ELSEVIER
Available online at www.sciencedirect.com
ScienceDirect
Ocean Modelling xxx (2008) xxx–xxx
Ocean Modelling
www.elsevier.com/locate/ocemod

SELFE: A semi-implicit Eulerian–Lagrangian finite-element model for cross-scale ocean circulation

Yinglong Zhang *, António M. Baptista

OGI School of Science and Engineering, Oregon Health ~~and~~ Science University, 20000 NW Walker Road, Beaverton, OR 97006, USA

Received 25 April 2007; received in revised form 20 November 2007; accepted 21 November 2007

Abstract

Unstructured-grid models grounded on semi-implicit, finite-volume, Eulerian–Lagrangian algorithms, such as UnTRIM and ELCIRC, have enjoyed considerable success recently in simulating 3D estuarine and coastal circulation. However, opportunities for improving the accuracy of this type of models were identified during extensive simulations of a tightly coupled estuary–plume–shelf system in the Columbia River system. Efforts to improve numerical accuracy resulted in SELFE, a new finite-element model for cross-scale ocean modeling. SELFE retains key benefits, including computational efficiency of existing semi-implicit Eulerian–Lagrangian finite-volume models, but relaxes restrictions on grids, uses higher-order shape functions for elevation, and enables superior flexibility in representing the bathymetry. Better representation of the bathymetry is enabled by a novel, "localized" vertical grid that resembles unstructured grids. At a particular horizontal location, SELFE uses either *S* coordinates or *SZ* coordinates, but the equations are consistently solved in *Z* space. SELFE also performs well relative to volume conservation and spurious oscillations, two problems that plague some finite-element models. This paper introduces SELFE as an open-source code available for community use and enhancement. The main focus here is on describing the formulation of the model and on showing results for a range of progressively demanding benchmark tests. While leaving details to separate publications, we also briefly illustrate the superior performance of SELFE over ELCIRC in a field application to the Columbia River estuary and plume.

Keywords: Cross-scale ocean modeling; Estuaries; Plumes; Finite elements; Semi-implicit time stepping; Eulerian–Lagrangian methods

1. Introduction

Numerical modeling of ocean circulation, at scales ranging from estuaries to ocean basins, is a mature field. A plethora of codes are available, many of which are open-source. Most modern ocean circulation codes solve for some form of the 3D Navier–Stokes equations, complemented with conservation equations for water volume, salt and heat. Common codes use either structured (POM (Blumberg and Mellor, 1987); TRIM (Casulli and Cheng, 1992); ROMS (Shchepetkin and McWilliams, 2005); NCOM (Barron et al., 2006)) or unstructured grids (ADCIRC (Luettich et al., 1991); QUODDY (Lynch and Werner, 1991); UnTRIM (Casulli and Walters, 2000); ELCIRC (Zhang et al., 2004); SEOM (Iskandarani et al., 2003); FVCOM (Chen et al., 2003)) and are typically based on finite differences (POM, TRIM, ROMS, NCOM), finite elements (SEOM, ADCIRC, QUODDY), or hybrid approaches involving finite volumes (UnTRIM, ELCIRC, FVCOM).

However, modeling circulation *across* ocean scales still poses serious challenges and remains an open issue in modern oceanography. Particularly challenging is the modeling of river–estuary–plume–shelf systems, given the range of processes and the tight coupling among the temporal and spatial scales involved. Although a domain nesting with same or different models remains an option, for these systems there is a clear incentive to develop cross-scale circulation models that can extend from the estuary into the

* Corresponding author. Tel.: +1 503 748 1960; fax: +1 503 748 1273.
E-mail address: yinglong@stccmop.org (Y. Zhang).

1463-5003/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ocemod.2007.11.005

shelf and beyond. Properly addressing the modeling of river–estuary–plume–shelf systems is particularly pertinent at a time when ocean observing systems (Baptista, 2006; Martin, 2003; Clark and Isern, 2003) are being implemented across the United States and around the world. Indeed, inherent to the concept of ocean observing is ubiquitous land-to-ocean modeling, in the form of near real-time forecasting and climate-scale simulation databases.

The main challenge to a cross-scale circulation model is to resolve the complex geometry and bathymetry commonly found in coasts, estuaries, tidal flats and rivers in an accurate, efficient and robust way, while maintaining adequate resolution in the deep ocean. Circulation models based on an unstructured grid are ideal for this, but so far, there have been few that can meet the challenge mentioned above, despite considerable research efforts that have gone into developing such models. Models based on fully 3D unstructured grids (Cheng et al., 2000; Labeur and Pietrzak, 2005; Danilov et al., 2004) are so far too expensive for large applications. Models based on the explicit mode splitting technique (POM, ROMS, FVCOM, ADCIRC, QUODDY, SEOM), in addition to having errors associated with the splitting of the internal and external modes (Shchepetkin and McWilliams, 2005), suffer from numerical stability constraints (e.g., the Courant–Friedrich–Lewy (CFL) condition) that restrict the maximum allowable time step and thus the size of the problem.

