Correction and Commentary for "Ocean Forecasting in Terrain-Following Coordinates: Formulation and Skill Assessment of the Regional Ocean Modeling System" by Haidvogel *et al.*, J. Comp. Phys. 227, pp. 3595-3624

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

Received: 22 May 2008; *received in revised form*: 19 June 2009; *accepted*: 2 September 2009;

Keywords: Regional ocean modeling; Split-explicit time stepping; Terrain-following coordinates; Conservation and constancy preservation;

Published in: *Journal of Computational Physics*, **Vol. 228**(2009), pp. 8985-9000,

doi:10.1016/j.jcp.2009.09.002

Abstract

Although our names appear as co-authors in the above article (Ref. [1] hereafter H2008), we were not aware of its existence until after it was published. In reading the article, we discovered that a significant portion of it ($\sim 40\%$, or 10 pages) repeats three large fragments from our own previously published work, Shchepetkin and McWilliams (2000) [2] (hereafter SM2005), but now presented in such a way that the motivation for the specific algorithmic choices made in ROMS and the relations among the different model components are no longer clear. The model equations appearing in H2008, Sec. 2.1 (taken from an earlier article Haidvogel *et al.*(2000) [3]) are not entirely consistent with the actual equations solved in the ROMS code, resulting in contradictions within H2008 itself. In our view the description in H2008 does not constitute a mathematically accurate statement about the hydrodynamic core of ROMS. The purpose of this note is to clarify and correct this, as well as to explain some of the algorithmic differences among ROMS versions now in use.

1 Overview

As a review article, H2008 is expected to provide readers with a balanced overview of the current status of development, capabilities, and applications of the ROMS model, which at this time has passed its tenth anniversary as a collaborative project. Naturally, with this goal in mind, we expected this article to reuse previously published material with a reasonable degree of summary and condensation. However, we discovered that H2008 cuts and pastes three large chunks from SM2005 and reuses the model-formulation section from Haidvogel *et al.*(2000) [3] in a way that impairs the original relations

between the parts. There are also omissions of material that is critical for understanding the ROMS code kernel design, and even several reversals of the original meaning of SM2005. In retrospect, we characterize the latter as a mathematical optimization study using a somewhat engineering approach where the attention to how different algorithmic parts interact among each other takes precedence over the novelty of individual components. This changes the burden of proof from demonstrating the functionality of the model to showing that it cannot be made more efficient while staying within the class of split-explicit, free-surface models. Accordingly, SM2005 surveys a wide range of possible time-stepping algorithms, even thought only a few are actually used in the code. It also indicates the possibility of multiple variants of the code where competitive algorithms are identified (*cf.,* Section 5 below). The purpose of this commentary is to diagnose and partially remedy the possible confusions inadvertently caused by publication of H2008, and to add some information that may help readers understand relations among several variant algorithmic approaches within the ROMS family of codes. Much of our discussion is quite technical as are the parts we refer to in H2008. In the absence of alternative community conventions for communicating code design specifications, it is important that the published algorithms be accurately expressed. A recent, more complete discussion of the design criteria for an oceanic model such as ROMS can be found in Shchepetkin and McWilliams (2008) [4]. Having said that, we have no doubt about the various illustrations of successful ROMS solutions and forecasts that comprise the second half of H2008, starting with Section 4.

1.1 Major Critical Comments

• Section 2.1, *Hydrodynamic Core*, H2008 presents the model equations in a shortened and simplified version (*i.e.,* omitting horizontal curvilinear coordinates) of Section 2.1 in [3] (hereafter cited as DAMEE2000) [1] . DAMEE2000, in its turn, is drawn from [5] (hereafter SH94). The way the continuity equation is written — Eq. (4) in H2008 (same as Eq. (6) in DAMEE2000) *vs.* Eq. (2.26) in SH94 — indicates an unawareness that the way the vertical coordinate is perturbed by the free-surface elevation, ζ, in ROMS is different from that in SH94 since

$$\frac{\partial}{\partial t}\left(\frac{\zeta}{mn}\right) \neq \frac{\partial}{\partial t}\left(\frac{H_z}{mn}\right) \tag{1.1}$$

for ROMS vertical coordinate [2] . This is explained in detail in Section 2 below. The change of vertical coordinate also leads to a change in the definition of vertical velocity in transformed coordinates, which in turn affects constancy-preservation for tracers. This is further discussed in Sections 3 and 4 below.

• In SM2005, Section 2 surveys a wide range of time-stepping algorithms, Section 4 discusses several possible choices for building an actual code, but Section 5 describes a specific combination of algorithms for a particular version of the code. However, because Section 3.6 of H2008 just follows Section 5 of SM2005 with no discussion of alternatives, a reader easily gets the impression that the ROMS kernel is just the algorithm formulated there as *Stage 1* to *6*. This is not accurate. In fact, there are four well-identified variants of hydrodynamic time-stepping kernel for the ROMS-family codes currently in use (Section 5 below). Furthermore, the specific ROMS kernel described in Sec. 3.6 in H2008 was not used in the examples reported in the following Section 4 in H2008, rather an alternative variant was used instead. Section 5 below clarifies this situation.

[1] Data Assimilation and Model Evaluation Experiments

[2] All symbols appearing here have the same meaning as in the articles cited.

• Section 3.1 in H2008 repeats the beginning of Section 1.1 in SM2005, while omitting a 3/4-page discussion there that ends with "Throughout this study we assume that our vertical system of coordinates is no longer separable...". In contrast, Section 3.1 in H2008 ends with Eq. (9) [same as Eq. (1.5) in SM2005], which is $z'(x, y, \sigma) = S(\sigma) \cdot h(x, y)$ and is a special-case example of a separable transformation. After Eq. (9), H2008, there is only a single sentence referring to the S-coordinate in SH94. This sentence is also taken from SM2005. However, cutting off the discussion here inadvertently leaves an impression that the ROMS vertical coordinate is either Eq. (9) in H2008 or is the same as in SH94. This is not consistent with the original meaning of SM2005 which deliberately leaves the design of vertical coordinate open, while facilitating implementation of alternative coordinates, nor does it reflect the actual current state of ROMS where newer, more advanced options became available (Section 2 below).

• Section 3.2, the last sentence on p. 3599 in H2008 states, "Except where noted otherwise (*e.g.,* Section 3.7), a centered second-order finite-difference approximation is adopted in the horizontal". This is misleading. Section 3.7 in H2008 describes only two algorithms: a QUICK-type, third-order upstream-biased interpolation used to compute advective fluxes and a conservative parabolic spline reconstruction optionally used for vertical advection and viscosity and diffusion. Besides these there are many other algorithms employed in the code that go beyond second-order differencing, *e.g.,* the pressure-gradient force (PGF) computed using monotonized cubic interpolations in all directions [6]; a similar operator for advection of tracers; a positive-definite MPDATA advection scheme [7] for sea-ice concentration and thickness, biological tracers, and sediment concentration; a monotonized, high-order vertical advection scheme for vertical migration of biological species and sinking of sediments, *etc.* Most importantly, the whole spirit of spatial discretization in a hydrodynamic code like ROMS is finite-volume rather than finite-difference.

• Section 3.2 in H2008 repeats Section 1.2 in SM2005, but now with a different title, *Spatial Discretization*. With the exception of the first two sentences, the only topic covered in this section is how the vertical coordinate is perturbed when ζ deviates from its resting state, which is reflected in the original title, *Perturbed Vertical Coordinate System*. In SM2005 the sole purpose of Section 1.2 is to introduce the proportional perturbation of grid-box heights [Eq. (1.10) there] as the framework for constructing a conservative and constancy-preserving mode splitting procedure which also incorporates fast-time-averaging for barotropic fields: at first, to expose the problem in the following Section 1.3, and subsequently in Sections 3 and 4 describe in detail an algorithm having all the needed properties. H2008 does not makes this emphasis, and in fact, as we see later, misses entirely the aspect of compatibility between the fast-time-averaging procedure for the barotropic mode and the slow-time discrete continuity equation, which is the basis for constancy-preserving time stepping algorithm for tracers.

• Section 3.3 in H2008 repeats Section 1.3 in SM2005 that originally had the title *Conflict between integral and constancy-preservation for tracers*. The word "conflict" has now been removed from the title, and the discussion at the end is shortened by removing the sentence, "Perhaps the most delicate matter here is replacement of ζ at $n+1$ with its fast-time-averaged value: not doing so leaves room for aliasing error, while the replacement makes the "slow-time" discrete 2D continuity equation ... hold only within the order of temporal accuracy, but no longer exactly even though it is exact at every time step)" (SM2005). Both changes impair the meaning of the section. In fact, the entire purpose of Section 1.3 in SM2005 is to emphasize that in an oceanic model with a changing vertical coordinate (due to changing ζ) and with different time-stepping algorithms for barotropic and baroclinic sub-systems, having exact finite-volume, finite-time-step consistency of the 3D, slow-time discrete continuity equation is not automatic; a special procedure must be designed to guarantee it. This applies to both split-explicit and implicit models. (If SM2005 were written today, three more references would be cited

here [8, 9, 10].) Section 1.3 in H2008 merely states that there is a conflict but provides no solution. The conflict is resolved in Section 3.2 in SM2005 by constructing a 2-way fast-time-averaging procedure that computes five fields – $\langle\zeta\rangle^{n+1}$, $\langle\overline{U}\rangle^{n+1}$, $\langle\overline{V}\rangle^{n+1}$, along with $\left\langle\left\langle\overline{U}\right\rangle\right\rangle^{n+1/2}$ and $\left\langle\left\langle\overline{V}\right\rangle\right\rangle^{n+1/2}$ – that satisfy Eq. (3.39), the slow-time, vertically integrated continuity equation for fast-time-averaged values. Then before performing the corrector step for tracers, the 3D fluxes are forced to have exactly the same vertical integrals as $\left\langle\left\langle\overline{U}\right\rangle\right\rangle^{n+1/2}$ and $\left\langle\left\langle\overline{V}\right\rangle\right\rangle^{n+1/2}$. H2008 provides no explanation and makes little mention of the fast-time-averaging procedure in ROMS; *e.g.,* there is no evidence of even the existence of secondary-weighted fast-time-averaged barotropic fluxes (denoted by $\langle\langle.\rangle\rangle$), with the exception of the unexplained symbol $\left\langle\left\langle\overline{\mathbf{U}}\right\rangle\right\rangle^{n+1/2}$ in *Stage 5* on p. 3608. The ending sentences of Section 3.1 in H2008 are: "Alternatively, one might distribute the mismatch of discrepancy in (25) throughout the water column so that the top boundary condition holds but at the expense of discrepancy in (21) [SH94]. In either case, a conservative update of the tracer fields (16) loses its constancy-preservation property." This unavoidably gives the reader the impression that ROMS is not different from SH94 with respect to this matter, which is false.

• p. 3607, *Stage 1* and *Stage 2*: The original text in SM2005 for these stages mentions "the use of artificial compressibility equation (*i.e.,* the pseudo-compressible algorithm)", but these words are removed in H2008. This leads to a loss of meaning because a finite-volume time step requires new-time-step grid-box heights in order to compute a control volume. This cannot be done at this stage because ζ is not yet updated. Nevertheless, H2008 states, "this step is constancy-preserving, but not conservative", and then explains why it is acceptable at this stage. In Section 4 below, we provide an overview of conservative and constancy-preserving time-stepping algorithms for tracers, focusing on the role of the continuity equation in the presence of $\zeta \neq 0$.

