = \frac{\Delta\rho_{k+\frac{1}{2}} + \Delta\rho_{k-\frac{1}{2}}}{\Delta z_{k+\frac{1}{2}} + \Delta z_{k-\frac{1}{2}}} = \rho' + \frac{1}{2}\rho''\left(\Delta z_{k+\frac{1}{2}} - \Delta z_{k-\frac{1}{2}}\right) + \mathcal{O}\left(\Delta z^2\right)\,. \tag{5.14}$$

This is only first-order accurate. It evaluates the derivative at the location $\left(z_{k+1}+z_{k-1}\right)/2$ rather than the desired z_k. In contrast, (5.13) and (5.8) applied to the elementary differences $\Delta\rho$ and Δz leads to

$$\left.\frac{\partial\rho}{\partial z}\right|_{z=z_k} = \frac{\Delta\rho_{k+\frac{1}{2}}\Delta\rho_{k-\frac{1}{2}}\left(\Delta z_{k+\frac{1}{2}} + \Delta z_{k-\frac{1}{2}}\right)}{\left(\Delta\rho_{k+\frac{1}{2}} + \Delta\rho_{k-\frac{1}{2}}\right)\Delta z_{k+\frac{1}{2}}\Delta z_{k-\frac{1}{2}}} = \frac{\partial\rho_{k+\frac{1}{2}}\partial\rho_{k-\frac{1}{2}}\left(\Delta z_{k+\frac{1}{2}} + \Delta z_{k-\frac{1}{2}}\right)}{\partial\rho_{k+\frac{1}{2}}\Delta z_{k+\frac{1}{2}} + \partial\rho_{k-\frac{1}{2}}\Delta z_{k-\frac{1}{2}}}\,. \tag{5.15}$$

We assume that $\Delta\rho_{k+\frac{1}{2}}$ and $\Delta\rho_{k-\frac{1}{2}}$ have the same sign and that $\rho = \rho(z)$ is sufficiently smooth on the grid scale to be accurately represented by a Taylor series. This essentially translates into the assumptions that

$$|\rho''\cdot\Delta z| \ll |\rho'| \qquad \text{and} \qquad \left|\rho'''\cdot\Delta z^2\right| \ll |\rho'| \tag{5.16}$$

since the high-order derivatives in the Taylor series are presumed to be finite. Substitution of (5.11) into (5.15) yields

$$\left.\frac{\partial\rho}{\partial z}\right|_{z=z_k} = \rho' \frac{\left(1+\frac{1}{2}\frac{\rho''}{\rho'}\Delta z_{k+\frac{1}{2}}+\frac{1}{6}\frac{\rho'''}{\rho'}\Delta z^2_{k+\frac{1}{2}}+...\right)\left(1-\frac{1}{2}\frac{\rho''}{\rho'}\Delta z_{k-\frac{1}{2}}+\frac{1}{6}\frac{\rho'''}{\rho'}\Delta z^2_{k-\frac{1}{2}}+...\right)}{1+\frac{1}{2}\frac{\rho''}{\rho'}\frac{\Delta z^2_{k+\frac{1}{2}}-\Delta z^2_{k-\frac{1}{2}}}{\Delta z_{k-\frac{1}{2}}+\Delta z_{k-\frac{1}{2}}}+\frac{1}{6}\frac{\rho'''}{\rho'}\frac{\Delta z^3_{k+\frac{1}{2}}+\Delta z^3_{k-\frac{1}{2}}}{\Delta z_{k-\frac{1}{2}}+\Delta z_{k-\frac{1}{2}}}+...}$$

$$= \rho' + \left(\frac{1}{6}\rho''' - \frac{1}{4}\frac{(\rho'')^2}{\rho'}\right)\Delta z_{k+\frac{1}{2}}\Delta z_{k-\frac{1}{2}} + \mathcal{O}\left(\Delta z^3\right) \tag{5.17}$$

indicating second-order accuracy of the estimator for $\partial\rho/\partial z|_{z=z_k}$. The leading order truncation term of (5.17) consists of two parts: the first one proportional to ρ''' is exactly the same as in (5.12), and the second is a nonlinear term,

$$-\frac{1}{4}\frac{(\rho'')^2}{\rho'}\Delta z_{k+\frac{1}{2}}\Delta z_{k-\frac{1}{2}} \approx -\rho'\left(\frac{\Delta\rho_{k+\frac{1}{2}}-\Delta\rho_{k-\frac{1}{2}}}{\Delta\rho_{k+\frac{1}{2}}+\Delta\rho_{k-\frac{1}{2}}}\right)^2 .$$

The second formula always tends to reduce the estimated derivative and acts as a slope limiter if consecutive differences change abruptly on the grid scale. Because the same interpolation algorithm is applied to ρ and z, the discrete Jacobian guarantees the symmetry, $\mathscr{J}_{x,s}(\rho,z) = -\mathscr{J}_{x,s}(z,\rho)$. Although PGF cannot be eliminated entirely, it can be verified that for flat stratification, the cancellation of terms in (5.1) is fourth-order accurate, and the new scheme is robustly tolerant of "hydrostatically inconsistent" grids with $(\Delta x/\Delta z)\cdot\partial z/\partial x|_s > 1$ (Haney, 1991).

6 Impact of Compressibility

The compressibility of seawater in the EOS raises two important design issues for oceanic models. The first is that the monotonicity constraint for $\rho(z)$ interpolation in (5.5) and (5.8) no longer guarantees positive stratification for the interpolated profile if the constraint is applied to the *in situ* ρ, even if the point-wise stratification is strictly positive. This is because the grid-scale smoothness of ρ is judged by the ratio of consecutive differences, $\Delta\rho_{k+\frac{1}{2}}$ and $\Delta\rho_{k-\frac{1}{2}}$, both containing a component associated with bulk compressibility (*i.e.*, a vertical change of *in situ* density that occurs even when potential temperature Θ and salinity S are spatially uniform). As a result, $\Delta\rho \approx -\Delta z\cdot g\rho_0/c_s^2 - \Delta z\cdot\rho_0 N^2/g$ (c_s is speed of sound and N is Brunt-Väisäla frequency), and the first term dominates under most oceanic conditions (Dukowicz, 2001). This obscures the detection of abrupt changes in stratification. The second issue is a consequence of the mode-splitting algorithm (3.21)-(3.28) where $\overline{\rho}$ and ρ_* do not change in fast time, being kept constant at a time centered at $n+\frac{1}{2}$ to achieve second-order temporal accuracy during the barotropic time-stepping. When ρ is compressible, it depends on ζ though hydrostatic effects on pressure P. These changes are unaccounted for in the barotropic integration and thus are an additional source of mode-splitting error.

6.1 Compressibility and Baroclinic PGF

The EOS for seawater expresses *in situ* ρ in terms of Θ, S, and P,

$$\rho = \rho\left(\Theta, S, P\right) . \tag{6.1}$$

For oceanic modeling *in situ* ρ is very interesting by itself, but it plays intermediate roles in several r.h.s. terms for prognostic variables, *viz.*, horizontal PGF, stratification in vertical mixing parameterizations, and inclination of neutral surfaces along which lateral mixing occurs. The Boussinesq approximation

replaces *in situ* ρ by a representative constant ρ_0 everywhere except in the gravitational force gravity; *i.e.,* it retains the "gravitational" ρ in the gravitational force, but it approximates the "inertial" ρ in the Lagrangian acceleration by a constant ρ_0 that can be absorbed into PGF and otherwise disappear from the model. This approximation limits the EOS exclusively to the three purposes stated above, and the model is only sensitive to adiabatic gradients of ρ (defined in (6.6) below), but not to ρ itself. A consequence of the Boussinesq approximation is the replacement of mass conservation with an equivalent volume conservation based on a constant inertial ρ_0.