Since the 1990s, a family of semi-implicit unstructured grid models (UnTRIM; SUNTANS (Fringer et al., 2006); ELCIRC), hereafter referred to as "UnTRIM-like models", have shown great promise as the new generation of cross-scale circulation models – a promise reinforced by variations[1] to these models such as Walters (2005) and Leupi and Altinakar (2005). The UnTRIM-like models treat implicitly the terms – barotropic-pressure gradient, vertical viscosity in the momentum equations, and divergence term in the continuity equation – that place the most severe constraints on numerical stability (e.g., CFL condition), and treat all other terms explicitly. With this approach, there is no mode splitting into external and internal modes. Moreover, the resulting matrix is positive definite, symmetric and sparse, and therefore very efficient solvers (e.g., Jacobian Conjugate Gradient) can be used that guarantee fast convergence. As a result, most severe numerical constraints are by-passed, resulting in greater numerical efficiency. However, because piecewise constant shape functions are used to represent the elevation, over dissipation may occur, as shown for ELCIRC by Baptista et al. (2005). In addition, due to the finite-difference method used in UnTRIM-like models, grid orthogonality is required (Casulli and Walters, 2000), which has important implications for convergence and accuracy, as we will demonstrate in this paper. Ham et al. (2005) proposed a path integral method that does not require grid orthogonality; however, in their method, the resulting matrix is no longer symmetric positive definite, thus destroying one of the main advantages of UnTRIM-like models.

The opportunity for improving ELCIRC was well illustrated by its application to the Columbia River estuary–plume–shelf system. The Columbia River plume is a major oceanographic feature of the eastern boundary of the North Pacific Ocean (Hickey et al., 1998; Hickey and Banas, 2003). Ranked second in annual river discharge in the United States (first worldwide among rivers without a delta), the Columbia River has extensive tidal flats, strong tides, and very large velocities and velocity gradients coupled with extreme density gradients. Depending on the river discharge, the estuary can change from well mixed to highly stratified, salt wedge conditions (Jay and Smith, 1990a,b). The Columbia River plume extends hundreds of kilometers into the ocean along a continental shelf that is subject to strong wind-driven upwelling and downwelling regimes, the influence of which is felt deep in the estuary.

In spite of the complexity of the Columbia River, the robustness and computational efficiency of ELCIRC has enabled daily forecasts and multi-year simulation databases of 3D baroclinic circulation in the estuary and plume (Baptista et al., 2005) to become core capabilities of an observing system (CORIE, Baptista, 2006). By contrast, our prior attempts to study the system with 3D baroclinic models (QUODDY and POM), circa 1996–1999, were largely unsuccessful due to computational constraints or to limitations in representing key circulation processes and scales (e.g., wetting and drying of tidal flats).

Although the successful use of ELCIRC in CORIE and in other systems (Pinto et al., 2004; Foreman et al., 2006) demonstrates the model's ability to capture important features of complex cross-scale circulation, limitations persist. Figs. 1 and 2 illustrate two complementary aspects of one such limitation in the context of the application to the Columbia River: ELCIRC tends to under-predict the intrusion of salt in the estuary (Fig. 1), resulting in and being augmented by general overestimation of the plume freshness, despite qualitative plume features being well captured (Fig. 2).[2]

The need to further improve ELCIRC triggered the development of SELFE (Semi-implicit Eulerian–Lagrangian Finite Element). SELFE retains the robustness and computational efficiency of ELCIRC, while eliminating grid orthogonality requirements, allowing higher-order shape functions to be used, and enabling superior flexibility in representing bathymetry and vertical structure of the water column (see the discussions in the next paragraph). As will be documented in this paper and elsewhere, and

[1] While the UnTRIM-like models use a finite-difference/finite-volume method, the models of Walters (2005), Leupi and Altinakar (2005), and Miglio et al. (1999) use the lowest-order Raviart–Thomas element in the horizontal direction. The resulting equations are essentially the same as those in UnTRIM-like models.

[2] While ELCIRC results can be improved through aggressive grid refinement, we only found marginal improvement due to its slow convergence rate.