• Eq. (65) on p. 3609 and Table 1 on p. 3611 in H2008: This definition of model skill is also known as the index of agreement [11, 12]. Willmott argues that more standard measures, such as the correlation coefficient between model and data, may be insufficient and often misleading. Eqs. (1)-(4)) in **(author?)** [12] distinguish four kinds of errors: mean bias, quadratic variance (related to root-mean-square error) with and without subtracting mean bias, and mean absolute error. The proposed index of agreement is a single composite measure sensitive to all these kinds of errors, and it is intended for cross-comparison among several different models. In simple cases where all errors are presumed to be of a single type, the criterion can be applied to rank the overall performance. However, in application to oceanic modeling it is an oversimplification because the index cannot distinguish between the different types of errors, nor pinpoint a specific cause for error to be addressed in a model redesign. Therefore, it is difficult to judge the results in H2008, Table 1, solely from the fact that the skill defined by Eq. (65) is close to 1. In this respect the approach used by H2008 is rather different from the common model assessment practices within the ROMS community [*cf.,* 3, 13, 14, and the remainder of Section 4 in H2008 itself]) as well as from the related effort in data assimilation [15, 16]. The latter tends to use cost functions with subjectively specified weights; *i.e.,* some errors are given more tolerance than others. This differentiation is necessary to address uncertainties associated with differing quality of observational data and with the fundamental predictability limit to how well model and data can match each other in a deterministic sense when the flows are dynamically unstable and turbulent.

Although not always expressed as quantitative measures, there is a common oceanic model assessment procedure. At first, one evaluates the model ability to reproduce known results in controlled (often idealized) simulations. Secondly, for a realistic configuration the key issue is that most measurements are sparsely distributed and cannot be meaningfully compared with model results directly. Thus, one bests gauges model results against observations of the mean state [*e.g.,* 17], variance, or other sta-

tistical measure, which requires averaging of the model results in various ways (climatological mean, high- and low-path filtering, extracting signals with selected frequencies, *etc.*) before comparisons can be made. Examples of meaningful measures are the path of the Gulf Stream, the seasonal cycle of the depth of particular isotherms, and eddy kinetic energy. Most often these comparisons are not easily quantified into a single number for "skill". In dealing with turbulent flows, there is little hope to achieve a close phase correspondence between the model and the measurement (periodically-forced barotropic tides are a counter-example). From the point of view of Eq. (65), a phase differences is counted as error, as is the ability of the model to produce fine-scale features (*e.g.,* fronts and eddies) not present in observational climatologies due to under-sampling. This may lead to a paradoxical result that Eq. (65) may yield a higher skill for a coarse-resolution model than for a fine one, because it interprets fine-scale phenomena as noise. On the other hand, the time-and-space averaged effect of the latter may lead to a significant difference in the mean circulation, which is observable in sparse data but cannot be accurately simulated with a coarse model. This leads to a greater skill of a fine-resolution model, if its output is averaged appropriately to match the smoothness of the available data.

• H2008 makes a rather skewed citation of published literature with several major omissions. For example, a significant milestone in realistic simulations of US West Coast and California Current System using ROMS, Marchesiello *et al.*(2003) [13], is not cited. Neither is Centurioni *et al.*(2008) [18], which compares drifter data from the same area against the results from four different ocean models including ROMS. Nor are Penven *et al.*(2006) [19], which is a demonstration of code working on a set of progressively refined grids; and Capet *et al.*(2008) [20], which is the first in series of high-resolution ($\Delta x < 1\,km$) studies dedicated to transition toward submesoscale dynamics – all of which were available before publication of H2008. Also not cited is a line of late 199x development with significant influences on ROMS codes [21, 22, 23, 24], which undertake a major redesign of SPEM code with subsequent use for realistic simulations (Section 5).

1.2 Minor Comments and Typos

• p. 3595, Abstract, the one before the last line appearing on this page "...quasi-monotone advection produces both more robust and accurate...": this is misleading because "quasi-monotone" advection schemes [25] use a special numerical mechanism to suppress dispersive overshoots of high-order, centered advection. No such thing is actually used in ROMS. What needs to be said instead is that, unlike virtually all other ocean models, ROMS uses higher-than second order advection for both tracers and momentum.

• p. 3603: The subtitle *3.5. Improved Mode-Splitting* is misplaced, because Eq. (29) above it already belongs to the improved splitting algorithm.

• p. 3604, Eq. (36): The l.h.s. term should be $\dfrac{\partial}{\partial t}(D\overline{u}) + ...$, *i.e.,* with lowercase $\overline{u}$ instead of $\overline{U}$. This typo is traced back to the typo in the original Eq. (3.15) in SM2005.

• p. 3605, Eq. (37): The second r.h.s. term should be $\dfrac{\rho_{k+1/2} - \rho_{k-1/2}}{H_k} z'$, *i.e.,* with minus sign instead of a plus sign. This typo is traced back to the original Eq. (3.16) in SM2005.

• p. 3607, *Stage 1*: The symbol $\langle \overline{\mathbf{U}} \rangle^n$ at the end is undefined.

• p. 3607, *Stage 3*: The symbol $\langle \overline{\mathbf{U}} \rangle^{n+1}$ is undefined.

• p. 3607, *Stage 4*: The symbol $\langle \zeta \rangle^{n+1}$ at the end is undefined; also $H_{i,j,k}^{n+1}$ should not be bold-face.

• p. 3608, *Stage 5* and *Stage 6*: The symbols $\left\langle\left\langle \overline{\mathbf{U}} \right\rangle\right\rangle^{n+1/2}$ and $\langle \overline{\mathbf{U}} \rangle^{n+1}$ are undefined. Moreover these two symbols appear in wrong places, and should be switched: $\langle \overline{\mathbf{U}} \rangle^{n+1}$ should appear in *Stage 5*, which

finalizes the computation of horizontal velocity components at time step $n+1$. Conversely, $\left\langle\!\left\langle \overline{\mathbf{U}} \right\rangle\!\right\rangle^{n+1/2}$ should appear in *Stage 6* where it is used to construct a set of finite-volume fluxes at $n+1/2$ in such a way that the discretized slow-time continuity equation (*cf.,* Eq. (4.8 in Section 4 below) holds exactly. This misplacement of the two symbols occurs only in H2008, while the original version in SM2005 is correct.

• p. 3609, just before Eq. (61), "The spline is represented by Eq. (43)..": This should instead be Eq. (37); Eq. (43) has nothing to do with the spline.

• p. 3609, Eq. (61): This is notationally inconsistent with Eq. (37) for two reasons. First, the symbol z, which in this context has the meaning of a local vertical coordinate defined within a single vertical grid box, should be replaced with z', $(-\Delta z/2 \le z' \le +\Delta z/2)$ to avoid confusion with the vertical z-coordinate as it is used elsewhere $(-h \le z \le \zeta)$. Second, ζ in the l.h.s. should be replaced with z' as well, and, possibly, $\Delta z_k \to H_k$ to be more consistent with Eq. (37).

• Fig. 8 on p. 3614, and the associated discussion on p. 3613: it should be explained that the date of July 2003 for the SeaWiFS chlorophyll data was chosen because of data availability, and it is used here just as a representative image. Otherwise comparing two fields which are 9 years apart in time does not seem to be very meaningful.

• p. 3623, Ref. [38] in H2008: author name Willmott is misspelled; should be with double "ll".

• p. 3623, Ref. [42] in H2008: one name is missing from the list of authors, A. Kaplan should be added at the end after T. M. Powell.

2 Vertical Coordinate of ROMS

2.1 Design Considerations for Terrain-following Coordinates

For an extended period of time the ROMS *de facto* default vertical coordinate was functionally similar to Eq. (2.16) in SH94 (except for how it is perturbed by the changing ζ, which was different from the start),

$$z^{(0)}(x,y,s) = h_c \cdot s + (h - h_c) \cdot C(s)\,, \qquad -1 \le s \le 0\,, \tag{2.1}$$

however, the code is written in such a way that one transformation can be easily replaced with another with all necessary changes occurring strictly within the routines that set the vertical coordinate. In practice this is implemented by storing the perturbed coordinate $z = z(x,y,s)$ in an three-dimensional array, while the algorithm creating it is hidden from rest of the code and at no point does the code rely on the specific properties of the transformation (*e.g.,* unlike POM [3], where virtually all components of the code are written with the assumption that the transformation is separable, $z^{(0)}(x,y,\sigma) = C(\sigma) \cdot h(x,y)$). ROMS-related literature often uses the symbol s instead of σ for independent sigma-coordinate variable, so the symbols s and σ are interchangeable in most situations. The subtle difference between the two is that σ is normally reserved for the proportional (non-stretched) sigma-coordinate, $\sigma = (z-\zeta)/(h+\zeta)$, which uniquely defines the remapping due to both the bottom topography and the perturbation in free surface, while s appears in the context of a more general terrain-following coordinate. In ROMS the $z \leftrightarrow s$ remapping is done first assuming the unperturbed state of free surface, then the resultant vertical coordinate system is perturbed in a proportional man-

[3] Princeton Ocean Model [26].

ner to accommodate non-resting free surface (this Sec. below). $h = h(x, y)$ is bottom depth for a resting sea surface. $C(s)$ is a monotone nonlinear function constrained by $-1 \le C(s) \le 0$ along with $C(-1) = -1$ and $C(0) = 0$. C introduces a stretching of the vertical grid, typically $\mathrm{d}C(s)/\mathrm{d}s \ll 1$, $s \to 0$ resulting in refinement near the surface, while, conversely $\mathrm{d}C(s)/\mathrm{d}s \gg 1$ in the abyss resulting in coarsening. In practice it is common to have a minimum vertical grid spacing of $\Delta z \sim 5m$ near the surface while Δz may be as large as $\sim 500m$ in the deep abyss. $h_c = \mathrm{const}$ (also positive) is normally chosen to be comparable with the expected depth of the pycnocline. It introduces a set of nearly horizontal levels in the upper ocean, $-h_c < z < 0$, in deep regions where $h_c \ll h$. Besides local refinement near the surface, $h_c > 0$ produces set of vertical levels with nearly uniform vertical spacing by deferring vertical stretching until $z < -h_c$. This is needed for optimal resolution of the surface boundary layer, and it yields a less-than-proportional dependency of the upper level depths on the topography that overall mitigates sigma-errors in pressure-gradient and advection terms. At $z \sim -h_c$ the transform smoothly blends with the levels below that behave more and more like sigma levels following topography.

A major limitation of (2.1) is that it becomes a non-monotonic function of s if $h < h_c$ because a typical choice of stretching function $C(s)$ has refinement near the surface, $\mathrm{d}C(s)/\mathrm{d}s \ll 1$ at the expense of the existence of a range of s where $\mathrm{d}C(s)/\mathrm{d}s > 1$. So once $(h - h_c)$ becomes negative, $\partial z^{(0)}/\partial s$ may become negative as well. This imposes the limitation of $h_c \le h_{\min}$, which essentially negates the usefulness of having h_c in (2.1) by forcing an uncomfortable compromise between choosing h_c too shallow, or modifying/masking out model bottom topography in shallowest places. For several years already, the default choice in the UCLA ROMS is

$$z^{(0)}(x, y, s) = h \cdot \frac{h_c \cdot s + h \cdot C(s)}{h + h_c}\,, \tag{2.2}$$

instead of (2.1), thus removing the need for choosing $h_c \le h_{\min}$. Recently this was adopted into the Rutgers version (see Sec. 5 for explanation different branches of ROMS) as the preferred option there as well. The choice of stretching function $C(s)$ is application-dependent. For large-scale, open-ocean simulations a common selection is

$$C(s) = \left[1 - \cosh(\theta s)\right] / \left[\cosh(\theta) - 1\right]\,, \tag{2.3}$$

with parameter value $\theta = 5.5 \,...\, 6.5$ and $h_c = 120...300m$. In coastal configurations, where there is a need to resolve bottom boundary layer, a two-stage stretching

$$C(s) \equiv C[S(s)] \qquad \text{where} \qquad S(s) = \frac{1 - \cosh(\theta_s s)}{\cosh(\theta_s) - 1} \qquad \text{and} \qquad C(S) = \frac{\exp(\theta_b S) - 1}{1 - \exp(-\theta_b)}\,, \tag{2.4}$$

is more preferred. Parameters θ_s and θ_b control surface and bottom refinement. Note that the functional limit of $C(S)$ when $\theta_b \to 0$ is $C(S) = S$, meaning that (2.3) is just a special case of (2.4). For all settings of parameters $C(s)$ has the property $\mathrm{d}C(s)/\mathrm{d}s\Big|_{s\to 0} \to 0$, which, in combination with (2.2), makes $\partial z^{(0)}/\partial s\Big|_{s\to 0} \sim h \cdot h_c/(h + h_c) \sim h_c$ in the deep areas, $h_c \ll h$. As a consequence, the vertical size of the uppermost grid-boxes generated by (2.2)-(2.3) is nearly independent from the bottom topography (the uppermost grid thickness can be estimated as h_c/N, where N is the number of vertical levels). This makes the coordinate behave effectively like a z-coordinate near the surface, which is optimal for surface boundary layer simulation, and, with the natural increase of stratification in the upper part, this design helps mitigate sigma-coordinate pressure gradient errors. In shallow places (*e.g.,* over

continental shelves and coastal areas), (2.2) changes its behavior toward more a sigma-like coordinate, and it asymptotes toward a uniform-resolution sigma-grid when $h \ll h_c$ regardless of the choice of $C(s)$.