A common OGCM approximation is the replacement of *in situ* P in (6.1) with its bulk background value $P_0 = -g\rho_0 z$, *viz.,*

$$\rho = \rho\left(\Theta, S, |z|\right) , \tag{6.2}$$

justified by $\rho - \rho_0 \ll \rho_0$. Free-surface, σ-coordinate models (Mellor, 1991; Robinson *et al.*, 2001; Shchepetkin & McWilliams, 2003) often use an EOS in the form,

$$\rho = \rho\left(\Theta, S, \zeta - z\right) , \tag{6.3}$$

that selectively includes the barotropic contribution to the P used in the EOS but disregards the baroclinic part. The motivation for this choice comes not entirely from a physical consideration (*i.e.,* $g\rho_0\zeta$ is often small compared to P), but more from a coding convenience where the vertical coordinate system is regenerated at every time step from ζ and then used in the EOS routine. The use of standard EOS schemes, either as (6.1) or (6.2), implies a nonlinear dependence of ρ on z even if Θ and S are spatially uniform. For σ-coordinate models with coarse vertical resolution (often with a grid size as large as 500 m in the abyss), compressibility can cause significant PGF errors through hydrostatic non-cancellation in (5.1) (Shchepetkin & McWilliams, 2003). This type of error also exists in isopycnic models due to the non-equivalence of isopycnic and neutral surfaces caused by compressibility (Hallberg, 2005).

Consider for simplicity an EOS form within the approximation class of (6.2),

$$\rho\left(\Theta, S, z\right) = \rho^{(0)} + \rho'_1\left(\Theta, S\right) + \sum_{m=1}^{\infty} \left(q_m^{(0)} + q'_m\left(\Theta, S\right)\right) \cdot z^m \, . \tag{6.4}$$

$\rho^{(0)}$, and $q_m^{(0)}$, $m = 1, 2, ...$, are constant background values chosen so that $\rho^{(0)} \gg \rho'$, $q_1^{(0)} \gg q'_1$, *etc.* In practice these are chosen by specifying representative constant values for Θ and S and treating (6.4) as a series expansion around them. $q_1^{(0)}$ is the same as $g\rho^{(0)}/c_0^2$ (with c_0 a background value for c_s). With this EOS form, the density-Jacobian (5.2) is

$$\mathscr{J}_{x,s}(\rho, z) = \mathscr{J}_{x,s}(\rho', z) + \sum_{m=1}^{\infty} \mathscr{J}_{x,s}(q'_m z) \cdot z^m \, . \tag{6.5}$$

Note that $\rho^{(0)}$ and $q_m^{(0)}$ contribute nothing.

Jackett & McDougall (1995) defined *in situ* adiabatic derivatives of ρ as differences of potential density with a local reference pressure (*n.b.,* it is impossible to define potential density with any global reference pressure as a meaningful basis for determining stratification, unlike with the EOS for an ideal gas). In terms of (6.4), the adiabatic derivative with respect to the s coordinate is

$$\left.\frac{\partial \rho\left(\Theta, S, z\right)}{\partial s}\right|_{\rm ad} = \frac{\partial \rho'_1\left(\Theta, S\right)}{\partial s} + \sum_{m=1}^{\infty} z^m \frac{\partial q'_m\left(\Theta, S\right)}{\partial s} \, . \tag{6.6}$$

A similar expression applies to the horizontal (along $s = \text{const}$) derivative $\partial\rho\,(\Theta, S, z)\,/\partial\xi|_{\text{ad}}$. The baroclinic PGF (6.5) can be expressed entirely in terms of *in situ* adiabatic derivatives of ρ. For comparison, substituting the EOS (6.2) into (5.2) yields

$$\mathscr{J}_{x,s}(\rho, z) = -\widehat{\alpha}\,\mathscr{J}_{x,s}(\Theta, z) + \widehat{\beta}\,\mathscr{J}_{x,s}(S, z)\,. \tag{6.7}$$

Here $\widehat{\alpha} = -\,\partial\rho\big/\partial\Theta|_{S,z}$ and $\widehat{\beta} = \partial\rho\big/\partial S|_{\Theta,z}$ are adiabatic thermal expansion and saline contraction factors (*n.b.*, these differ from the conventional α, β coefficients by an added ρ multiplier). On the other hand, if the exact EOS (6.1) is used instead of (6.2), then the r.h.s. of (6.7) has an additional term, $-(1/c_s^2)\,\mathscr{J}_{x,s}(P, z)$ (*i.e.*, $\propto \kappa$ in (6.17) below). This shows that the ability to express the baroclinic PGF entirely in terms of adiabatic ρ derivatives inherently relies on the EOS approximation $P \to z$ in (6.2). If the approximation in (6.2) is assumed valid (this aspect will be addressed in more detail in Sec. 6.3), then (6.7) indicates that the only requirement for accurately relating the gradients of Θ and S to the PGF is the correct computation of $\widehat{\alpha}$ and $\widehat{\beta}$, including their dependence on P or z (*i.e.*, thermobaric effect). The *in situ* ρ by itself is irrelevant. This is also seen by the independence of (6.5) from the background terms $\rho^{(0)}$ and $q_m^{(0)}$.

Most of vertical change of ρ and much of the horizontal (along $s = constant$) change occur due to the bulk compressibility terms, *i.e.*, $\partial\rho_{in\ situ}/\partial z \neq 0$ in (6.5). Consequently, a non-oscillatory profile of $\rho_{\text{in situ}}$ does not necessarily correspond to monotonic stratification. Therefore, it is meaningless to apply harmonic averaging (5.8) to consecutive differences of *in situ* ρ and to expect that monotonic positive stratification is guaranteed, even if the grid-box values of ρ are positively stratified. To achieve a monotonic stratification profile, we introduce elementary adiabatic differences, similar to (6.6) above; *e.g.*, for $m = 1$

$$\Delta\rho^{(\text{ad})}_{i,k+\frac{1}{2}} = \rho'_{1i,k+1} - \rho'_{1i,k} + \left(q'_{1i,k+1} - q'_{1i,k}\right)\frac{z_{i,k+1} + z_{i,k}}{2}\,. \tag{6.8}$$

The averaged gradient (5.8) translates into

$$d_{i,k} \equiv \left.\frac{\partial\rho}{\partial s}\right|_{i,k} = \frac{2\Delta\rho^{(\text{ad})}_{i,k+\frac{1}{2}}\cdot\Delta\rho^{(\text{ad})}_{i,k-\frac{1}{2}}}{\Delta\rho^{(\text{ad})}_{i,k+\frac{1}{2}} + \Delta\rho^{(\text{ad})}_{i,k-\frac{1}{2}}} \;+\; q'_{1i,k}\left.\frac{\partial z}{\partial s}\right|_{i,k}\,, \tag{6.9}$$

where the adiabatic and compressible parts are separated at first, interpolated separately, and recombined at the end. This guarantees monotonicity of stratification for the interpolated profile. Because of the nonlinearity in (6.9), the resulting PGF scheme is incompatible with the common practice of subtracting a horizontally uniform background profile $\rho_{\text{bak}} = \rho_{\text{bak}}(z)$ in an attempt to reduce σ-coordinate PGF error. Similarly, the use of (6.7) as a basis for the PGF scheme is not desirable because separate computations of the Jacobians for Θ and S cannot ensure monotonicity of stratification if the Θ and S profiles are interpolated separately. For example, if there is a "spice" anomaly with large, smooth Θ and S gradients largely canceling each other to yield a ρ gradient that is small but non-smooth on the grid scale, then the monotonicity algorithm separately applied to Θ and S will fail to detect the sudden change in the ρ gradient.