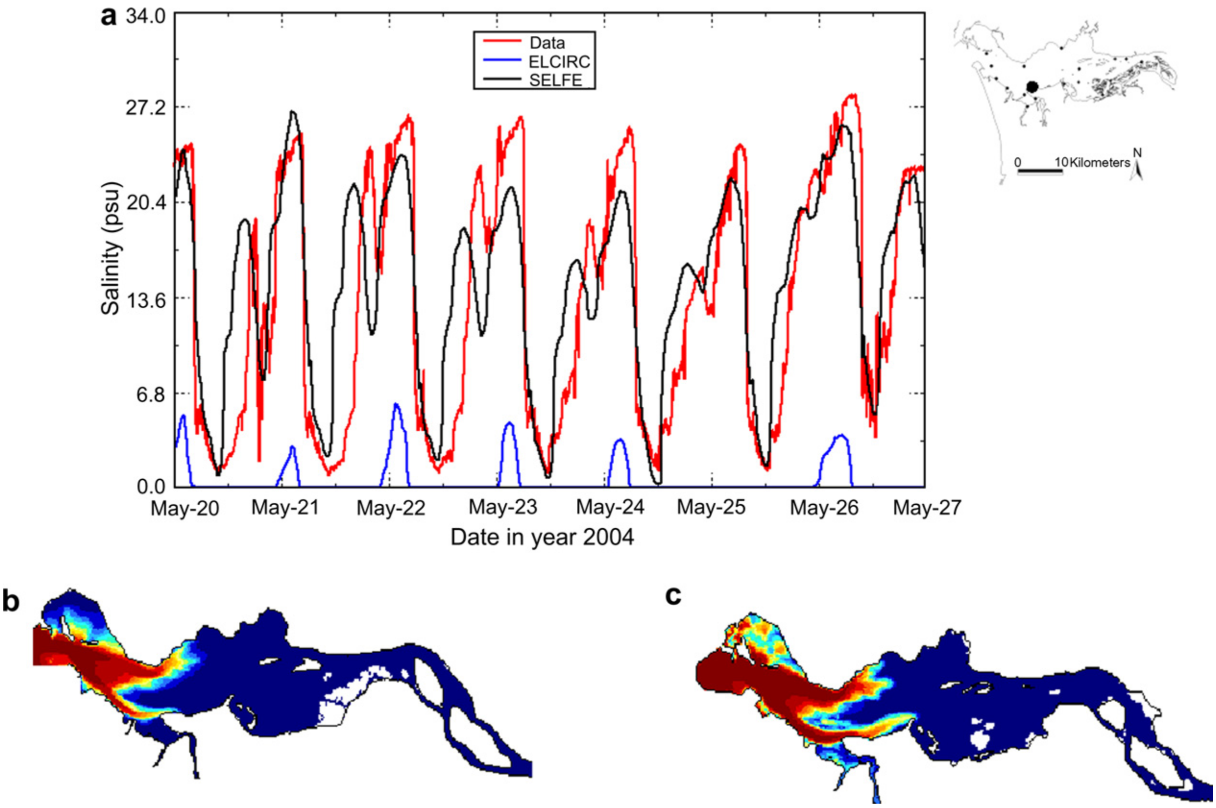


Fig. 1. The figure compares (a) week-long time histories of salinity at a bottom sensor at a real-time observation (am169; see insert) in the Columbia River estuary. It also contrasts maximum salinity penetration at the bottom of the estuary computed by (b) ELCIRC and (c) SELFE. During freshet periods, such as the week shown, salinity at am169 is a severe benchmark for model performance, because of its relative proximity to upper limit of salinity intrusion. For the period shown, and as a general tendency throughout the year, SELFE outperforms ELCIRC in the ability to simulate salinities at am169, as well as extent of salinity intrusion (not shown).

partially illustrated in Figs. 1 and 2, the improvements introduced in SELFE have important consequences: in general, SELFE describes complex ocean circulation features more accurately than does ELCIRC, with the differences being large enough to be relevant for scientific understanding and for management and operation of the Columbia River.

We attribute the superior performance of SELFE to overcoming the ELCIRC limitations described below:

1. ELCIRC uses constant shape functions to solve for elevations from the depth-integrated continuity equation, a natural choice given the model's finite-volume framework. In addition to the obvious drawback of low accuracy, this strategy also causes difficulty in evaluating first derivatives, a crucial step in correctly treating the Coriolis term (Zhang et al., 2004).
2. The orthogonality requirement, which originates from the finite-difference framework used in UnTRIM-like models, is very restrictive for the design and construction of unstructured grids. In practice, this requirement is often not strictly adhered to, resulting in degradation of accuracy. For example, we will demonstrate in Section 4.1.1 that no convergence is guaranteed for *non-orthogonal* grids (see Fig. 6a).
3. ELCIRC uses Z coordinates in the vertical, which introduces a staircase representation of the bottom and therefore fails to resolve the bottom boundary layer, with significant limitations for the representation of bottom-controlled estuarine processes (Zhang et al., 2004).

Although ELCIRC does not necessarily represent all UnTRIM-like models, the above limitations appear common among such models (e.g., see p. 335 and Fig. 1 of Casulli and Walters, 2000).