2.2 Perturbation of Vertical Coordinate by Free Surface

With the exception of the use of horizontal Cartesian coordinates instead of curvilinear ones, Eqs. (1)-(5) in Sec. 2.1 in H2008 are equivalent to those in Sec. 2.1 in DAMEE2000, which were derived from Eqs. (2.21)-(2.26) in SH94. Although it may seem to be very subtle and possibly insignificant, the modification of how the continuity equation is written — specifically Eq. (2.26) in SH94 *vs.* Eq. (6) in DAMEE2000 that also appears as Eq. (4) in H2008 — exposes a major misunderstanding about how the vertical coordinate of ROMS is perturbed by a moving free surface. H2008 gives an impression that the coordinate is still the same as in SH94, but it is not. Consequently, Eq. (4) in H2008 contradicts Eqs. (11)-(12) in H2008.

2.3 Perturbed Vertical Coordinate in SH94

The vertical coordinate transformation in SH94 is its Eq. (2.16),

$$z = \zeta(1+s) + h_c s + (h - h_c)C(s)\,, \qquad -1 \le s \le 0\,, \tag{2.5}$$

where $s = 0$ corresponds to the free surface $z = \zeta$, and $s = -1$ the bottom, $z = -h$. The other symbols, h_c and $C(s)$ are the same as in (2.1). The vertical derivative of (2.5),

$$H_z \equiv \frac{\partial z}{\partial s} = (\zeta + h_c) + (h - h_c) \cdot \frac{d}{ds}C(s)\,, \tag{2.6}$$

[*cf.,* Eq. (2.18) in SH94] appears as a metric function in all terms in Eqs. (1)-(5) in H2008. In the discrete code H_z sets the height of control volumes. Usually $dC(s)/ds \ll 1$ for $s \to 0$, implying a vertical grid refinement near the surface. Obviously, as a consequence of (2.6),

$$\frac{\partial H_z}{\partial t} = \frac{\partial \zeta}{\partial t}\,. \tag{2.7}$$

This indicates that Eq. (6) in DAMEE2000 is equivalent to Eq. (2.26) in SH94 only as long as ζ participates in the vertical coordinate transformation as in (2.5). In the discrete code (2.7) means that all vertical grid boxes $H_{i,j,k}$ receive exactly the same perturbation with ζ, regardless of their size,

$$H_{i,j,k} = H^{(0)}_{i,j,k} + \zeta/N\,, \tag{2.8}$$

where $H^{(0)}_{i,j,k}$ is unperturbed (corresponding to a $\zeta_{i,j} = 0$ grid-box height) and N is the number of vertical sigma-levels.

2.4 Perturbed Vertical Coordinate in ROMS

ROMS is not tied to a specific functional form for the vertical coordinate transformation; rather it assumes that there is an unperturbed (*i.e.,* corresponding to $\zeta \equiv 0$) mapping $z^{(0)} \leftrightarrow s$,

$$z^{(0)} = z^{(0)}(x, y, s)\,, \tag{2.9}$$

such that $z^{(0)} = 0$, if $s = 0$ and $z^{(0)} = -h$, $s = -1$. It is presumed to be differentiable in all directions but otherwise general. Once the mapping (2.9) is chosen, the perturbation due to $\zeta \neq 0$ is introduced as

$$z = z^{(0)} + \zeta \left(1 + z^{(0)}/h\right) , \tag{2.10}$$

[*cf.,* Eq. (1.9) in SM2005], so the counterpart of (2.6) becomes

$$H_z = \frac{\partial z^{(0)}}{\partial s} \cdot \left(1 + \frac{\zeta}{h}\right) = H_z^{(0)} \cdot \left(1 + \frac{\zeta}{h}\right) \tag{2.11}$$

[*cf.,* Eq. (1.10) in SM2005]. Then (2.7) becomes

$$\frac{\partial H_z}{\partial t} = \frac{H_z^{(0)}}{h} \cdot \frac{\partial \zeta}{\partial t} , \tag{2.12}$$

after which it is no longer possible to replace Eq. (2.26) in SH94 with Eq. (6) in DAMEE2000, which essentially invalidates the latter and Eq. (4) in H2008 as well [4] . Finally, the counterpart of (2.8) becomes

$$H_{i,j,k} = H^{(0)}_{i,j,k} + H^{(0)}_{i,j,k} \cdot \frac{\zeta}{h} . \tag{2.13}$$

This means that each grid-box receives a perturbation proportional to its unperturbed height [5] . This property was discussed at the end of Sec. 1.2 in SM2005, including how it differs from SH94.

3 ROMS Equations of Motion

The set of ROMS equations is comprised of horizontal momentum equations written in horizontal orthogonal curvilinear coordinates (ξ, η) (hence the length of an infinitesimal arc $\mathrm{d}l$ associated with increments in coordinates is $\mathrm{d}l^2 = \mathrm{d}\xi/m^2 + \mathrm{d}\eta^2/n^2$ where m^{-1} and n^{-1} are Lamé metric coefficients):

$$\frac{\mathscr{D}u}{\mathscr{D}t} - \hat{\mathcal{F}}v = -\frac{H_z}{n} \cdot \frac{1}{\rho_0} \cdot \left.\frac{\partial p}{\partial \xi}\right|_z + \mathcal{G}_u \qquad \frac{\mathscr{D}v}{\mathscr{D}t} + \hat{\mathcal{F}}u = -\frac{H_z}{m} \cdot \frac{1}{\rho_0} \cdot \left.\frac{\partial p}{\partial \eta}\right|_z + \mathcal{G}_v . \tag{3.1}$$

$\mathscr{D}/\mathscr{D}t$ is the material derivative in conservation form in curvilinear coordinates,

$$\frac{\mathscr{D}*}{\mathscr{D}t} = \frac{\partial}{\partial t}\left(\frac{H_z}{mn}*\right) + \frac{\partial}{\partial \xi}\left(\frac{H_z u}{n}*\right) + \frac{\partial}{\partial \eta}\left(\frac{H_z v}{m}*\right) + \frac{\partial}{\partial s}\left(\frac{\omega_s}{mn}*\right) . \tag{3.2}$$

[4] This comment also applies to Sec. 2.1 in DAMEE2000 because the code used to compute its results already had its vertical coordinate perturbation using (2.10) instead of (2.5). The correct version of the continuity equation (6) in DAMEE2000 should have its first term as $\dfrac{\partial}{\partial t}\left(\dfrac{H_z}{mn}\right)$, and the correct relationship between the z-coordinate and the transformed-coordinate vertical velocity (unnumbered equation on p. 244, DAMEE2000) should be

$$\Omega = \frac{1}{H_z}\left[w - \frac{z+h}{\zeta+h} \cdot \frac{\partial \zeta}{\partial t} - mu\frac{\partial z}{\partial \xi} - nu\frac{\partial z}{\partial \eta}\right]$$

using the DAMEE2000 notation. This remark does not put into question any of the computational results presented there.

[5] The way that the ROMS vertical coordinate is perturbed by $\zeta \neq 0$ is rather similar to how it occurs in POM. The separable coordinate transformation in POM is $z = \zeta + D \cdot C(s) = \zeta + (h + \zeta) \cdot C(s)$. Hence, in our terms, $z^{(0)} = h \cdot C(s)$, and $H_z = \partial z/\partial s = (h + \zeta) \cdot dC/ds = H_z^{(0)} \cdot (h + \zeta)/h$, which is the same as (2.11).

The horizontal derivatives $\partial/\partial\xi = \partial/\partial\xi|_s$ and $\partial/\partial\eta = \partial/\partial\eta|_s$ are along s-surfaces. $\hat{\mathcal{F}}$ is a generalized Coriolis parameter,

$$\hat{\mathcal{F}} = \frac{H_z}{mn}\left\{ f + mn\cdot\left(v\,\frac{\partial(1/n)}{\partial\xi} - u\,\frac{\partial(1/m)}{\partial\eta}\right)\right\}, \tag{3.3}$$

that combines the Coriolis force due to Earth rotation (with frequency $f = 2\Omega\,\sin\varphi$) with the fictitious inertial forces due to curvature of the horizontal coordinates. This approach follows [27] [Eqs. (257)-(263) there] and ensures that all these terms cancel identically when a kinetic-energy equation is derived from (3.1). The same also applies to their discretized versions.

Continuity and tracer equations are written as

$$\frac{\mathscr{D}q}{\mathscr{D}t} = \mathcal{G}_q\,, \qquad\qquad q\in\{\text{const},\,\Theta,\,S,\,...\}\,, \tag{3.4}$$

where $\Theta,\,S,...$ are potential temperature, salinity, and other tracers associated with possible submodels for sediment and biogeochemical concentrations. $\mathcal{G}_q$ in (3.1) and (3.4) denotes dissipation and forcing terms: viscosity and diffusion, wind stress and thermal forcing, heating by light absorption, biological conversions, vertical migration of biota, cohesive processes in suspended sediments, *etc.* The pressure and horizontal PGF are computed from hydrostatic balance,

$$p = g\int_s^0 \rho H_z\,\mathrm{d}s' = g\int_z^\zeta \rho\,\mathrm{d}z' \quad\text{hence}\quad -\left.\frac{\partial p}{\partial\xi}\right|_z = -g\rho\Big|_{s=0}\cdot\frac{\partial\zeta}{\partial\xi} - g\int_z^\zeta \left.\frac{\partial\rho}{\partial\xi}\right|_{z=\text{const}}\mathrm{d}z'\,, \tag{3.5}$$

which is then transformed back into the s-coordinate,

$$-\left.\frac{\partial p}{\partial\xi}\right|_z = -g\rho\Big|_{s=0}\cdot\frac{\partial\zeta}{\partial\xi} - g\int_s^0\left[\frac{\partial z}{\partial s}\cdot\left.\frac{\partial\rho}{\partial\xi}\right|_s - \frac{\partial\rho}{\partial s}\cdot\left.\frac{\partial z}{\partial\xi}\right|_s\right]\mathrm{d}s'\,. \tag{3.6}$$

The expression $[...]$ inside the rightmost integral (essentially a density Jacobian $\mathscr{J}_{\xi,s}(\rho,z)$) serves as a prototype for the method for accurate approximation of the baroclinic PGF term [6] that mitigates the infamous sigma-coordinate PGF error [6].