6.2 Compressibility and Barotropic Mode-Splitting

The mode-splitting algorithm described in Sec. 3.2 is derived using the assumption that ρ does not depend on ζ. This is no longer the case if the exact P dependence is included in the EOS (6.1) or even in its simplified version (6.3). Although the magnitude of the change is always small, a danger comes from

the sensitivity of the EOS to ζ that implies a PGF contribution when ζ is computed at the previous time step and kept constant during the barotropic time-stepping (*i.e.,* effectively receiving a forward-in-time treatment). Consider a purely barotropic case with ρ changes due only to compressibility,

$$\rho = \rho(P) = \rho_1^{(0)} + \sum_m q_m^{(0)} P^m \,, \tag{6.10}$$

where $\rho_1^{(0)}$ and are $q_m^{(0)}$ are spatially uniform. Without loss of generality, this can be replaced with

$$\rho = \rho_1^{(0)} + \sum_m q_m^{(0)} (\zeta - z)^m \,, \tag{6.11}$$

because the hydrostatic balance, $\partial P/\partial z = -g\rho$, makes it possible to remap (6.10) into (6.11) with an alternative set of coefficients q_m (*e.g.,* $\rho = \rho_1 + q_1 P$ translates into $\rho = \rho_1 \exp\{g q_1 (\zeta - z)\}$) [12] . A derivation similar to (3.23) yields the net PGF applied to the fluid element (Fig. 14),

$$\mathscr{F}_{i+\frac{1}{2}} = g\,(\zeta_i - \zeta_{i+1}) \left[\rho_1^{(0)} \frac{D_i + D_{i+1}}{2} + \sum_m q_m^{(0)} \frac{D_i^{m+1} + D_i^m D_{i+1} + ... + D_i D_{i+1}^m + D_{i+1}^{m+1}}{(m+1)(m+2)} \right] . \tag{6.12}$$

This corresponds to the continuous form,

$$-g \left[\rho_1^{(0)} D + \sum_m q_m^{(0)} \frac{D^{m+1}}{m+1} \right] \nabla_x \zeta \equiv -g \overline{\rho} D \nabla_x \zeta \,, \tag{6.13}$$

where $\overline{\rho} = \rho_1^{(0)} + \sum_m q_m^{(0)} \dfrac{D^m}{m+1}$ is identified as the vertically averaged ρ. Therefore, we conclude that, if ρ non-uniformity is caused exclusively by compressibility, then $\nabla_x \zeta$ generates exactly the same acceleration,

$$\frac{1}{\overline{\rho} D} \frac{\partial}{\partial t} \int_{-h}^{\zeta} \rho u \, dz + ... = -g \nabla_x \zeta \,, \tag{6.14}$$

as in a uniform-density, shallow-water fluid. Furthermore, the acceleration by the full PGF, $-\dfrac{1}{\rho}\nabla_x P = -g\nabla_x \zeta$, is independent of depth throughout the vertical column even though both $P = P(z)$ and $\rho = \rho(z)$ are nonlinear functions of z; hence, a purely barotropic (*i.e.,* vertically uniform) flow can remain barotropic.

Note that (6.11) is similar to (6.4), except that now it is expanded in powers of perturbed depth $\zeta - z$, rather than just z, and therefore, from (6.11), $\nabla_x \rho \neq 0$ as long as $\nabla_x \zeta \neq 0$. Still the $q_m^{(0)}$-terms in the EOS do not change the acceleration caused by the PGF. Here — unlike in the baroclinic case (6.4)-(6.5) — the absence of spurious acceleration by the barotropic PGF is valid only in the non-Boussinesq case, with $\overline{u}, \overline{v}$ defined as ρ-averaged rather than z-averaged velocity. The Boussinesq replacement of the inertial *in situ* ρ with ρ_0 creates a spurious multiplier ρ/ρ_0 in the PGF that destroys this property [13] . At a first glance (6.14)

[12] Another consequence of this $P \leftrightarrow z$ remapping is that it eliminates acoustic waves regardless whether or not the Boussinesq approximation is used. This makes it possible to build a hydrostatic, non-Boussinesq codes with relatively small additional effort. Non-hydrostatic, non-Boussinesq models must use other means to deal with unwanted acoustic waves (*e.g.,* implicit time-stepping, or the use of anelastic approximation), that may cause in a dramatic increase in code complexity.

[13] This situation is similar to Case A of Dewar *et al.* (1998) discussed in Sec. 6.3 but in reverse: the dependency $\rho = \rho(P)$ in (6.21) brings in PGF error when used within the modified Boussinesq model.

suggests that taking into account ρ non-uniformity in the barotropic mode with ρ^* and $\overline{\rho}$ in (3.28) offers no benefit relative to the use of the shallow-water-like PGF term, $-gD\nabla_x\zeta$. However, (3.28) and (6.14) are derived under two opposite assumptions about the ρ structure: (3.28) assumes that the ρ non-uniformity comes purely from baroclinic effects, and the flow is incompressible, hence ρ is conserved as Lagrangian tracer; whereas (6.14) assumes that all non-uniformity comes exclusively from the bulk compressibility. Besides the spurious ρ/ρ_0 factor, we identify two types of dangerous error: (**i**) the mode-splitting error due to the $\rho = \rho(..., \zeta - z)$ dependency, since the computation of the 3D PGF in (3.21) is based on the previous-time ζ and thus receives a forward-in-time treatment; and (**ii**) an erroneous sensitivity of ρ_* and $\overline{\rho}$ to the vertical increase of *in situ* ρ by bulk compressibility that is mistaken for vertical stratification.