SELFE overcomes the first two limitations by using a formal Galerkin finite-element framework, supported at a minimum by linear shape functions. It partially addresses the third limitation by using hybrid SZ coordinates in the vertical direction. The flexibility afforded by the hybrid SZ coordinates is extremely important to enable a single model to correctly model estuary-plume systems, where depths change from O(1 m) to O(1000 m). Indeed, Z coordinates are necessary to properly represent thin surface plumes (Section 4.4), while terrain-following S coordinates are necessary to properly represent the bottom boundary layer and thus processes such as frictional losses and salinity intrusion into the estuary (e.g., Fig. 1).

This paper describes the physical and numerical formulations of SELFE (Sections 2 and 3) and compares the per-

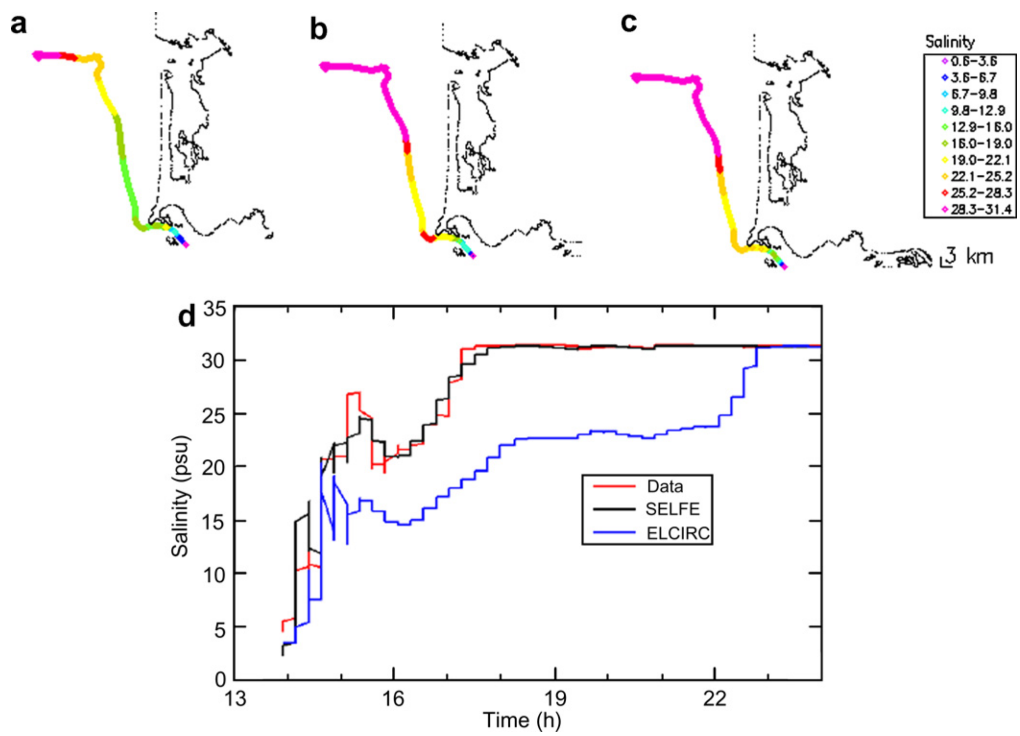


Fig. 2. The top three images show salinities along the path of the R/V Piky during a July 10, 2005 cruise in the Columbia River plume. Salinities are described by: (a) ELCIRC daily forecasts; (b) observations; and (c) SELFE daily forecasts. The bottom image shows contrasting time histories of the same information. SELFE forecasts clearly outperform ELCIRC forecasts for this specific period, and, although details do vary, also consistently do so during multiple cruises conducted by three different vessels since June 13, 2005. All times shown are Pacific Standard Time.

formance of SELFE and ELCIRC using a range of controlled synthetic benchmarks (Section 4). A description of the performance of SELFE in actual field applications is left to future publications.

2. Physical formulation of SELFE

SELFE solves the 3D shallow-water equations, with hydrostatic and Boussinesq approximations, and transport equations for salt and heat. The primary variables that SELFE solves are free-surface elevation, 3D velocity, 3D salinity, and 3D temperature of the water. In a Cartesian frame, the equations read:

$$\nabla \cdot \mathbf{u} + \frac{\partial w}{\partial z} = 0 \tag{1}$$

$$\frac{\partial \eta}{\partial t} + \nabla \cdot \int_{-h}^{\eta} \mathbf{u}\, \mathrm{d}z = 0 \tag{2}$$

$$\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t} = \mathbf{f} - g\nabla\eta + \frac{\partial}{\partial z}\left(\nu \frac{\partial \mathbf{u}}{\partial z}\right); \quad \mathbf{f} = -f\mathbf{k}\times\mathbf{u} + \alpha g \nabla\hat{\psi}$$
$$-\frac{1}{\rho_0}\nabla p_A - \frac{g}{\rho_0}\int_z^{\eta}\nabla\rho\, \mathrm{d}\zeta + \nabla\cdot(\mu\nabla\mathbf{u}) \tag{3}$$