The system (3.1), (3.4), and (3.6) is closed by the Equation of State (EOS) $\rho = \rho_{\text{EOS}}(\Theta,S,P)$, which is is adopted from [28] to allow the use of potential rather than *in situ* temperature, and furthermore, following the approach of [29] it is "stiffened" by noting that $\rho_{\text{EOS}}(\Theta,S,P) = r(P)\cdot\rho^\bullet_{\text{EOS}}(\Theta,S,P)$, where $r(P)$ (a function of pressure only) absorbs most of the variation of density in the ocean. Stiffening by the replacements $\rho\to\rho^\bullet$ and $\rho_0\to\rho_0^\bullet$ eliminates up to $\sim 90\%$ of the errors associated with the Boussinesq approximation. To facilitate adiabatic differencing — critical to preventing the appearance of spurious negative stratification in the polynomial interpolation of density [6] — the EOS is split into two components, $\rho^\bullet_{\text{EOS}}(\Theta,S,P) = \rho_1(\Theta,S) + q_1^\bullet(\Theta,S)\cdot P\,(1+q_2P)$, where $q_2 = \text{const}$. Both ρ_1 and $q_1^\bullet$ are available for the PGF algorithm and for evaluating the vertical stratification used in parameterized vertical mixing processes. For simplicity and consistency with the Boussinesq approximation, the EOS pressure is replaced with $P = \rho_0 g(\zeta - z)$.

Besides the use of curvilinear horizontal coordinates and a few extra details, the set of equations in this section is similar to Eqs. (1)-(5) in H2008 but with two exceptions: the definition of vertical ve-

[6] Note that although the first term in the r.h.s. of (3.6) is predominantly of barotropic nature, while the second one is mainly baroclinic, Eq. (3.6) should not be viewed as the basis for barotropic-baroclinic mode-splitting since the second term has some dependency on ζ. The actual split used in ROMS is discussed in SM2005.

locity in transformed vertical coordinates (essentially the replacement $\Omega \leftrightarrow \omega_s$, where Ω is as in SH94 and DAMEE2000) and the formulation of the continuity equation (due to how the vertical coordinate is perturbed by ζ). (A non-hydrostatic version of this system is formulated in [30].) The next section discusses the consequences of these differences, focusing on the role of the continuity equation and its use in the code.

4 Continuity, Conservation, Constancy-Preservation, and Time-Splitting

It is convenient to rewrite (3.2) as

$$\frac{\mathscr{D}*}{\mathscr{D}t} = \frac{\partial}{\partial t}\left(\frac{H_z}{mn}*\right) + \frac{\partial\,(U*)}{\partial\xi} + \frac{\partial\,(V*)}{\partial\eta} + \frac{\partial\,(W*)}{\partial s}\,, \tag{4.1}$$

where we introduce the volumetric fluxes, $U = H_z u/n$, $V = H_z v/m$, and $W = \omega_s/(mn)$. The continuity equation (*cf.,* (3.4) with $q = \text{const}$) then is

$$\frac{\partial}{\partial t}\left(\frac{H_z}{mn}\right) + \frac{\partial U}{\partial\xi} + \frac{\partial V}{\partial\eta} + \frac{\partial W}{\partial s} = 0\,. \tag{4.2}$$

This W directly corresponds to W in the discretized Eq. (1.19) in SM2005 [the same as Eq. (21) in H2008], and it is the only one that appears in a hydrostatic code. Note that the horizontal components appear both as the momentum components u, v and volume fluxes U, V, while W does not have its lower-case counterpart [7]. The lower- and upper-case velocities can be traced back as *co-* and *contra*variant vector components that are distinguishable due to the non-orthogonality of the coordinate system.

Eq. (4.2) plays a dual role: (*i*) its vertical integral serves as a prognostic equation for ζ, and (*ii*) it allows the computation of vertical velocity W,

$$\frac{\partial\zeta}{\partial t} + mn\cdot\left(\frac{\partial\overline{U}}{\partial\xi} + \frac{\partial\overline{V}}{\partial\eta}\right) = 0 \qquad \text{and} \qquad W = \int_{-1}^{s}\left[\frac{\partial U}{\partial\xi} + \frac{\partial V}{\partial\eta}\right]\mathrm{d}s' - \frac{1}{mn}\cdot\frac{z+h}{\zeta+h}\cdot\frac{\partial\zeta}{\partial t}\,. \tag{4.3}$$

$\overline{U}$ and $\overline{V}$ are barotropic, volume-weighted fluxes. To derive (4.3) we use

$$\frac{\partial z}{\partial t} = \frac{\partial\zeta}{\partial t}\cdot\frac{z^{(0)}+h}{h} = \frac{\partial\zeta}{\partial t}\cdot\frac{z+h}{\zeta+h}\,, \tag{4.4}$$

a consequence of (2.10). This illustrates the meaning of W as the volumetric flux across the elemental area of the s-surface that moves up and down following the free surface according to (2.10). The ratio $(z+h)/(\zeta+h)$ does not actually depend on ζ, so any available time-state of the vertical coordinate $z = z(\xi,\eta,s)$ is suitable for its computation.

In the case of purely barotropic motion $u = \overline{u}$, $v = \overline{v}$ and a separable coordinate transformation ($z^{(0)} = C(s)\cdot h(\xi,\eta)$), the vertical velocity W vanishes. Since $z = z^{(0)} + \zeta\left(1 + z^{(0)}/h\right) = \zeta + D\cdot C(s)$

[7] This does appear in the non-hydrostatic model [30] and there only as w, the "true" vertical velocity rather than ω_s.

(with $D = h + \zeta$), hence $H_z = D \cdot C'(s)$ (with $C'(s) = dC(s)/ds$), the right Eq. (4.3) becomes

$$\begin{aligned} W &= \int_{-1}^{s} \left[\frac{\partial}{\partial \xi} \left(C'(s) \cdot \frac{D\overline{u}}{n} \right) + \frac{\partial}{\partial \eta} \left(C'(s) \cdot \frac{D\overline{v}}{m} \right) \right] \mathrm{d}s' - \frac{1}{mn} \cdot (C(s)+1) \cdot \frac{\partial \zeta}{\partial t} \\ &= (C(s)+1) \cdot \left[\frac{\partial \overline{U}}{\partial \xi} + \frac{\partial \overline{V}}{\partial \eta} \right] - \frac{1}{mn} \cdot (C(s)+1) \cdot \frac{\partial \zeta}{\partial t} \equiv 0 \,, \end{aligned} \tag{4.5}$$

because of the left-side Eq. (4.3) for ζ. This property reflects the fact that according to (2.10) the perturbation of $z^{(0)}$ is proportional to the distance from the bottom. So too is the vertical velocity (in the z-coordinate sense) of a barotropic flow, resulting in zero vertical velocity relative to the moving sigma-levels. Ultimately, we prefer (2.10) to (2.5) or any other alternative (*e.g.,* allowing only the uppermost grid box to change [8]).

In the case of the general, non-separable transformation (2.9), the vertical velocity W in a barotropic flow no longer vanishes as it does in (4.5). From this point of view, the statement made at the end of Sec. 3.2 in H2008 (top of p. 3601) [and a similar statement in Sec. 1.2 in SM2005] that "vertical mass fluxes generated by a purely barotropic motion vanish identically at every interface $z_{k+1/2}$" is not generally correct. However, the tendency of sigma-levels to move in sync with the vertical velocity (in the z-sense) of the barotropic flow remains valid in general, still making (2.10) the best possible choice. The property of canceling the vertical motion of the barotropic flow is useful for mode-splitting the PGF because sudden changes in ζ do not result in a large vertical velocity relative to the grid (hence no associated CFL violation) and do not cause spurious vertical redistribution of density. This justifies "freezing" $\overline{\rho}$ and ρ_* [Sec. 3.5 in H2008; Sec. 3.1 in SM2005] during the barotropic time-stepping.

The actual procedure as it appears in the code consists of two stages,

$$\widetilde{W} = \int_{-1}^{s} \left[\frac{\partial U}{\partial \xi} + \frac{\partial V}{\partial \eta} \right] \mathrm{d}s' \qquad \text{followed by} \qquad W = \widetilde{W} - \widetilde{W}\Big|_{s=0} \cdot \frac{z+h}{\zeta+h} \,. \tag{4.6}$$

This guarantees that surface and bottom no-flux boundary conditions, $W\big|_{s=0} = W\big|_{s=-1} = 0$, are satisfied. Its appearance somewhat hides the time-dependent nature of (4.3) because it does not contain the time derivative of ζ, but we can verify that they are equivalent. This equivalence is destroyed when continuous time is replaced with discrete time-stepping and time-splitting and when different time-stepping algorithms are used to advance different parts of the model equations. This is because the left-side (4.3) and the vertically-integrated momentum equation (3.1) comprise a system of equations for the barotropic mode that are advanced using a much smaller time step than used for the baroclinic fields and then fast-time-averaged for their slow baroclinic interactions. In contrast, the r.h.s. of (4.3) belongs to the slow baroclinic sub-system, and its $\partial\zeta/\partial t$ (and, implicitly, in (4.6)) is approximated using the slow-time $\langle\zeta\rangle$ and slow time-step Δt.

The procedure (4.6) is used during both predictor and corrector stages of the ROMS time step; however, it has two different meanings in these two stages. During the corrector stage (after the barotropic-mode stepping is complete and all five fast-time-averaged barotropic fields ($\langle\zeta\rangle^{n+1}$, $\langle\overline{U}\rangle^{n+1}$, $\langle\overline{V}\rangle^{n+1}$, $\left\langle\left\langle\overline{U}\right\rangle\right\rangle^{n+1/2}$, $\left\langle\left\langle\overline{V}\right\rangle\right\rangle^{n+1/2}$) are available and constructed such that

$$\frac{\langle\zeta\rangle^{n+1} - \langle\zeta\rangle^{n}}{\Delta t} + mn \cdot \left(\frac{\partial}{\partial \xi} \left\langle\left\langle\overline{U}\right\rangle\right\rangle^{n+1/2} + \frac{\partial}{\partial \eta} \left\langle\left\langle\overline{V}\right\rangle\right\rangle^{n+1/2} \right) = 0 \tag{4.7}$$

[*cf.,* Eq. (3.39) in Sec. 3.2 in SM2005]), the 3D fluxes $U^{n+1/2}$ and $V^{n+1/2}$ are corrected to yield the same vertical integral as $\left\langle\left\langle \overline{U} \right\rangle\right\rangle^{n+1/2}$ and $\left\langle\left\langle \overline{V} \right\rangle\right\rangle^{n+1/2}$. Then $W^{n+1/2}$ is computed via (4.6). The resulting set of fluxes ($U^{n+1/2}$, $V^{n+1/2}$, $W^{n+1/2}$) satisfies the finite-time-step version of (4.2),

$$\frac{H_z^{n+1} - H_z^n}{mn \cdot \Delta t} + \frac{\partial U^{n+1/2}}{\partial \xi} + \frac{\partial V^{n+1/2}}{\partial \eta} + \frac{\partial W^{n+1/2}}{\partial s} = 0 \tag{4.8}$$

[*cf.,* Eq. (19) in H2008, the same as Eq. (1.17) in SM2005]. This provides the basis for updating the tracer fields so they satisfy both conservation and constancy-preservation properties.