The magnitude of the mode-splitting error of type (**i**) is estimated from the vertical integral of the PGF due to ζ modulated by compressibility,

$$-g\nabla_x\zeta \cdot \int_{-h}^{\zeta} \exp\left\{\frac{g\,(\zeta - z')}{c_s^2}\right\} dz' \approx -\underbrace{gD\nabla_x\zeta}_{\text{“fast”}} - \underbrace{\frac{1}{2}\cdot\frac{gD}{c_s^2}\cdot gD\nabla_x\zeta}_{\text{“slow”}}\,. \tag{6.15}$$

The “fast” term is treated within the barotropic mode using a small time step. The “slow” term is never computed explicitly, but is rather an outcome of computing the vertical integral of 3D PGF based on ρ with the EOS using the ζ at the old time step — the most recent available value before barotropic time-stepping begins. As a result it remains unchanged during barotropic time-stepping even though it contains a gradient of ζ. $D = 5$ km and $c_s = 1500$ m/s yield an error estimate of $gD/(2c_s^2) = 0.01$, about 1% of the PGF due to the ζ perturbation. This is comparable with levels of other mode-splitting errors discussed in Sec. 3.2. It is expected to stay within the Courant-number limit of baroclinic (slow) time-stepping, leaving its forward-in-time treatment as the primary remaining concern. This type of splitting occurs whether or not the barotropic mode accounts for ρ non-uniformity, and furthermore, it occurs in non-Boussinesq models as well. For example, Robinson *et al.* (2001) identifies a similar error (although they do not classify it as mode-splitting error) and an associated instability in a model that uses a shallow-water form for the PGF in the barotropic mode. The instability is manifested as a tidal response with spuriously elevated amplitude. The source of instability is traced back to an inconsistency between ρ and the horizontally-averaged $\rho(z)$ profile (subtracted out in hopes of reducing PGF error); the former is computed using instantaneous ζ and the latter using $\zeta = 0$. Their proposed remedies include abandonment of the averaged $\rho(z)$ subtraction — a relatively minor effect — and total suppression of compressibility in the EOS — sufficient to suppress the instability but not acceptable in OGCMs because of loss of the thermobaric effect. Griffies *et al.* (2001) and McDougall *et al.* (2002) advocate the use of the exact EOS (6.1) with a P that includes dynamic components due to both ζ and ρ taken from the previous time step. However, we believe that this brings a similar mode-splitting error and potential instability that most likely is only controlled by their heavy barotropic-mode time-filtering by averaging over two baroclinic time steps (Sec. 3.3).

A better treatment for both type (**i**) and (**ii**) errors is presented in Sec. 6.3 after an analysis of alternative forms for the EOS.

6.3 Consistency of EOS and Boussinesq Approximation

The EOS form (6.2) as an approximation to (6.1) was challenged by Dewar *et al.* (1998) [14] . Consider the response of a compressible barotropic fluid with uniform Θ and S to an imposed surface PGF $\nabla_x p_s$ (their Case A, Fig. 1). If (6.2) is used for the EOS, the PGF is constant and equal to its surface $\nabla_x p_s$ value throughout the vertical column. However, compressibility leads to changes in ρ, and if the EOS more correctly uses *in situ* P, the ρ changes depend on the PGF itself, and the true PGF will change with depth. Substituting their Eq. (2.3) into (2.2) yields

$$\nabla_x P = \nabla_x p_s + g \int_z^0 \frac{\nabla_x P}{c_s^2}\, dz' \,. \tag{6.16}$$

This has the solution $\nabla_x P = \nabla_x p_s \cdot \mathrm{e}^{-gz/c_s^2}$; *i.e.,* now the PGF has an exponential amplifier with depth. With typical abyssal values of $c_s = 1500m/s$ and $z = -5000m$, the amplification factor is about 1.022, which is comparable to a typical PGF error due to the Boussinesq approximation. However, the pressure gradient does not appear in the PGF by itself but in the combination $(1/\rho)\nabla_x P$. Thus, by using the exact *in situ* ρ that has the same compressibility amplifying factor, instead of the Boussinesq ρ_0 without it, the depth-amplification effect is canceled in the PGF. For example, the balancing geostrophic velocity (*cf.,* their Eq. (2.3)) does not change at all between a Boussinesq code with an approximate EOS and a non-Boussinesq code with the exact EOS. Their Cases B and C, Fig. 1, involve baroclinic variations of Θ and S. In contrast to the purely barotropic Case A, these cases do not have an exact cancellation of the compressibility errors. However, as shown by Dukowicz (2001), more than 90% of the error can be eliminated by a further modification of the EOS, so the danger identified by Dewar *et al.* (1998) is largely avoidable.

The approach of Dukowicz (2001) splits the compressibility coefficient κ into two parts [15] ,

$$\kappa = \frac{1}{\rho}\left(\frac{\partial \rho}{\partial P}\right)_{\Theta,S} \,, \qquad \kappa = \kappa^{(P)}(P) + \delta\kappa(\Theta, S, P) \,, \tag{6.17}$$

where $\kappa^{(P)}$ is much larger than $\delta\kappa$. The exact EOS (6.1) can be rewritten with two ρ factors,

$$\rho = r(P) \cdot \rho^{\bullet}(\Theta, S, P) \,. \tag{6.18}$$

Without any approximation the PGF, hydrostatic balance, and EOS can be alternatively be expressed in terms of $\rho^{\bullet}$ and a related pressure quantity $P^{\bullet}$:

---

[14] Although ROMS uses an intermediate approximation to EOS (6.3), this criticism is still a concern because of the absence of the $-(1/c_s^2)\mathscr{J}_{x,s}(P,z)$ term in (6.7) and its counterpart in (6.5). Secs. 5.1-5.2 of Shchepetkin & McWilliams (2003) introduce two PGF schemes. One computes the density-Jacobian directly and then integrates it vertically (hence, entirely avoiding computation of P), and the other is a primitive form that first explicitly computes P. These two schemes are identical for an incompressible EOS, but the statement that the PGF can be expressed entirely in terms of adiabatic ρ differences applies only to the first scheme. Unless the EOS is modified to exclude bulk compressibility, the primitive form implicitly contains an equivalent of the $-(1/c_s^2)\mathscr{J}_{x,s}(P,z)$ component.

[15] To avoid confusion with ρ^* in the barotropic PGF in Secs. 3.2 and 6.2, we modified the original notation of Dukowicz (2001) by $\rho^* \to \rho^{\bullet}$ and $P^* \to P^{\bullet}$.

$$\frac{1}{\rho}\nabla_x P \qquad \leftrightarrow \qquad \frac{1}{\rho^\bullet}\nabla_x P^\bullet \tag{6.19}$$

$$\frac{\partial P}{\partial z} = -g\rho \qquad \leftrightarrow \qquad \frac{\partial P^\bullet}{\partial z} = -g\rho^\bullet \tag{6.20}$$

$$\rho = \rho(\Theta, S, P) \qquad \leftrightarrow \qquad \rho^\bullet = \frac{\rho\Big(\Theta, S, P(P^\bullet)\Big)}{r(P^\bullet)} = \rho^\bullet(\Theta, S, P^\bullet)\,. \tag{6.21}$$

The relations in the right column have the same functional forms as the original ones in the left column, and the scaling factor $r(P)$ does not explicitly appear.