$$\frac{\mathrm{D}S}{\mathrm{D}t} = \frac{\partial}{\partial z}\left(\kappa\frac{\partial S}{\partial z}\right) + F_s \tag{4}$$

$$\frac{\mathrm{D}T}{\mathrm{D}t} = \frac{\partial}{\partial z}\left(\kappa\frac{\partial T}{\partial z}\right) + \frac{\dot{Q}}{\rho_0 C_p} + F_h \tag{5}$$

where

- (x, y) horizontal Cartesian coordinates (m)
- z vertical coordinate, positive upward (m)
- ∇ $\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)$
- t time (s)
- $\eta(x, y, t)$ free-surface elevation (m)
- $h(x, y)$ bathymetric depth (m)
- $\mathbf{u}(x, y, z, t)$ horizontal velocity, with Cartesian components (u, v) (m s^{-1})
- w vertical velocity (m s^{-1})
- f Coriolis factor (s^{-1}) (Section 2.5)
- g acceleration of gravity (m s^{-2})
- $\hat{\psi}(\phi, \lambda)$ earth-tidal potential (m) (Section 2.5)
- α effective earth-elasticity factor
- $\rho(\mathbf{x}, t)$ water density; by default, reference value ρ_0 is set as 1025 kg m^{-3}
- $p_A(x, y, t)$ atmospheric pressure at the free surface (N m^{-2})
- S, T salinity and temperature of the water (practical salinity units (psu), °C)
- ν vertical eddy viscosity (m^2 s^{-1})
- μ horizontal eddy viscosity (m^2 s^{-1})
- κ vertical eddy diffusivity, for salt and heat (m^2 s^{-1})
- F_s, F_h horizontal diffusion for transport equations (neglected in SELFE)
- $\dot{Q}$ rate of absorption of solar radiation (W m^{-2})
- C_p specific heat of water (J kg^{-1} K^{-1})

The differential system Eqs. (1)–(5) are closed with: (a) the equation of state describing the water density as a function of salinity and temperature, (b) the definition of the tidal potential and Coriolis factor, (c) parameterizations for horizontal and vertical mixing, via turbulence closure equations, and (d) appropriate initial and boundary conditions. Details for (a) and (b) can be found in Zhang et al. (2004); (c) will be discussed in Sections 2.1 and 3.3.2, and (d) in Sections 2.2 and 3.2.

2.1. Turbulence closure model

SELFE uses the Generic Length Scale (GLS) turbulence closure of Umlauf and Burchard (2003), which has the advantage of encompassing most of the 2.5-equation closure models (k–ε(Rodi, 1984); k–ω (Wilcox, 1998); Mellor and Yamada, 1982). In this framework, the transport, production, and dissipation of the turbulent kinetic energy (K) and of a generic length-scale variable (ψ) are governed by:

$$\frac{DK}{Dt} = \frac{\partial}{\partial z}\left(\nu_k^{\psi}\frac{\partial K}{\partial z}\right) + \nu M^2 + \mu N^2 - \varepsilon, \tag{6}$$

$$\frac{D\psi}{Dt} = \frac{\partial}{\partial z}\left(\nu_{\psi}\frac{\partial \psi}{\partial z}\right) + \frac{\psi}{K}\left(c_{\psi 1}\nu M^2 + c_{\psi 3}\mu N^2 - c_{\psi 2}F_{\mathrm{w}}\varepsilon\right), \tag{7}$$

where ν_k^{ψ} and ν_{ψ} are vertical turbulent diffusivities, $c_{\psi 1}$, $c_{\psi 2}$ and $c_{\psi 3}$ are model-specific constants (Umlauf and Burchard, 2003; Zhang et al., 2004), F_{w} is a wall proximity function, M and N are shear and buoyancy frequencies, and ε is a dissipation rate. The generic length-scale is defined as

$$\psi = \left(c_{\mu}^{0}\right)^{p} K^{m}\ell^{n}, \tag{8}$$

where $c_{\mu}^{0} = 0.3^{1/2}$ and ℓ is the turbulence mixing length. The specific choices of the constants p, m and n lead to the different closure models mentioned above. Finally, vertical viscosities and diffusivities as appeared in Eqs. (3)–(5) are related to K, ℓ, and stability functions:

$$\begin{aligned} \nu &= \sqrt{2}s_m K^{1/2}\ell \\ \mu &= \sqrt{2}s_h K^{1/2}\ell \\ \nu_k^{\psi} &= \frac{\nu}{\sigma_k^{\psi}} \\ \nu_{\psi} &= \frac{\nu}{\sigma_{\psi}}, \end{aligned} \tag{9}$$

where the Schmidt numbers σ_k^{ψ} and σ_{ψ} are model-specific constants. The stability functions (s_m and s_h) are given by an Algebraic Stress Model (e.g.: Kantha and Clayson, 1994; Canuto et al., 2001; or Galperin et al., 1988).