During the predictor stage (before the barotropic-mode time-stepping), the new-time-step ζ field is not available; hence, it is impossible to construct a W that satisfies an equation similar to (4.8). So W is computed via (4.6). The resulting set (U^n, V^n, W^n) is used to compute two auxiliary sets of grid-box thicknesses,

$$H_z^{n\pm1/2} = H_z^n \mp \frac{\Delta t}{2} \cdot mn \cdot \left(\frac{\partial U^n}{\partial \xi} + \frac{\partial V^n}{\partial \eta} + \frac{\partial W^n}{\partial s} \right) = H_z^n \mp \frac{\Delta t}{2} \cdot \mathrm{div}\mathbf{U}^n \,. \tag{4.9}$$

This is known as the *artificial continuity equation*. It is formally consistent with (4.2) with second-order temporal accuracy. However, inherent in (4.9) is the fact that its vertical integral corresponds to applying a time step $\pm\Delta t/2$ to the ζ equation that is normally advanced using a much smaller barotropic time step. Nevertheless, the resulting $H_z^{n\pm1/2}$ is appropriate because vertical integrals of U^n and V^n yield $\langle\overline{U}\rangle^n$ and $\langle\overline{V}\rangle^n$, respectively — a property inherited from the previous time step – and therefore do not contain fast-oscillating components dangerous for numerical stability [8] .

Once $H_z^{n-1/2}$ and $H_z^{n+1/2}$ are available, the tracer fields are updated as [9]

$$q^{n+1/2} = \frac{1}{H_z^{n+1/2}} \left[H_z^{n-1/2} \cdot \frac{q^n + q^{n-1}}{2} - \Delta t \cdot \mathrm{div}\left(\mathbf{U}^n q^n\right) \right] , \tag{4.10}$$

and $H_z^{n\pm1/2}$ is discarded. Equations (4.9)-(4.10) are constancy-preserving for tracer q. However, the integrated concentration is not conserved because there is no guarantee that $H_z^{n-1/2}$, computed during the next time step, is the same as $H_z^{n+1/2}$ from the previous one. Hence, this is a method to regain constancy-preservation at the expense of conservation. This is acceptable because the resulting fields $q^{n+1/2}$ are used exclusively to compute advective fluxes during the subsequent corrector stage. In SM2005 this method is called *pseudo-compressible* [10] .

---

[8] The appropriateness relies on the design of the filter for the fast-time-averaging of the barotropic mode.

[9] For simplicity we present the Leap-Frog-Trapezoidal Rule (LF-TR) version of the predictor-corrector algorithm. The actual code utilizes the LF-AM3 (Adams-Moulton 3rd order, a.k.a. closed parabolic integration rule) method that has a different set of coefficients for the q^n, q^{n-1}, and r.h.s. terms. Refer to Eqs. (4.7)-(4.8) in SM2005 for more detail.

[10] A philosophically similar, but not equivalent, method was proposed by [31] to recover the constancy-preservation property for a split-directional advection scheme written in conservation form. In the case of a non-divergent deformational flow, the directional splitting causes the component flows in consecutive directional updates to be divergent, resulting in a loss of constancy. An artificial continuity equation, essentially compression followed by subsequent expansion (or *vice versa*), was proposed to compensate for the loss.

5 Multiple Hydrodynamic Kernels in the ROMS Family

H2008 emphasizes early model developments leading toward ROMS as moving along the line of [32] $\rightarrow$ SH94 $\rightarrow$ [33] $\rightarrow$ DAMEE2000 (p. 3596 and refs. 1,3,4,6 in H2008; also Sec. 2 in DAMEE2000. We partly disagree with this as an oversimplification, both because of omission of notable references and also because of missing critical code components inherent to ROMS. The purposes of this section are to clarify the ROMS lineage and to reflect the present status of ROMS-family codes.

5.1 Early Developments

The earliest publication related to ROMS development is [semi-]Spectral Primitive-Equation Model (SPEM) [32] (hereafter HWY91). It introduces a sigma-coordinate model using Chebyshev polynomials in the vertical direction and a staggered (Arakawa C), orthogonal curvilinear grid in horizontal directions with second-order finite-differences. The barotropic mode is treated using a rigid-lid, streamfunction approach (via the prognostic barotropic vorticity equation). For computational efficiency SPEM does not perform spectral transforms — no efficient FFT-like procedure is available with Chebyshev — at every time step, but instead all discretized fields are defined at equivalent vertical co-location points σ_k at the locations of extrema of the highest-order polynomial used. The points have a natural tendency of refinement toward the ends, surface and bottom. However, for most oceanic situations surface-intensified shears flows are expected, which makes an asymmetric refinement (more toward the surface) more suitable, and it is rare to use high resolution in the parameterized turbulent boundary layers. SPEM provides some further flexibility by introducing an additional remapping $s_k = s(\sigma_k)$, where $s(\sigma)$ is a user-defined, continuously-differentiable function. The common operations of vertical differentiation and integration are implemented as matrix multiplies applied to the entire vertical column. The associated $N \times N$ matrices are not sparse, resulting in N^2 operations (N is the number of vertical polynomials), which effectively limits the model to modest vertical resolution. However, this was expected to be offset by the spectral accuracy in the vertical direction. For efficiency the matrices are precomputed at initialization and stored thereafter; so for this to be practical, they must be independent of local topography, which mandates the vertical sigma-transformation be strictly separable, $z = \dfrac{s(\sigma)-1}{2} \cdot h(x,y)$, hence $z_{i,j,k} = \dfrac{s_k - 1}{2} \cdot h_{i,j}$. Immediately after its introduction, SPEM was used in several studies for both idealized and realistic applications, notably U. S. West Coast simulation [34] with a modest horizontal resolution of $\sim 7km$, 128×80 horizontal grid with 8 vertical polynomials, and a semi-idealized coastal configuration (f-plane, periodic north-south with large-scale flow data supplied via relaxation). The model showed topographically-modulated instabilities of the coastal upwelling front and a poleward deep undercurrent — prominent features studied in a more realistic configuration a decade later [13]. Other studies used spectral SPEM dealt with topographically trapped waves and rectified flow around a seamount [35, 36].

Subsequently it was realized that a vertical Chebyshev-polynomial representation imposes a set of limitations that are too restrictive in practice. Spectral convergence requires smoothness of the fields on the grid scale, which is not often the case. The use of a separable σ-coordinate in combination with different discretizations of different directions (spectral in vertical and finite-difference in horizontal) leaves no other choice but to rely on the accuracy of approximation of each term individually in order to achieve hydrostatic error cancellation of pressure gradient terms (a central problem in sigma modeling). An effort was made to address this issue by increasing the order of accuracy of horizontal finite-differencing [37], but it never went beyond idealized test problems. The semi-spectral framework

makes it impossible, or at least very hard, to construct iso-geopotential or isopycnal lateral diffusion operators for tracers to avoid spurious diapycnal mixing where sigma surfaces cross density surfaces. Another limitation of the original SPEM is the lack of an implicit treatment of vertical viscosity and diffusivity terms, which results in a severe restriction on time step size [21]. This can be implemented efficiently as a matrix multiply only if the viscosity and diffusivity are known *a priori* and are constant in time, but in a more general case it requires an inversion of a full $N \times N$ matrix, albeit with a useful significant computational savings due to the increase in time step size [38]. (Note that near the surface and bottom, $\Delta\sigma = \sigma_{k=1} - \sigma_{k=2} \sim 1/N^2$, resulting in a CFL limitation for explicit vertical viscosity $\Delta t A_v/\Delta z^2 \sim \Delta t A_v \cdot N^4/h^2$.)

A major redesign of SPEM was introduced by [21], who replaced vertical Chebyshev polynomials with finite-differences, and at the same time, introduced several important algorithms, *viz.,* implicit vertical viscosity and diffusion, and lateral diffusion for tracers along geopotentials (rather than along sigma surfaces). At this stage also developed were algorithms for handling open-boundary conditions, allowing regional simulation with non-trivial inflow and outflow through the side boundaries. The resultant code was used for Southern Atlantic simulation [21, 22]. This was further refined by [23] using a stretched vertical coordinate similar to SH94 (except $\zeta \equiv 0$ due to rigid lid, *cf.,* (2.1) in Sec. 2 here) and staggering of vertical velocity relative to tracer points. A similar, but independent development yielding an all-finite-difference version SPEM was undertaken by [39], who also made a thorough comparison of its performance with the spectral version. These developments were consolidated into SPEM 5.1 code, which was used in series of studies, notably [40, 41]. Ultimately it participated in the DYNAMO Atlantic modeling intercomparison [24, 42]. Despite all these transformations, the original acronym "SPEM" was kept unchanged, and there are publications well after 2000 that still refer it as "semi-spectral", even though Chebyshev polynomials were out of use already for several years.

SH94 is another notable reference in the ROMS-related literature, and sometimes even used as the introductory reference to ROMS (*e.g.,* pp. 241 and 242 in DAMEE2000). The book [33] positions SH94 as a free-surface variant of SPEM (then existing as a purely finite-difference code, hence an alternative meaning for acronym SPEM is proposed on p. 133, replacing "semi-spectral" with "S-coordinate"), so the two, SPEM and SCRUM are described together. SH94 is credited for essentially two reasons: (i) its Eq. (2.16) introduces the S-coordinate to overcome the shortcomings of the standard separable σ-coordinate of POM and SPEM (Sec. 2.1) — which has been the most common reason for its citations — and (ii) its development of a free-surface model called SCRUM. However, in contrast with the SPEM developments mentioned above, the SCRUM code described in SH94 was never used for any simulations besides those presented in SH94 itself and the pressure-gradient error tests in [43]. Unlike the rigid-lid algorithm of SPEM which is well documented, [33] provides no detail about the free-surface, split-explicit algorithm of SCRUM other than briefly mentioning it on p. 136 (referring to Sec. 2.12 which is a general introduction to fractional-step methods) and in a footnote on p. 91 (citing [44] and explaining the necessity of temporal filtering of the barotropic mode in order to avoid numerical instability). In our own examination of a code presented to us in 2005 by the first author of SH94 (called SCRUM 2.1 and recorded as part of a US Copyrights registration dated April, 1996), there is no mechanism for baroclinic-barotropic mode-coupling that is mathematically consistent with the set of equations it is declared to solve. Specifically, the code does not contain a procedure for vertical integration of r.h.s. terms for the momentum equation to supply coupling terms for the barotropic mode in the way it occurs in a later SCRUM3.0 code of Hernan Arango (Fig. 1) nor as is stated in the last paragraph of Sec. 3.1 of SH94 starting with "In Eqs. (3.5)-(3.6), $\overline{uu}$, $\overline{uv}$, and $\overline{vv}$ are evaluated from the internal mode...". Nor does the code contain a fast-time-averaging procedure for the barotropic fields, instead using instantaneous barotropic velocities to correct the vertical integrals

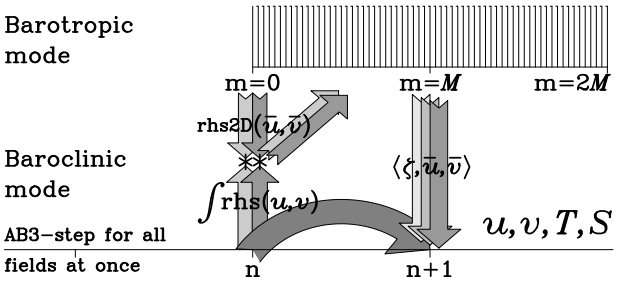

Fig. 1. Schematic diagram of SCRUM 3.0 code of Hernan Arango: r.h.s. terms for the barotropic momentum equations (small descending arrows on the left) are subtracted from the vertically-integrated r.h.s. for 3D momentum (ascending arrows) to yield forcing terms (diagonal arrows originating from $**$) for the barotropic mode. The barotropic mode is averaged over a $2\Delta t$ interval, so the outcome is time–centered at $n+1$.

of horizontal velocities for the 3D mode at the end of each main time step. No such procedure is ever mentioned in SH94. There is a striking difference between the topographically-constrained flow in Fig. 9 of SH94 and the earlier results using SPEM [34] for a similar problem. While SPEM develops strong current instabilities and filaments, no such features are present in SH94. From our own early experience with U. S. West Coast simulations we know that posing a purely barotropic problem results in topographic contour-following flow, while producing correct instabilities of an upwelling front requires correct coupling between barotropic and baroclinic modes. Another indication of lacking proper coupling between the barotropic and baroclinic modes is an almost purely baroclinic flow generated by pressure-gradient error in the Seamount test problem (see butterfly-shaped pattern in the upper-right panel of Fig. 1 in [45]). This strongly contrasts to the emergence of a significant barotropic component of the flow in the same problem reported by [46] (note the appearance of Sigma Error of the Second Kind [47, 6, 48, 49]).