The practical value of the approximate EOS (6.4) or *a fortiori* the factored EOS (6.18), for oceanic simulations comes from a dramatic narrowing with depth of the range of realistic values for Θ and S (*cf.*, Fig. 19 in Shchepetkin & McWilliams (2003) and Fig. 2 in McDougall *et al.* (2003)). For the factored EOS form, $r(P)$ can be chosen so that $\kappa^{(P)}$ strongly dominates $\delta\kappa$ in (6.17) in the abyss; hence, the nonlinear dependence of ρ on P or z is mostly absorbed into $r(P)$, which is subsequently scaled out in the $\rho, P \;\rightarrow\; \rho^\bullet, P^\bullet$ transformation (6.21). In the upper ocean Θ and S are more variable, and factoring is not as effective in keeping $\delta\kappa$ small compared to $\kappa^{(P)}$; however, the nonlinear compressibility is not as important there, and useful approximations to the EOS can be made without sacrificing accuracy. We choose the definition,

$$r(P) = \rho_{\mathrm{JM95}}\left(\Theta_0, S_0, P\right) / \rho_{\mathrm{JM95}}\left(\Theta_0, S_0, 0\right) \tag{6.22}$$

where $\rho_{\mathrm{JM95}}(\Theta, S, P)$ refers to the particular form of the EOS in Jackett & McDougall (1995), and Θ_0 and S_0 are representative abyssal values (***e.g.,*** $\Theta_0 = 1.5$ and $S_0 = 34.74$ are good choices for global or basin-scale modeling). Then

$$\rho^\bullet = \rho_{\mathrm{JM95}}(\Theta, S, P)/r(P) \tag{6.23}$$

has a substantially narrower dynamical range than the original $\rho = \rho_{\mathrm{JM95}}(\Theta, S, P)$, and, even more importantly, it does not grow with P or z. In the terminology of Dukowicz (2001), this procedure "stiffens" the EOS. In a Boussinesq model based on (6.21), $\rho^\bullet$ is replaced with the reference value ρ_0 (***e.g.,*** $\rho_0 = 1027.8 kg/m^3$ is consistent with the Θ_0 and S_0 choices above and is closer to the actual $\rho^\bullet$ than the more widely used ρ_0 values of 1000 or 1025 kg/m^3). The *in situ* P used inside the EOS routine is approximated with a background $P_0(z)$ computed from

$$\frac{dP_0}{dz} = -g\rho_0 \cdot r(P_0)\,. \tag{6.24}$$

This approximates the EOS in (6.23) as

$$\rho^\bullet = \rho^\bullet(\Theta, S, z)\,. \tag{6.25}$$

This is the same functional form as (6.2), but it accounts for the main effect of ρ variation in computing P in the EOS; thus, it is closer to the exact EOS (6.1) in representing the changes of $\widehat{\alpha}$ and $\widehat{\beta}$ with depth. Finally, as in the PGF algorithm in Shchepetkin & McWilliams (2003), the resulting EOS (6.25) is split as in (6.4), except that the expansion in powers of z is replaced with a $(\zeta - z)$ expansion that is truncated after the linear term. To minimize round-off errors, the EOS is expressed as a perturbation relative to ρ_0.

This form of the EOS in (6.25) allows computation of adiabatic ρ differences (6.8). These are averaged with a harmonic mean (6.9) that is subsequently needed to construct cubic interpolants (5.5), segment integrals (5.4), and discrete density-Jacobian. The interpolant is guaranteed to maintain positive stratification as long as the discrete density field is positively stratified. Although removing the dominant part of the bulk compressibility, (6.21) makes point-wise differences of $\rho^\bullet$ much closer to adiabatic differences, one might be tempted to compute the baroclinic PGF directly from $\rho^\bullet$ without using adiabatic differencing. However, our experience has shown that this is neither sufficiently accurate in practice, nor robust when there are sudden changes in stratification.

The transformation (6.21) offers a natural, simple remedy to reduce the mode-splitting errors of both types (**i**) and (**ii**) in Sec. 6.2: the elimination of bulk passive compressibility in the EOS effectively removes the second r.h.s. term in (6.15), but unlike the remedy of Robinson *et al.* (2001), it retains a physically accurate representation of the thermobaric effect. Computing ρ^* and $\overline{\rho}$ from $\rho^\bullet$ is sufficient to eliminate their biases due to bulk compressibility, hence to avoid a type (**ii**) error.

Despite the multi-stage transformation described here, the functional forms of the EOS and PGF schemes in Shchepetkin & McWilliams (2003) remain unchanged, requiring only a refitting of the polynomial coefficients in the EOS [16] .

6.4 Accuracy of the Boussinesq Approximation

The accuracy and utility of using the Boussinesq approximation for an OGCM is assessed in several papers (McDougall & Garret, 1992; Greatbatch, 2001; McDougall *et al.*, 2002; Greatbatch & McDougall, 2003), identifying, among other issues, an inherent conflict between the assumption of constancy of ρ (hence replacement of mass conservation with volume conservation) and the need to use the fully compressible EOS that implies $\approx 5\%$ variation in ρ. This limits the accuracy of the Boussinesq approximation, and there has emerged a slow but steady advocacy for non-Boussinesq OGCMs (*e.g.*, Griffies *et al.* (2001) and the citations above).

In this situation Dukowicz (2001) stands out because it constitutes a revision of the Boussinesq approximation as traditionally applied to OGCMs that include a compressible EOS in an *ad hoc* manner, breaking the internal consistency of the Boussinesq approximation. The revision restores consistency by bringing the properties of the EOS close to that for an incompressible fluid while still including the thermobaric effect. This approach stays within the spirit of the Boussinesq approximation by making the approximate PGF close to the full non-Boussinesq version without explicitly including any non-uniformity of the inertial ρ. The bulk compressibility ratio $r(P)$ is not used anywhere except in the $P_0 \leftrightarrow z$ remapping (6.24)-(6.25) for the stiffened EOS, which brings a minor effect relative to the more traditional choice of replacing $P_0 = -\rho_0 g z$ in EOS.

This aspect of Dukowicz (2001) was criticized by McDougall *et al.* (2002) — in essence advocating discarding $r(P)$ — since it leaves "no choice but to interpret the horizontal velocity vector as the Eulerian-mean horizontal velocity, but not as the mass flux per unit area". This is viewed as a drawback because it prevents a re-interpretation of the prognostic variables in a Boussinesq model as density-weighted rather

[16] Although more recent and supposedly more accurate versions of EOS have become available, (McDougall *et al.*, 2003 ; Jackett *et al.*, 2006), the EOS functional form in Jackett & McDougall (1995), inherited from the UNESCO EOS, is preferable as the approximation standard because it is already close to the desired factored form, comprised of $\rho(\Theta, S)$ at 1 atm ($P = 0$ in our terms) multiplied by terms that account for compressibility effects. The rational functional form used in the newer EOS mixes P terms together with Θ and S terms and makes it harder to separate out P effects.

than Eulerian averages. When a solution reaches a stationary state the difference between the re-interpreted Boussinesq model and a non-Boussinesq model disappears (*cf.*, Sec. 4-5 in McDougall *et al.* (2002), as well as similar approaches for including some non-Boussinesq effects in Boussinesq models (Lu, 2000; Greatbatch, 2001)). This re-interpreted equivalence implies that the actual Boussinesq errors are less than the usual estimate of $\approx 5\%$ associated with the standard formulation. The use of $\rho^{\bullet}$ and $r(p)$ in a Boussinesq model prevent this re-interpretation.