At the free surface and at the bottom of rivers and oceans, the turbulent kinetic energy and the mixing length are specified as Direchlet boundary conditions:

$$K = \frac{1}{2}B_1^{2/3}|\boldsymbol{\tau}_{\mathrm{b}}|^2, \tag{10}$$

$$\ell = \kappa_0 d_{\mathrm{b}} \text{ or } \kappa_0 d_{\mathrm{s}}, \tag{11}$$

where $\boldsymbol{\tau}_{\mathrm{b}}$ is a bottom frictional stress (Eq. (14)), $\kappa_0 = 0.4$ is the von Karman's constant, B_1 is a constant, and d_{b} and d_{s} are the distances to the bottom and the free surface, respectively.

2.2. Vertical boundary conditions for the momentum equation

The vertical boundary conditions for the momentum equation – especially the bottom boundary condition – play an important role in the SELFE numerical formulation, as it involves the unknown velocity (see Section 3). In fact, as a crucial step in solving the differential system, SELFE uses the bottom boundary condition to decouple the free-surface Eq. (2) from the momentum Eq. (3).

At the sea surface, SELFE enforces the balance between the internal Reynolds stress and the applied shear stress:

$$\nu\frac{\partial \mathbf{u}}{\partial z} = \boldsymbol{\tau}_{\mathrm{w}}, \quad \text{at } z = \eta \tag{12}$$

where the stress $\boldsymbol{\tau}_{\mathrm{w}}$ can be parameterized using the approach of Zeng et al. (1998) or the simpler approach of Pond and Pickard (1998).

Because the bottom boundary layer is usually not well resolved in ocean models, the no-slip condition at the sea or river bottom ($\mathbf{u} = w = 0$) is replaced by a balance between the internal Reynolds stress and the bottom frictional stress,

$$\nu\frac{\partial \mathbf{u}}{\partial z} = \boldsymbol{\tau}_{\mathrm{b}}, \quad \text{at } z = -h. \tag{13}$$

The specific form of the bottom stress $\boldsymbol{\tau}_{\mathrm{b}}$ depends on the type of boundary layer used. While the numerical method for SELFE as outlined in Section 3 can be applied to other types of bottom boundary layer (e.g., laminar boundary layer), we will only discuss the turbulent boundary layer below (Blumberg and Mellor, 1987), given its prevalent usage in ocean modeling. The bottom stress in Eq. (13) is then:

$$\boldsymbol{\tau}_{\mathrm{b}} = C_{\mathrm{D}}|\mathbf{u}_{\mathrm{b}}|\mathbf{u}_{\mathrm{b}}. \tag{14}$$

The velocity profile in the interior of the bottom boundary layer obeys the logarithmic law:

$$\mathbf{u} = \frac{\ln[(z+h)/z_0]}{\ln(\delta_{\mathrm{b}}/z_0)}\mathbf{u}_{\mathrm{b}}, \quad (z_0 - h \leqslant z \leqslant \delta_{\mathrm{b}} - h), \tag{15}$$

which is smoothly matched to the exterior flow at the top of the boundary layer. In Eq. (15), δ_{b} is the thickness of the bottom computational cell (assuming that the bottom is sufficiently resolved in SELFE that the bottom cell is inside the boundary layer), z_0 is the bottom roughness, and $\mathbf{u}_{\mathrm{b}}$ is the velocity measured at the top of the bottom computational cell. Therefore the Reynolds stress inside the boundary layer is derived from Eq. (15) as

$$\nu\frac{\partial \mathbf{u}}{\partial z} = \frac{\nu}{(z+h)\ln(\delta_{\mathrm{b}}/z_0)}\mathbf{u}_{\mathrm{b}}. \tag{16}$$

Utilizing the turbulence closure theory discussed in Section 2.1, the eddy viscosity can be found from Eq. (9), with the stability function, the turbulent kinetic energy, and the mesoscale mixing length given by:

$$
\begin{aligned}
&s_m = g_2,\\
&K = \frac{1}{2}B_1^{2/3}C_{\mathrm{D}}|\mathbf{u}_{\mathrm{b}}|^2 \qquad (17)\\
&\ell = \kappa_0(z+h),
\end{aligned}
$$

where g_2 and B_1 are constants with $g_2B_1^{1/3} = 1$. Therefore, the Reynolds stress is constant inside the boundary layer:

$$\nu\frac{\partial \mathbf{u}}{\partial z} = \frac{\kappa_0}{\ln(\delta_{\mathrm{b}}/z_0)}C_{\mathrm{D}}^{1/2}|\mathbf{u}_{\mathrm{b}}|\mathbf{u}_{\mathrm{b}}, \quad (z_0 - h \leqslant z \leqslant \delta_{\mathrm{b}} - h), \tag{18}$$

and the drag coefficient is calculated from Eqs. (13), (14), and (18) as

$$C_{\mathrm{D}} = \left(\frac{1}{\kappa_0}\ln\frac{\delta_{\mathrm{b}}}{z_0}\right)^{-2}, \tag{19}$$

which is the drag formula as discussed in Blumberg and Mellor (1987). Eq. (18) also shows that the vertical viscosity term in the momentum equation Eq. (3) vanishes inside the boundary layer. This fact will be utilized in the numerical model of SELFE in Section 3.