5.2 Present-Day Codes

SM2005 loosely discusses different possibilities for building a code, but provides a detailed summary for only the most recent one at the time the article was written. Nevertheless, the other possibilities have since materialized but as yet received little attention in the published literature, besides merely citing SM2005 in a summary manner. Currently there are four distinct variants of the hydrodynamic time-stepping kernel for ROMS-family codes that are widely used.

1. Rutgers University ROMS (a.k.a. "official ROMS") (Fig. 2, *upper left*): This uses an AB3 time step for the momentum equation, inherited from an earlier code, SCRUM 3.0 by Hernan Arango in 1997, that is the direct ancestor of all ROMS-family codes (Fig. 1). In comparison with its predecessor, the code is characterized by

(*i*) a fast-time-averaging procedure for the barotropic mode using two sets of weights corresponding to $\langle . \rangle$- and $\langle\langle . \rangle\rangle$-averaging — a mechanism responsible for simultaneous conservation and constancy preservation for tracers regardless of the magnitude of the change in ζ from one time step to the next. Algorithmically the procedure allows a rather general choice of the shape of primary weights $\langle . \rangle$ as long as it properly normalized and centered at the baroclinic time step $n+1$. The secondary weights $\langle\langle . \rangle\rangle$ are always uniquely defined by the primary. There is no additional computational cost associated with the use of non-uniform weighting for $\langle . \rangle$ other than the necessity to extend computation of barotropic mode beyond $n+1$. It should also be noted there are publications, [44, 50, 51], which view $\langle . \rangle$-averaging as unnecessary and undesirable because it introduces numerical damping of time-resolved barotropic motions, and a effort was made to design a time-split algorithm without averaging (in our terms it is equivalent to setting $\langle . \rangle$ to delta-function centered at $n+1$, while $\langle\langle . \rangle\rangle$ becomes uniformly weighted averaging over the interval from n to $n+1$). However later [52] introduces rectangular-shaped averaging over

half of the baroclic time step, $\Delta t/2$, interval into his algorithm. This was done after realizing that in the presence of variable bottom topography mode splitting is never exact even for a linearized system, some measures are needed to ensure robustness of the algorithm. The rationale for selecting shape of $\langle . \rangle$ is provided in [2, 4], and is primarily motivated by the desire to achieve a more frequency selective fast-time filtering than that produced by a rectangular window;

(*ii*) a pseudo-compressible (using an artificial continuity equation), hence non-conservative, predictor step for tracers (4.9)-(4.10);

(*iii*) a forward extrapolation of baroclinic-to-barotropic mode forcing terms (using AB3 coefficients in this case) — a feature that avoids loss of second-order accuracy relative to just accepting their values computed at time step n and also improves code stability (***n.b.,*** no viscosity or the use of upstream-biased advection schemes is required for numerical stability even in situation where there is a significant barotropic component in the advection terms of the momentum equations, *e.g.,* in shallow coastal regions, *cf.,* [53], where it is found that the previous mode-splitting algorithms may experience difficulties. It its turn this allows much less dissipative, hence more accurate, fast-time-averaging);

(*iv*) a use of forward-backward feedback between momentum and tracer equations (the final update for tracers is delayed until the new-time-step velocities u, v^{n+1} become available, and they are used in an appropriate combination with their old-step counterparts to advect tracers; the particular algorithm used here maps onto AB3-TR generalized forward-backward scheme in Sec. 2.3 of SM2005); and

(*v*) taking into account the non-uniformity of density field when computing the PGF for the barotropic mode. This reduces the mode-splitting error relative to a commonly used simple shallow-water-like $-g\rho_0\nabla_x\zeta$ term.

Chronologically this is the first code in the ROMS family (first introduced in early 1998), and all its features (*i*)-(*iv*) are present in all ROMS codes. This is the code used in DAMEE2000 with the exception that the mode-splitting algorithm using $\overline{\rho}, \rho^*$ for the barotropic PGF was not available at that time: a simple $-gD\nabla\zeta$-term was used instead; the feature (*v*), the use of $\overline{\rho}, \rho^*$, was added in 2002. This updated version of ROMS is used for all the computations presented in Sec. 4 of H2008, and it is the basis for the adjoint code of ROMS [15].

2. LF-AM3 predictor-corrector main step with mode-coupling during the predictor stage (commonly known today as "AGRIF ROMS") (Fig. 2, *upper right*): This shares the mode-coupling mechanism with the Rutgers code, however it uses a predictor-corrector step for momentum equation as well. This improves the stability limit for internal waves (by approx. 1.6 times relative to AB3) at the expense of a mild increase in computational cost. Since tracer time-stepping was already using a predictor-corrector step and since the sole purpose of the predictor step is to provide values for the advective, Coriolis, and PGF terms during the corrector step, a simplified set of operations is used during the predictor step. For example, the viscous terms and mixing parameterizations are computed only once per time step, reducing their cost. Another new feature in this code is that r.h.s. terms are no longer stored from one time step to the next (as with AB3 stepping above), but rather combine LF stepping with AM3 interpolation during the predictor phase, so that the following corrector phase looks "logically-LF" as well (Fig. 3). The 2γ bias in setting the initial condition introduces a pre-distortion that cancels second-order truncation errors and yields an overall third-order accuracy (*cf.,* Eqs. (2.38)-(2.41) in SM2005 and the associated discussion). This code is used in [6], and it is the basis for AGRIF [11] , [19],

[11] Adaptive Grid Refinement In Fortran

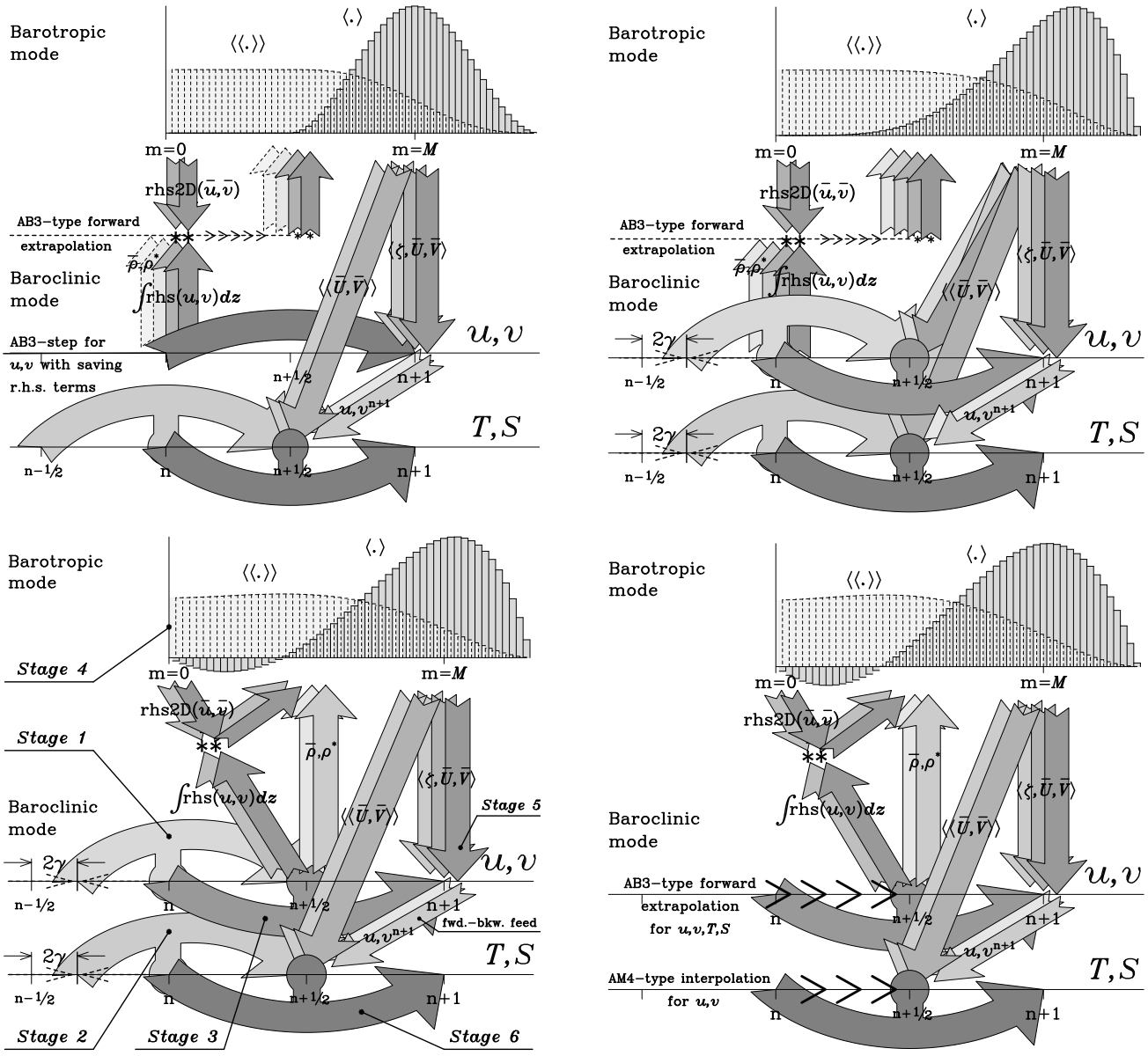

Fig. 2. Four variants of the ROMS kernel that are currently in use. *Upper left*: Rutgers University ROMS. *Upper right*: predictor-corrector main step with mode coupling during predictor stage. (This code is the basis for AGRIF.) *Lower left* predictor-corrector with coupling during corrector stage, same as in Sec. 5 of SM2005. (Note that the four arcs and barotropic mode on top correspond to SM2005 *Stage 1* to *6*). *Lower right*: generalized forward-backward main step. (This code is used as a basis for the non-hydrostatic code of [30].) *Legend*: The **arcs** (curved arrows) correspond to "steps"; *i.e.,* update of either momenta or tracers that involve computation of r.h.s. The **supporting pillar** near the middle of an arc connecting it to the circle indicates the timing of computating of r.h.s. terms needed to performed the step (also see Fig. 3 for explanation). **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. 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 arrow on top. The different **shape functions** for fast-time averaging are interchangeable and not tied to a specific code, but are separately presented here to illustrate their forms.

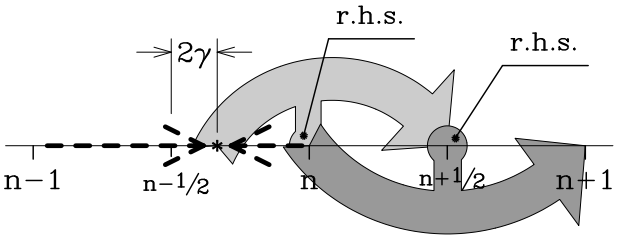

Fig. 3. Sub-diagram explaining the LF-AM3 step. First, using nth and $(n-1)$th values, the data is interpolated linearly to $n-1/2+2\gamma$, which is used as the initial condition ($\gamma = 1/12$ for LF-AM3; $\gamma = 0$ for LF-TR). Then it is advanced to $n+1/2$ using r.h.s. terms computed at nth step (predictor). Subsequently the nth field is advanced to $n+1$ using the r.h.s.at $n+1/2$ (corrector).

maintained and developed at IRD [12] .