This limitation of Dukowicz (2001) can be substantially mitigated in a finite-volume code by replacing $H_{i,j,k} = z_{i,j,k+\frac{1}{2}} - z_{i,j,k-\frac{1}{2}}$ in (3.5) with

$$H_{i,j,k} = \left(z_{i,j,k+\frac{1}{2}} - z_{i,j,k-\frac{1}{2}}\right) \cdot r\left[P_0\left(\zeta_{i,j} - \frac{z_{i,j,k+\frac{1}{2}} + z_{i,j,k-\frac{1}{2}}}{2}\right)\right] . \tag{6.26}$$

This replacement automatically, and without additional computational effort, implies a redefinition of the control volumes $\Delta\mathscr{V}_{i,j,k}$, interfacial contact surfaces, horizontal flux $(U_{i+\frac{1}{2},j,k},\ V_{i,j+\frac{1}{2},j,k})$, and vertical flux $W_{i,j,k+\frac{1}{2}}$ (3.9) as mass-weighted by $\rho = r(P_0(z))$. This yields the major part of non-uniform inertial ρ in transforming volume conservation into approximate mass conservation with $\int\int\int r(P_0(z))\, d^3\mathscr{V}$.

Density-Jacobian schemes use a contour integration to approximate $\Delta x \Delta z \cdot \mathscr{J}_{x,s}(\rho, z)$ which is then integrated vertically (via a simple summation) to compute point-wise pressure gradient. The later one is subsequently multiplied by a horizontally averaged $H_{i,j,k}$ to convert it into the force applied the the control volume. This makes the PGF be invariant with respect to a change of definition for $H_{i,j,k}$ from the original to (6.26) because the velocity component is also multiplied by the same $H_{i,j,k}$. The change in $H_{i,j,k}$ also leaves the transformation (6.21) unaffected. The analysis of Dukowicz (2001) only considers instantaneous errors associated with an inconsistent use of a fully compressible EOS in a Boussinesq model, but this is not a guarantee that the error will not grow in time. Recently, de Szoeke & Samelson (2002); Losch *et al.* (2004) pointed out that the hydrostatic, Boussinesq equations in z are isomorphic to the hydrostatic, non-Boussinesq equations in pressure coordinates. This implies that the solution differences between Boussinesq and non-Boussinesq models should stay bounded in time since P and z differences do so. Because (6.26) merely introduces a metric factor in the vertical coordinate while retaining the mathematical structure Boussinesq code, we expect that the Boussinesq errors using (6.26) also stay bounded.

The preceding discussion shows that the theoretical differences between Boussinesq and non-Boussinesq hydrostatic models are much less than the initial estimates of McDougall & Garret (1992) and Dewar *et al.* (1998). The differences can be further reduced by application of the transformation (6.21) in combination with the quasi-Boussinesq $r(P)$-remapping (6.26). The Boussinseq apprroximation offers an important advantage for a cleaner mode-spliting algorithm to avoid type (**i**) and (**ii**) errors (Sec. 6.2). Conversely, a more fundamental non-Boussinesq code does not escape the need to assure monotonic stratification profiles with higher-order Jacobian PGF schemes in generalized vertical coordinates and a compressible EOS that includes thermobaric effects. In summary, we do not presently see a strong case for preferring a non-Boussinesq OGCM.

7 Final Remarks

In this paper we have discussed many of the central algorithmic elements — the computational kernel — in an OGCM designed for large computations of highly turbulent flows. Our currently preferred choices

for these elements are summarized in Sec. 1 and discussed at length in the ensuing sections. A key aspect of OGCM design is the interplay among the kernel elements, with abundant possibilities for both destructive interference and constructive synergy. This perspective confounds any simple expectation that better code modularity is the principal software step toward better OGCMs: while a modular structure may facilitate code adaptation, the most important design consideration is the overall model performance in physical and numerical accuracy and computational efficiency.

The use of oceanic models has historically followed a path downward in scale, from basins and global domains to flows with smaller space and time scales and more turbulent dynamics. At the present time the simulation battle front is at mesoscale-eddy resolution, but we can anticipate continuing scale refinements through a combination of larger computers, further algorithmic advances, multi-scale (nested-grid) methods, and, of course, improved dynamical understanding of the simulated phenomena. We intend to participate in these developmental directions and mention, in closing, a newly constructed, non-hydrostatic version of ROMS (Kanarska *et al.*, 2007).

Acknowledgments: The authors appreciate the sustained support for model development research provided by the Office of Naval Research through its grants N00014-98-1-0165, N00014-00-1-0249, N00014-02-1-0236, and N00014-05-10293.

References:

Adcroft, A. and J.-M. Campin, 2004: Rescaled height coordinates for accurate free-surface flows in ocean circullation models. *Ocean Modelling*, **7**, 269-284. doi:10.1016/j.oceanmod.2003.09.003

Arakawa, A. and V. R. Lamb, 1977: Computational design of the basic dynamical processes of the UCLA General Circulation Model. *Meth. Comput. Phys.*, **17**, 174-267.

Arakawa, A., and M. J. Suarez, 1983, Vertical differencing of the primitive equations in sigma coordinates. *Monthly Weather Review*, **111**, 34-45.

Beckmann, A., and D. B. Haidvogel, 1993: Numerical simulation of flow around a tall isolated seamount. Part. 1: Problem formulation and model accuracy. *J. Phys. Ocean.*, **23**, 1736-1753.

Beckmann, A., and D. B. Haidvogel, 1997: A numerical simulation of flow at Fieberling Guyot. *J. Geophys. Res.*, **102**, 5595-5613.

Berntsen, H., Z. Kowalik, S. Sælid, and K. Sørli, 1981: Efficient numerical simulation of ocean hydrodynamics by a splitting procedure. *Modeling, Identification and Control*, **2**, No. 4, 181-199.

Bleck, R. and L. T. Smith, 1990: A wind-driven isopycnic coordinate model of the north and equatorial Atlantic Ocean: 1. Model development and supporting experiments. *J. Geophys. Res.*, **95C**, 3273-3285.

Blumberg, A. F. and G. L. Mellor, 1987: A description of a three-dimensional coastal ocean circulation model. In *Three-Dimensional Coastal Ocean Models*, ed. N. Heaps (Pub. AGU), 1-16.

Bryan, K., and M. Cox, 1969: A numerical method for the study of the circulation of the world ocean. *J. Comp. Phys.*, **4**, 347-376.

Campin, J.-M., A. Adcroft, C. Hill, J. Marshall, 2004: Conservation of properties in a free-surface model. *Ocean Modeling*, **6** , 221-244. doi:10.1016/S1463-5003(03)00009-X.

Canuto, C., M. Y. Hussaini, A. Quarteroni and T. A. Zang, 1988: *Spectral Methods in in Fluid Mechanics*. Springer-Verlag, 567 pp.

Casulli, V. and R. T. Cheng, 1992: Semi-implicit finite-difference methods for 3-dimensional shallow-water flow. *Int. J. Numer. Meth. Fluids.*, **15**, 629-648.

Casulli, V. and E. Cattani, 1994: Stability, accuracy and efficiency of a semi-implicit method for 3-dimensional shallow-water flow. *Comput. & Math. Al.*, **27**, 99-112.

Casulli, V. and G. Stelling, 1998: Numerical simulation of 3D quasi-hydrostatic free-surface flows. *J. Hydraulic Engineer.*, **124**, 678-686.

Casulli, V., 1999: A semi-implicit finite-difference method for non-hydrostatic free-surface flows. *Int. J. Numer.*

Meth. Fluids., **30**, 425-440.

Cheng, R. T. and V. Casulli, 2001: Evaluation of the UnTRIM model for 3-D tidal circulation. *Proceedings of the 7-th Intern. Conf. on Estuarine and Coastal Modeling*, St. Petersburg, FL, Nov. 2001, 682-642.