3. Numerical formulation of SELFE

Numerical efficiency and accuracy consideration dictates the numerical formulation of SELFE. SELFE solves the differential equation system described in Section 2 with finite-element and finite-volume schemes. No mode splitting is used in SELFE, thus eliminating the errors associated with the splitting between internal and external modes (Shchepetkin and McWilliams, 2005). Semi-implicit schemes are applied to all equations; the continuity and momentum equations (Eqs. (2) and (3)) are solved simultaneously, thus bypassing the most severe stability restrictions (e.g. CFL). A key step in SELFE is to decouple the continuity and momentum equations (Eqs. (2) and (3)) via the bottom boundary layer, as will be shown in Section 3.2. SELFE uses an Eulerian–Lagrangian method (ELM) to treat the advection in the momentum equation, thus further relaxing the numerical stability constraints. The advection terms in the transport equations (Eqs. (4) and (5)) are treated with either ELM or a finite-volume upwind method (FVUM), the latter being mass conservative.

3.1. Domain discretization

In SELFE, unstructured triangular grids are used in the horizontal direction, while hybrid vertical coordinates – partly terrain-following S coordinates and partly Z coordinates – are used in the vertical direction. The origin of the z-axis is at the undisturbed Mean Sea Level (MSL). The terrain-following S layers (Song and Haidvogel, 1994) are placed on top of a series of Z layers (Fig. 3a and b), with the demarcation line between S and Z layers located at level k^z ($z = -h_{\mathrm{s}}$). That is to say, the vertical grid is allowed to follow the terrain up to a maximum depth of h_{s}. The free surface is at level N_z throughout the domain (for all wet points), but the bottom level indices, k^b, may vary in space due to the staircase representation of the bottom in Z layers. Note that $k^b \leqslant k^z$ and the equality occurs when the local depth $h \leqslant h_{\mathrm{s}}$. A "pure S" representation is a special case with $k^b = k^z = 1$ and h_{s} greater than the maximum depth in the domain, but a "pure Z" model is not a special case in SELFE. The details of the terrain-following coordinates used in SELFE can be found in Appendix A. The rationale for using such a hybrid coordinate system is discussed next.

The "pure S" representation of SELFE was initially chosen by the authors to avoid the staircase representation of the bottom and surface, and thus loss of accuracy commonly associated with the Z coordinates. While sufficient and preferable for some applications, the "pure S" SELFE suffers from the so-called hydrostatic inconsistency (Section 3.3.3) commonly associated with the terrain-following coordinate models, and fails in applications involving steep bathymetry and strong stratification, as found in freshwater plumes of largest rivers like Columbia River. As will be demonstrated in the benchmark test in Section 4.4, the inclusion of Z layers effectively alleviates the hydrostatic inconsistency and results in a physically more realistic plume. Therefore the hybrid vertical coordinate system has the benefits of both S and Z coordinates: the S layers used in the shallow region ($h \leqslant h_{\mathrm{s}}$) resolve the bottom efficiently and the Z layers, which are only used in the deep region with $h > h_{\mathrm{s}}$, fend off the hydrostatic inconsistency. The effects of the staircase representation of the bottom are arguably small in the deep region because the velocities there are small; the effects can also be minimized by choosing the largest possible value for h_{s} for a given application.

The use of a hybrid vertical coordinate system raises the issue of in which coordinate system the equations should be solved. We solve all equations in their original forms in the untransformed Z coordinates and use the transformation in Eq. (47) (in Appendix A) only to generate a vertical grid and to evaluate the horizontal derivatives (such as the horizontal viscosity). The main reason for not transforming the equations into S coordinates is that the transformation is degenerate under the special circumstances described in Appendix A (Eqs. (50) and (51)). Therefore the role of vertical coordinates is mostly hidden in SELFE; all equations but one (the integrated continuity equation) are solved along the vertical direction only, which can be done on any vertical grid (including, in theory, an unstructured grid). The liberal treatment of the vertical coordinates makes the implementation of the hybrid vertical coordinates (SZ) system easier. A similar approach was also used by Shchepetkin and McWilliams (2005), who solved the equations in the Z space despite the S coordinates being used in the vertical direction.