3. LF-AM3 main step with coupling during corrector stage (commonly known as "UCLA ROMS") (Fig. 2, *lower left*): This is the same as in Sec. 5 of SM2005 and is used in ongoing research at UCLA. Its main time-stepping is similar to the previous one, but the mode-coupling mechanism is redesigned to eliminate the need for AB3 extrapolation of the baroclinic-to-barotropic forcing terms; since this occurs during the corrector stage, all relevant terms are already at the $n+1/2$ baroclinic step. This code involves a complete redesign of the barotropic mode itself; the LF-TR/AM3 stepping is replaced with a generalized forward-backward scheme, yielding a reduction in the computational cost by more than a factor of two. Because the computation of r.h.s. terms for the 3D momentum equations during corrector step uses "mixed-state" u, v fields (their baroclinic components are already advanced to $n+1/2$, but vertical averaged still correspond to main time step n), to maintain numerical stability for barotropically-dominated advective terms without resorting to explicit viscosity or upstream-biased advection schemes, this variant of the code requires recomputing of advective and Coriolis terms during barotropic time stepping (in the two previous variants, Rutgers and AGRIF, this is optional, since AB3-extrapolation of coupling terms ensures stability with respect to advection and Coriolis; however in all practical applications the recomputing is always done in these codes as well). For computational efficiency a simple second-order centered scheme is used for barotropic momentum advection terms, while a 3D advection is computed using a higher-order scheme (3rd-order upstream-biased, or 4th-order centered). Despite this, the overall spatial accuracy for barotropically-dominated advection terms is higher than second because the barotropic mode receives the higher-order correction terms (the difference between 3rd- or 4th-order vertically integrated 3D and second-order 2D advection) through the mode-coupling algorithm (diagonal ascending arrows on Fig. 2, *lower left*). The elimination of AB3-extrapolation offers the flexibility in applying different time stepping for different terms which influence barotropic mode, for example for the first time this variant of ROMS kernel allows fully implicit treatment of bottom drag terms, while avoiding loss of accuracy associated with mode splitting (this is reminiscent with the long-standing dilemma of accurate treatment of no-slip boundaries in incompressible flows using projection method, [54]). Also note the appearance of negative lobe in the shape of primary averaging weights $\langle . \rangle$. This choice formally retains the second-order accuracy for fast-time-filtered barotropic mode – hence achieve a more physically accurate representation of barotropic motions which are resolved by baroclinic time step.

4. Generalized forward-backward algorithm for the main time step (Fig. 2, *lower right*): This shares the mode-coupling mechanism with UCLA code, from which it was developed, but replaces the predictor step with AB3-type extrapolation of prognostic variables toward $n+1/2$. This variant is used as a basis for the non-hydrostatic code of [30]. The motivation to eliminate the predictor stage comes from the fact that it is required to be a non-hydrostatic step as well, in order to yield extra accuracy, resulting in solving the pressure-Poisson equation for non-hydrostatic pressure one more time per

[12] Institute for Research and Development, France

Table 1: Intercomparison of ROMS-family codes.

	SCRUM 3.0	Rutgers ROMS	AGRIF	UCLA	Non-hydrostatic
origin	Rutgers	UCLA–Rutgers	UCLA–IRD	UCLA	UCLA
maintained	obsolete	by Rutgers	by IRD	by UCLA	
introd. year [13]	1997	1998	1999 [14]	2002	2006
time-stepping algorithms and theoretical stability limits [15]					
coupl. stage [16]			predictor	corrector [17]	
barotropic mode	LF-TR	LF-AM3 with forw.-backw. feed. [18]		Generalized FB (AB3–AM4)	
$\alpha_{\max}$, barotr.	$\sqrt{2}$, (2) [19]	1.85 , (2)		1.78 , (1)	
3D momenta	AB3	AB3	LF-AM3		AB3 (mod)
tracers	AB3	LF-TR	LF-AM3		AB3 (mod)
intern. waves	AB3	Gen. FB (AB3-TR) [20]	LF-AM3, forw.-backw. feedback		Gen. FB (AB3–AM4)
$\alpha_{\max}$, advect.	0.72	0.72	1.587		0.78
$\alpha_{\max}$, Coriolis	0.72	0.72	1.587		0.78
$\alpha_{\max}$, int. w.	0.72, (1)	1.14 ; (1,2)	1.85, (2)		1.78 , (1)
storage [21]	4,4	4,3	3,3		4,4
miscellaneous code features and related developments					
parallelization	none	`MPI` or `OpenMP` (user selectable)		`MPI+OpenMP` (incl. hybrid)	`MPI` only [22]
nesting	N/A	off-line	on-line	off-line	
data assim.	N/A	Adjoint [15]	3DVar [16, 55, 56]	no	no
introductory reference [23]	none [24]	DAMEE2000	[2, 19] [25]	SM2005	[30]

time step. The entire mode-coupling mechanism is redesigned because the barotropic mode is also influenced by the non-hydrostatic pressure, and *vice versa*. The non-hydrostatic version of this code has an additional prognostic equation for vertical velocity w that is advanced in time simultaneously with (u, v), and solution of a pressure-Poisson equation to enforce non-divergence at $n+1$ inserted just after the completion of barotropic mode-stepping but before the update of tracers. Also, compared to the hydrostatic version, the barotropic mode in the non-hydrostatic code receives an additional forcing term, the vertically integrated non-hydrostatic PGF (see [30] for the complete diagram). The target applications for this code are on very small scales (with horizontal grid resolution in the sub-kilometer range), so the expected time step is very small and limited primarily by internal gravity waves (rather than Coriolis/inertial oscillations). This regime favors the use of a generalized forward-backward step over a predictor-corrector.

The algorithmic properties of the four ROMS-family codes are further summarized in Table 1.

Acknowledgments: We appreciate sustained support for ROMS development by the Office of Naval Research (currently grant N00014-08-1-0597).

References:

[1] D. B. Haidvogel, H. Arango, W. P. Budgell, B. D. Cornuelle, E. Curchitser, E. Di Lorenzo, K. Fennel, W. R. Geyer, A. J. Hermann, L. Lanerolle, J. Levin, J. C. McWilliams, A. J. Miller, A. M. Moore, T. M. Powell, A. F. Shchepetkin, C. R. Sherwood, R. P. Signell, J. C. Warner, and J. Wilkin. Ocean forecasting in terrain-following coordinates: Formulation and skill assessment of the Regional Ocean Modeling System. *J. Comput. Phys.*, 227:3595–3624, 2008. doi:10.1016/j.jcp.2007.06.016.

[2] A. F. Shchepetkin and J. C. McWilliams. The regional oceanic modeling system (ROMS): A split-explicit, free-surface, topography-following-coordinate oceanic model. *Ocean Modelling*, 9(4):347–404, 2005. doi:10.1016/j.ocemod.2004.08.002.

[13] The introduction year is the first year when a functional code first became available. This does not necessarily mean that it had all its features at that time (*e.g.,* in 1998 Rutgers ROMS did not use a variable density in the barotropic mode, and this feature was added later in 2002.)

[14] Prototype code only. The on-line nesting capability (*i.e.,* it *became* AGRIF as it is known today) was developed in 2001.

[15] The theoretical stability limit is defined with respect to a specific physical process taken alone, *i.e.,* without accounting for interference among different processes nor for the effects of spatial discretization. The latter is assumed to be "ideal", *i.e.,* centered (hence non-dissipative), infinite-accuracy (nondispersive), and one-dimensional. Under these conditions $\alpha_{\max}$ in the table matches the definition in Sec. 2 SM2005. For the barotropic mode $\alpha_{\max}$ is always limited by surface gravity waves. For the baroclinic mode three different limits are reported.

[16] Applies only to codes which use predictor-corrector stepping for the 3D momentum equations.

[17] The UCLA code can function both in predictor-coupled and corrector-coupled modes.

[18] The generalized FB barotropic mode was ported into the newest AGRIF code at the end of 2007.

[19] The number in parentheses (*e.g.,* in $\sqrt{2},\ (2)$) indicates the number of r.h.s. computations per time step. If there are two parenthesized numbers, the first one is for momenta, the second for tracers.

[20] AB3-TR is a Generalized Forward-Backward (FB) scheme corresponding to Eq. (2.49) from SM2005 with $\beta = 5/12$, $\delta = 1/2$, and $\gamma = \epsilon = 0$. Note that the LF-predictor step for tracers of Rutgers ROMS is outside the FB-feedback loop between momentum and tracer equations.

[21] Storage is the number time slices in 3D arrays needed to implement time stepping. The first figure refers to momentum equations, the second to tracers.

[22] Limited by the use of HYPRE library to solve pressure Poisson equation. This library does not support OpenMP parallelization at this time.

[23] Introductory reference is either the first reference where the algorithms of a code are described in detail as the primary purpose of the article, or, if not available, the first documented use of the code that indicates its existence at the time of publication. The algorithmic references [6] and SM2005 apply to the entire ROMS family.

[24] see Sec. 5.1.

[25] In 2003 the manuscript later known as SM2005 was submitted into *J. Comput. Phys.* where it received reference number JCOMP-D-03-00102. It was rejected soon after citing excessive length, complexity, and lack on new material. It coincides with SM2005 in more than 90word-by-word identical, including typos, in the portions which were copy-pasted into H2008), however, unlike SM2005, it contains a detailed description of the predictor-coupled time-stepping of AGRIF together with the corrector-coupled algorithm of UCLA ROMS. The description was excluded from the final, published version of SM2005 because of the desire to reduce article length, and because we consider the corrector-coupled version superior.

[3] D. B. Haidvogel, H. Arango, K. Hedstrom, A. Beckmann, P. Malanotte-Rizzoli, and A. F. Shchepetkin. Model evaluation experiments in the North Atlantic Basin: Simulations in non-linear terrain-following coordinates. *Dyn. Atmos. Oceans*, 32:239–281, 2000. doi:10.1016/S0377-0265(00)00049-X.

[4] A. F. Shchepetkin and J. C. McWilliams. Computational kernel algorithms for fine-scale, multi-process, long-term oceanic simulations. In R. Temam and J. Tribbia, editors, *Handbook of Numerical Analysis: Computational Methods for the Ocean and the Atmosphere*, volume 14, pages 121–183. Elsevier Science, 2009. doi:10.1016/S1570-8659(08)01202-0.

[5] Y. T. Song and D. Haidvogel. A semi-implicit ocean circulation model using a generalized topography following coordinate system. *J. Comput. Phys.*, 115:228–244, 1994. doi:10.1006/jcph.1994.1189.

[6] A. F. Shchepetkin and J. C. McWilliams. A method for computing horizontal pressure-gradient force in an oceanic model with a nonaligned vertical coordinate. *J. Geophys. Res.*, 108:3090, 2003. doi:10.1029/2001JC001047.

[7] Piotr K. Smolarkiewicz. A fully multidimensional positive-definite advection transport algorithm with small implicit diffusion. *J. Comput. Phys.*, 54:325–362, 1984. doi:10.1016/0021-9991(84)90121-9.

[8] S. M. Griffies, R. C. Pacanowski, M. Schmidt, and V. Balaji. Tracer conservation with an explicit free surface method for z-coordinate ocean models. *Month. Weath. Rev.*, 129:1081–1098, 2001. doi:10.1175/1520-0493(2001)129<1081:TCWAEF>2.0.CO;2.

[9] A. Adcroft and J.-M. Campin. Rescaled height coordinates for accurate representation of free-surface flows in ocean circulation models. *Ocean Modelling*, 7:269–284, 2004. doi:10.1016/j.ocemod.2003.09.003.