Chu, P. C., and C.-W. Fan, 1997: Sixth-order difference scheme for sigma-coordinate ocean models. *J. Phys. Ocean.*, **27**, 2064-2071.

Chu, P. C., and C.-W. Fan, 2003: Hydrostatic correction for sigma coordinate ocean models. *J. Geophys. Res.*, **106**, (C6), 3206. doi:10.1029/2002JC001668.

Dewar, W. K., Y. Hsueh, T. J. McDougall, and D. I. Yuan, 1998: Calculation of pressure in ocean simulations. *J. Phys. Ocean.*, **28**, 577-588.

Dietrich, D. E., C. A. Lin, A. Mestas-Nunez and D.-S. Ko, 1997: A high resolution numerical study of Gulf of Mexico fronts and eddies. *Meteorol. Atmos. Phys.*, **64**, 187-201.

Dietrich, D. E., M. G. Marietta, and P. J. Roache, 1987: An ocean modeling system with turbulent boundary layers and topography, Part 1: numerical description, *Int. J. Numer. Methods Fluids*, **7**, 833-855.

Dukowitz, J. K., and R. D. Smith, 1994: Implicit free-surface method for the Bryan-Cox-Semtner ocean model. *J. Geophys. Res.*, **99**, 7991-8014.

Dukowicz, J. K., 2001: Reduction of density and pressure gradient errors in ocean simulations. *J. Phys. Ocean.*, **31**, 1915-1921.

Dukowicz, J. K., 2006: Structure of the barotropic mode in layered ocean models. *Ocean Modelling*, **11**, 49-68. doi:10.1016/j.ocemod.2004.11.005.

Durran, D. R., 1991: The third order Adams-Bashforth method: An attractive alternative to leapfrog time differencing. *Monthly Weather Review*, **119**, 702-720.

Durran, D. R., 1998: *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer-Verlag, 1998. 465 pp.

Gent, P. R., and J. C. McWilliams, 1990: Isopycnal mixing in ocean circulation models, *J. Phys. Ocean.*, **20**, 150-155.

Goldberg, M. & E. Tadmor 1988: Simple stability criteria for difference approximations of hyperbolic initial-boundary value problems. *Nonlinear Hyperbolic Equations - Theory, Computation Methods, and Applications*, Proceedings of the Second International Conference on Nonlinear Hyperbolic Problems, Notes on Numerical Fluid Mechanics, Vol. 24 (J. Ballmann and R. Jeltsch eds.), Vieweg Verlag, 179-185.

Greatbatch, R.J., Y. Lu, and Y. Cai, 2001: Relaxing the Boussinesq approximation in ocean circulation models. *J. Atmos. Oceanic Technology*, **18**, 1911-1923.

Greatbatch, R.J., and T. J. McDougall, 2003: The non-Boussinesq temporal residual mean. *J. Phys. Ocean.*, **33**, 1231-1239.

Griffies, S. M., A. Gnanadesikan, R. C. Pacanowski, V. D. Larichev, J. K. Dukowicz, R. D. Smith, 1998: Isoneutral diffusion in a z-coordinate ocean model. *J. Phys. Ocean.*, **28**, 805-830.

Griffies, S. M., C. Böning, F. O. Bryan, E. P. Chassignet, R. Gerdes, H. Hasumi, A. Hirst, A.-M. Treguier, D. Webb, 2000: Development in ocean climate modeling. *Ocean Modelling*, **2**, 123-192.

Griffies, S. M., R. C. Pacanowski, M. Schmidt, and V. Balaji, 2001: Tracer conservation with an explicit free surface method for z-coordinate ocean models. *Monthly Weather Review*, **5**, 1081-1098.

Haidvogel, D. B., H. Arango, K. Hedstrom, A. Beckmann, P. Rizzoli, and A. F. Shchepetkin, 2000: Model evaluation experiments in the North Atlantic Basin: Simulations in non-linear terrain-following coordinates. *Dyn. Atmos. and Oceans*, **32**, 239-281.

Hallberg, R., 1997: Stable split time stepping schemes for large-scale ocean modeling. *J. Comp. Phys.*, **135**, 54-65.

Hallberg, R., 2004: A thermobaric instability of Lagrangian vertical coordinate ocean models.*Ocean Modelling*, **8**, 279-300. doi:10.1016/j.ocemod.2004.01.001.

Haney, R. L., 1991: On the pressure gradient force over steep topography in sigma coordinate ocean models. *J. Phys. Ocean.*, **21**, 610-618.

Higdon, R. L., and A. F. Bennett, 1996: Stability analysis of operator splitting for large-scale ocean modelling. *J. Comp. Phys.*, **123**, 311-329.

Higdon, R. L., and R. A. de Szoeke, 1997: Barotropic-baroclinic time splitting for ocean circullation modeling. *J.*

Comp. Phys., **135**, 31-53.

Higdon, R. L., 1999: Implementation of barotropic-baroclinic time splitting for isopycnic-coordinate ocean modeling. *J. Comp. Phys.*, **148**, 579-604.

Higdon, R. L., 2002: A two-level time-stepping method for layered ocean circulation models. *J. Comp. Phys.*, **177**, 59-94. doi:10.1006/jcph.2002.7003

Higdon, R. L., 2005: A two-level time-stepping method for layered ocean circulation models: further development and testing. *J. Comp. Phys.*, **206**, 463-504. doi:10.1016/j.jcp.2004.12.011

Holland, W. R., 1973: Baroclinic and topographic influences on the transport in western boundary currents. *Geophys. Fluid Dyn.*, **4**, 187-210.

Jackett, D. R., and T. J. McDougall, 1995: Minimal adjustment of hydrostatic profiles to achieve static stability. *J. Atmos. Oceanic Technology*, **12**, 381-389.

Jackett, D. R., T. J. McDougall, R. Feistel, D. G. Wright, and S. M. Griffies, 2006: Algorithms for density, potential temperature, conservative temperature and freezing temperature of seawater. Submitted to *J. Atmos. Oceanic Technol.*

Kanarska, Y., Shchepetkin, A. F., and J. C. McWilliams, 2007: Algorithm for non-hydrostatic dynamics in ROMS. *Ocean Modeling*, **18** , 143-174. doi:10.1016/j.ocemod.2007.04.01.

Killworth, P. D., D. Stainforth, D. J. Webb and S. M. Paterson, 1991: The development of a free-surface Bryan-Cox-Semtner ocean model. *J. Phys. Ocean.*, **21**, 1333-1348.

Kliem, N., and J. D. Pietrzak, 1999: On the pressure gradient errors in sigma coordinate ocean models: A comparison with laboratory experiments. *J. Geophys. Res.*, **104**, 29,781-29,799.

Leonard, B. P., A. P. Lock, and M. K. McVean, 1996: Conservative explicit unrestricted-time-step constancy preserving advection schemes. *Monthly Weather Review*, **124**, 2588-2606.

Lilly, D. K., 1965: On the computational stability of time-dependent non-linear geophysical fluid dynamics problem. *Monthly Weather Review*, **93**, 11-26.

Lin, S. J., 1997: A finite volume integration method for computing pressure gradient force in general vertical coordinates. *Q. J. R. Meteorol. Soc.*, **123**, 1749-1762.

Liu, H & E. Tadmor, 2003: Critical thresholds and conditional stability for Euler equations and related models. In *"Hyperbolic Problems: Theory, Numerics, Applications", Proceedings of the 9th International Conference on Hyperbolic Problems*, Pasadena, Mar. 2002 (T. Hou and E. Tadmor, eds.), Springer-Verlag, 227-240.

Losch, M., A. Adcroft, and J.-M. Campin, 2004: How sensitive are coarse general circulation models to fundamental approximations in the equations of motion? *J. Phys. Ocean.*, **34**, 306-319. doi:10.1175/1520-0485(2004)034<0306:HSACGC>2.0.CO;2

Lu, Y., 2000: Including non-Boussinesq effect in Boussinesq ocean circulation models. *J. Phys. Ocean.*, **31**, 1616-1622.

Marchesiello, P., J.C. McWilliams, and A. Shchepetkin, 2003: Equilibrium structure and dynamics of the California Current System. *J. Phys. Ocean.*, **33**, 753-783.

Marsaleix, P., F. Auclair, M.-J. Herrmann, C. Estournel, I. Pairaud, and C. Ulses, 2008: Energy conservation in sigma-coordinate free-surface ocean models, *Ocean Modeling*, **20** , 61-89, doi:10.101/j.ocemod.2007.07.005

Marshall, J., A. Adcroft, C. Hill, L. Perelman, and C. Heisey, 1997: A finite-volume, incompressible Navier Stokes model for studies of the ocean on parallel computers. *J. Geophys. Res.*, **102**, 5753-5766.

McDougall, T. J., and C. J. R. Garett, 1992: Scalar conservation equations in a turbulent ocean. *Deep-Sea Res.*, **39**, 1953-1966.

McDougall, T. J., R. J. Greatbatch, and Y. Lu., 2002: On conservation equations in oceanography: How accurate are Boussinesq ocean models? *J. Phys. Ocean.*, **32**, 1574-1584. doi:10.1175/1520-0485(2002)032<1574:OCEIOH>2.0.CO;2

McDougall, T. J., D. J. Jackett, D. G. Wright, and R. Feistel, 2003: Accurate and computationally efficient algorithms for potential temperature and density and of seawater. *J. Atmos. Oceanic Technology*, **20**, 730-741.

McWilliams, J.C., 1996: Modeling the oceanic general circulation. *Annual Rev. of Fluid Mech.*, **28**, 1-34.

Mellor, G. L.,1991: An Equation of State for numerical modeling of oceans and estuaries. *J. Atmos. Oceanic Tech-*

nology, **1991**, 609-611.

Mellor, G. L., T. Ezer and L.-Y. Oey, 1994: The pressure gradient conundrum of sigma coordinate ocean models. *J. Atmos. Oceanic Technology*, **11**, 1126-1134.

Mellor, G. L., L.-Y. Oey, and T. Ezer, 1998: Sigma coordinate pressure gradient errors and the seamount problem. *J. Atmos. Oceanic Technology*, **15**, 1122-1131.

Mesinger, F., 1982: On the convergence and error problems of the calculation of the pressure gradient force in sigma coordinate ocean models. *Geophys. Astrophys. Fluid Dyn.*, **19**, 105-117.

Mesinger, F. and Z. I. Janjic, 1985: Problems and numerical methods of the incorporation of mountains in atmospheric models. *Lect. Appl. Math.*, **22**, 81-120.

Nadiga, B. T., M. W. Hecht, L. G. Margolin, and P. K. Smolarkiewicz, 1997: On simulating flows with multiple time scales using a method of averages. *Theor. and Comput. Fluid Dyn.*, **9**, 281-292.

Oppenheim, A.V., and R.W. Schafer, 1980: Discrete-Time Signal Processing, *Prentice-Hall*, pp. 447-448.

Orszag, S. A., 1971: Numerical simulation of incompressible flows with simple boundaries: Accuracy. *J. Fluid Mech.*, **49**, 76-112

Robinson, R., L. Padman, and M. D. Levine, 2001: A correction to the baroclinic pressure gradient term in the Princeton ocean model. *J. Atmos. Oceanic Technology*, **18**, 1068-1075.

Roe, P. L., 1981: Approximate Riemann solvers, parameter vectors and difference schemes. *J. Comp. Phys.*, **43**, 357-372.

Rueda, F. J., E. Sanmiguel-Rojas, and B. R. Hodges, 2007: Baroclinic stability for a family of two-level, semi-implicit numerical methods for the 3D shallow water equations *Int. J. Numer. Meth. Fluids*, **54**, 237-268, doi:10.1002/fld.1391

Shchepetkin, A. F. and J. C. McWilliams, 1998: Quasi-monotone advection schemes based on explicit locally adaptive dissipation, *Monthly Weather Review*, **126**, 1541-1580.

Shchepetkin, A. F. and J. C. McWilliams, 2003: A method for computing horizontal pressure-gradient force in an oceanic model with a non-aligned vertical coordinate. *J. Geophys. Res.*, **108**(C3), 3090. doi:10.1029/2001JC001047 35.1-35.34

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 Modeling*, **9** , 347-404. doi:10.1016/j.ocemod.2004.08.002.

Shulman, I., J. K. Lewis, and J. G. Mayer, 1999: Local data assimilation in the estimation of barotropic and baroclinic open boundary conditions. *J. Geophys. Res.*, **104**, C6, pp. 13,667-13,680.

Slordal, L. H., 1997: The pressure gradient force in sigma coordinate ocean models. *Int. J. of Num. Meth. in Fluids*, **24** , 987-1017.

Skamarock, W. C., and J. B. Klemp, 1992, The stability of time-split numerical methods for the hydrostatic and the nonhydrostatic elastic equations. *Monthly Weather Review*, **120**, 2109-2197.

Song, Y. T. 1998: A general pressure gradient formulation for ocean models. Part I: scheme design and diagnostic analysis. *Monthly Weather Review*, **126**, 3213-3230.

Song, Y. T. and D. G. Wright, 1998: A general pressure gradient formulation for ocean models. Part II: energy, momentum and bottom torque consistency. *Monthly Weather Review*, **126**, 3213-3230.

Stelling, G. S., and S. P. A. Duinmeijer, 2003: A staggered conservative scheme for every Froude number in rapidly varied shallow water flows. *Int. J. of Num. Meth. in Fluids*, **43** , 1329-1354. doi:10.1002/fld.537

Stelling, G. S., and J. A. Th. van Kester, 1994: On the approximation of horizontal gradients in sigma coordinates for bathymetry with steep bottom slopes. *Int. J. of Num. Meth. in Fluids*, **18** , 915-935.

de Szoeke, R. A., and R. M. Samelson, 2002: The duality between the Boussinesq and non-Boussinesq hydrostatic equations of motion. *J. Phys. Ocean.*, **32**, 2194-2203. doi:10.1175/1520-0485(2002)032<2194:TDBTBA>2.0.CO;2

Willebrand, J., B. Barnier, C. Böning, C. Dieterich, P. D. Killworth, C. LeProvost, Y. Jia, J. M. Molines, and A. L. New, 2001: Circulation characteristics in three eddy-permitting models of the North Atlantic. *Progress in Oceanography*, **48**, 123-161.