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

[11] C. J. Willmott. On the validation of models. *Phys. Geogr.*, 2:184–194, 1981.

[12] C. J. Willmott. Some comments on the evaluation of model performance. *Bull. Amer. Meteorol. Soc.*, 63:1309–1313, 1981.

[13] P. Marchesiello, J. C. McWilliams, and A. F. Shchepetkin. Equilibrium structure and dynamics of the California Current system. *J. Phys. Oceanogr.*, 33:753–783, 2003. doi:10.1175/1520-0485(2003)33<753:ESADOT>2.0.CO;2 .

[14] E. N. Curchitser, D. B. Haidvogel, A. J. Hermann, E. L. Dobbins, T. M. Powell, and A. Kaplan. Multi-scale modeling of the North Pacific Ocean: Assessment and analysis of simulated basin-scale variability (1996-2003). *J. Geophys. Res.*, 110:C11021, 2005. doi:10.1029/2005JC002902.

[15] A. M. Moore, H. G. Arango, E. Di Lorenzo, B. D. Cornuelle, A. J. Miller, and D. J. Neilson. A comprehensive ocean prediction and analysis system based on the tangent linear and adjoint of a regional ocean model. *Ocean Modelling*, 7:227–258, 2004. doi:10.1016/j.ocemod.2003.11.001.

[16] Z. Li, Y. Chao, J. C. McWilliams, and K. Ide. A three-dimensional variational data assimilation scheme for the Regional Ocean Modeling System in support of Coastal Ocean Observing System. In *Proc. of SPIE: Assimilation of Remote Sensing and In Situ Data in Modern Numerical Weather and Environmental Prediction Models*, number 6685, 2007. 66850B1-B12.

[17] R. A. Locarnini, A. V. Mishonov, J. I. Antonov, T. P. Boyer, and H. E. Garcia. World Ocean Atlas 2005, Vol. 1: Ttemperature. In S. Levitus, editor, *NOAA Atlas NESDIS 61*. U.S. Gov. Printing Office,Washington, D.C., 2006. 182pp.

[18] L. R. Centurioni, J. C. Ohlmann, and P. P. Niiler. Permanent meanders in the california current system. *J. Phys. Oceanogr.*, 38:1690–1710, 2008. doi:10.1175/2008JPO3746.1.

[19] P. Penven, L. Debreu, P. Marchesiello, and J. C. McWilliams. Evaluation and application of the ROMS 1-way embedding procedure to the central California upwelling system. *Ocean Modelling*, 12:157–187, 2006. doi:10.1016/j.ocemod.2005.05.002.

[20] X. Capet, J. C. McWilliams, M. J. Molemaker, and A. F. Shchepetkin. Mesoscale to submesoscale transition in the California Current System. Part I: Flow structure, eddy flux, and observational tests. *J. Phys.*

Oceanogr., 38:29–43, 2008. doi:10.1175/2007JPO3671.1.

[21] B. Barnier, P. Marchesiello, A. P. de Miranda, J. M. Molines, and M. Coulibaly. A sigma-coordinate primitive equation model for studying the circulation in the South Atlantic. Part I: Model configuration with error estimates. *Deep-Sea Res. I*, 45:543–572, 1998. doi:10.1016/S0967-0637(97)00086-1.

[22] P. Marchesiello, B. Barnier, and A. P. de Miranda. A sigma-coordinate primitive equation model for studying the circulation in the South Atlantic. Part II: Meridional transports and seasonal variablity. *Deep-Sea Res.*, 45:543–572, 1998. doi:10.1016/S0967-0637(97)00087-3.

[23] A. P. de Miranda, B. Barnier, and W. K. Dewar. Mode waters and subduction rates in high-resolution South Atlantic simulation. *J. Mar. Res.*, 58:213–244, 1999.

[24] J. Willebrand, B. Barnier, C. Böning, C. Dietrich, P. D. Killworth, C. Le Provost, Y. Jia, J.-M. Molines, and A. L. New. Circulation characteristics in three eddy-permitting models of the North Atlantic. *Progress in Oceanography*, 48:123–161, 2001. doi:10.1016/S0079-6611(01)00003-9.

[25] A. F. Shchepetkin and J. C. McWilliams. Quasi-monotone advection schemes based on explicit locally adaptive dissipation. *Month. Weath. Rev.*, 126:1541–1580, 1998. doi:10.1175/1520-0493(1998)126<1541:QMASBO>2.0.CO;2 .

[26] A. F. Blumberg and G. L. Mellor. A description of a three-dimensional coastal ocean circulation model. In N. Heaps, editor, *Editor, Three-dimensional Coastal Ocean Models*, pages 1–16. AGU Press, 1987.

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

[28] D. R. Jackett and T. J. McDougall. Minimal adjustment of hydrostatic profiles to achieve static stability. *J. Atmos. Oceanic Technology*, 12:381–389, 1995. doi:10.1175/1520-0426(1995)012<0381:MAOHPT>2.0.CO;2.

[29] J. K. Dukowicz. Reduction of density and pressure gradient errors in ocean simulations. *J. Phys. Oceanogr.*, 31:1915–1921, 2001. doi:10.1175/1520-0485(2001)031<1915:RODAPG>2.0.CO;2 .

[30] Y. Kanarska, A. Shchepetkin, and J. C. McWilliams. Algorithm for non-hydrostatic dynamics in the Regional Oceanic Modeling System. *Ocean Modelling*, 18:143–174, 2007. doi:10.1016/j.ocemod.2007.04.01.

[31] R. C. Easter. Two modified versions of Bott's positive-definite numerical advection scheme. *Month. Weath. Rev.*, 121:297–304, 1993.

[32] D. B. Haidvogel, J. L. Wilkin, and R. Young. A semi-spectral primitive-equation ocean circulation model using vertical sigma and orthogonal cirvilinear horizontal coordinates. *J. Comput. Phys.*, 94:151–185, 1991. doi:10.1016/0021-9991(91)90141-7.

[33] D. B. Haidvogel and A. Beckmann. *Numerical ocean circulation modeling*. Imperial College Press, London, 1999.

[34] D. B. Haidvogel, A. Beckmann, and K. S. Hedström. Dynamical simulations of filament formation and evolution in the coastal transition zone. *J. Geophys. Res.*, 96:15,017–15,040, 1991. doi:10.1029/91JC00943.

[35] A. Beckmann and D. B. Haidvogel. Numerical simulation of flow around a tall isolated seamount Part I. Problem formulation and model accuracy. *J. Phys. Oceanogr.*, 23:1736–1753, 1993. doi:10.1175/1520-0485(1993)023<1736:NSOFAA>2.0.CO;2.

[36] D. B. Haidvogel, A. Beckmann, D. C. Chapman, and R.-Q. Lin. Numerical simulations of flow around tall isolated seamount: Part ii: Resonant generation of trapped waves. *J. Phys. Oceanogr.*, 23:2373–2391, 1993. 10.1175/1520-0485(1993)023<2373:NSOFAA>2.0.CO;2 .

[37] J. D. McCalpin. A comparison of second-order and fourth-order pressure gradient algorithms in a σ-coordinate ocean model. *Int. J. Num. Meth. Fluids*, 18:361–383, 1994. doi:10.1002/fld.1650180404.

[38] Y. Wang and K. Hutter. A semi-implicit semispectral primitive-equation model for lake circulation dynamics and its stability and performance. *J. Comput. Phys.*, 139:209–241, 1998.

[39] L. Umlauf, Y. Wang, and J. Hutter. Comparing two topography-following primitive equation models for

lake circullation. *J. Comput. Phys.*, 153:638–659, 1999.

[40] Jean-Marie Beckers. On some stability properties of the discretization of damped propagation of shallow-water inertia-gravity waves on the Arakawa B-grid. *Ocean Modelling*, 1:53–69, 1999. doi:10.1016/S1463-5003(99)00005-0.

[41] T. Penduff, B. Barnier, M.-A. Kerbiriou, and J. Verron. How topographic smoothing contributes to differences between the eddy flows simulated by sigma- and geopotential-coordinate models. *J. Phys. Oceanogr.*, 32:122–137, 2002. doi:10.1175/1520-0485(2002)032<0122:HTSCTD>2.0.CO;2.

[42] B. Barnier, T. Reynaud, A. Beckmann, C. W. Böning, J.-M. Molines, S. Barnard, and Y. Jia. On the seasonal variability and eddies in the North Brazil Current: Insights from model intercomparison experiments. *Progress in Oceanogr.*, 48:195–230, 2001.

[43] Y. T. Song. A general pressure gradient formulation for ocean models. Part I: Scheme design and diagnostic analysis. *Month. Weath. Rev.*, 126:3213–3230, 1998.

[44] R. L. Higdon and A. F. Bennett. Stability analysis of operator splitting for large-scale ocean modelling. *J. Comput. Phys.*, 123:311–329, 1996. doi:10.1006/jcph.1996.0026.

[45] Y. T. Song and D. G. Wright. A general pressure gradient formulation for ocean models. Part II: Energy, momentum, and bottom torque consistency. *Month. Weath. Rev.*, 126:3231–3247, 1998.

[46] G. L. Mellor, L.-Y. Oey, and T. Ezer. Sigma coordinate pressure gradient errors and the seamount problem. *J. Atmos. Oceanic Technology*, 15:1122–1131, 1998. doi:10.1175/1520-0426(1998)015<1122:SCPGEA>2.0.CO;2.

[47] J. Berntsen. Internal pressure errors in sigma-coordinate ocean modeling. *J. Phys. Oceanogr.*, 19:1403–1414, 2002.

[48] Jarle Berntsen and Gunnar Furnes. Internal pressure errors in sigma-coordinate ocean – sensitivity of the growth of the flow to time stepping method and possible nonhydrostatic effects. *Cont. Shelf Res.*, 25:829–848, 2005. doi:10.1016/j.csr.2004.09.025 .

[49] A. C. Ciappa. An operational comparative test of z-levels PGF schemes to reduce pressure gradient errors of the ocean model POM. *Ocean Modelling*, 12:80–100, 2006. doi:10.1016/j.ocemod.2005.04.002.

[50] R. L. Higdon and R. A. de Szoeke. Barotropic-baroclinic time splitting for ocean circullation modeling. *J. Comput. Phys.*, 135:30–53, 1997. doi:10.1006/jcph.1997.5733.

[51] R. Hallberg. Stable split time stepping schemes for large-scale ocean modeling. *J. Comput. Phys.*, 135:54–65, 1997. doi:10.1006/jcph.1997.5734.

[52] R. L. Higdon. Implementation of a barotropic-baroclinic time splitting for isopycnic coordinate ocean modeling. *J. Comput. Phys.*, 148:579–604, 1999. doi:10.1006/jcph.1998.6130.

[53] Y. Morel, R. Baraille, and A. Pichon. Time splitting and linear stability of the slow part of the barotropic component. *Ocean Modelling*, 23:73–81, 2008. doi:10.1016/j.ocemod.2008.04.001.

[54] John K. Dukowicz and Arkady S. Dvinsky. Approximate factorization as a high order splitting for the implicit incompressible flow equations. *J. Comput. Phys.*, 102:336–347, 1992. doi:10.1016/0021-9991(92)90376-A.

[55] Z. Li, Y. Chao, J. C. McWilliams, and K. Ide. A three-dimensional variational data assimilation system for the Regional Ocean Modeling System. *J. Atmos. Oceanic Technology*, 25:2074–2090, 2008. doi:10.1175/2008JTECHO594.1.

[56] Z. Li, Y. Chao, J. C. McWilliams, and K. Ide. A three-dimensional variational data assimilation scheme for the Regional Ocean Modeling System: Implementation and basic experiments. *J. Geophys. Res.*, 113:C0500, 2008. doi:10.1029/2006JC004042.