Strictly speaking, since the free surface is moving and so are the upper S levels (in the original Z space), all variables

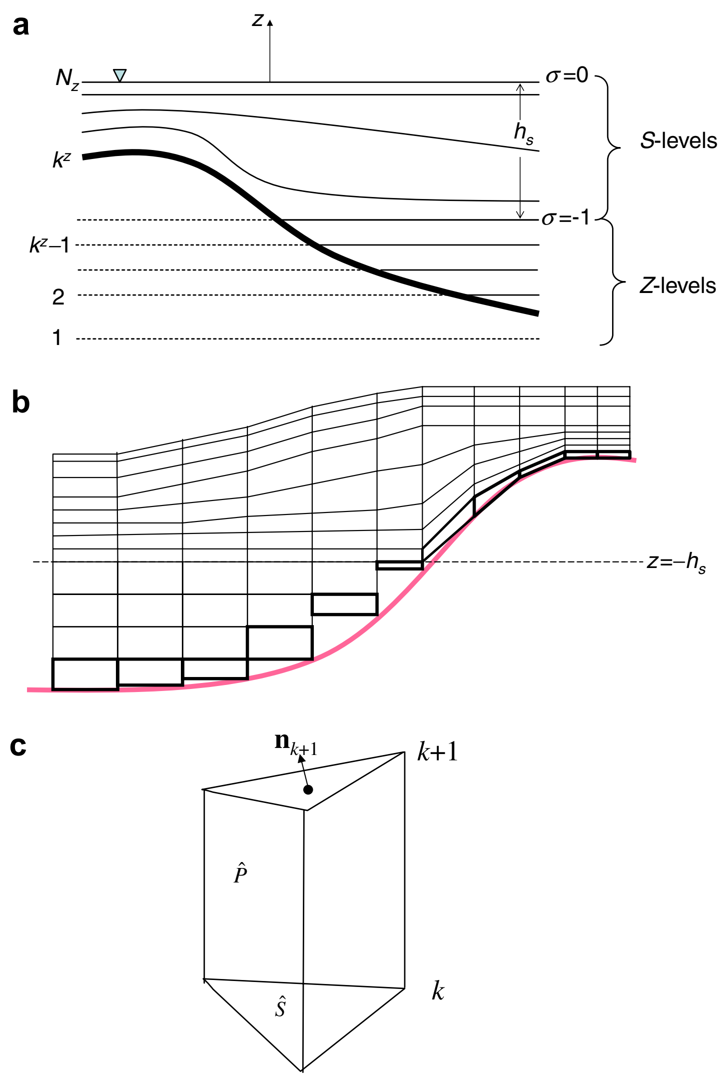


Fig. 3. Vertical grid and hybrid coordinate system. (a) Schematic view. S-levels are always on top of Z-levels. (b) Vertical transect view of the discretized domain. Bottom cells are highlighted. (c) Basic computational unit of a triangular prism, with uneven top and bottom surfaces.

need to be re-interpolated onto the new vertical grid after the levels are updated at the end of each time step. However, the effects of the movement of the S levels from one time step to the next are negligible, as long as the vertical movement of the free surface within a step is much smaller than the minimum layer thickness. This condition is easily satisfied in most practical applications; for example, in typical tidal-driven circulations, the maximum displacement of the free surface in a time step as large as 5 minutes is only a few centimeters or less, which is much smaller than a typical top layer thickness of a few meters or more. Therefore, we chose to skip this interpolation step in SELFE, as a linear interpolation would introduce additional numerical diffusion, and a higher-order interpolation would introduce numerical dispersion into the solution. Note that a similar omission also occurs in many Z coordinate models, where the top layers also change with time.

In many parts of SELFE, interpolation at an arbitrary location in 3D space is necessary; examples include the interpolation at the foot of the characteristic line (Section 3.3.1) and the conversion of velocity from element sides to nodes (Section 3.2). The horizontal interpolation is usually done on a fixed Z-plane (instead of along an S plane). One problem with this approach is the loss of accuracy near the bottom and the free surface (Fortunato and Baptista, 1996). Therefore in SELFE, the interpolation can be optionally done in the transformed S space in regions where no Z layers are used ("pure S region" with $h \leqslant h_s$). The latter approach is more accurate in shallow regions where rapid changes in bathymetry are common.

In the horizontal dimension, unstructured triangular grids are used, and the connectivity of the grid is defined as follows: the three sides of an element i are enumerated as $js(i,l)$ $(l = 1, 2, 3)$. The surrounding elements of a particular node i are enumerated as $ine(i,l)$ $(l = 1, \ldots, nne(i))$, where $nne(i)$ is the total number of elements in the "ball" of the node.

Please cite this article in press as: Zhang, Y., Baptista, A.M., SELFE: A semi-implicit Eulerian–Lagrangian finite-element model ..., Ocean Modell. (2008), doi:10.1016/j.ocemod.2007.11.005

After the domain is discretized horizontally and vertically, the basic 3D computational units of SELFE are triangular prisms (see Fig. 3c). In the original Z space, the prisms may not have level bottom and top surfaces. A staggering scheme is used to define variables. The surface elevations are defined at the nodes. The horizontal velocities are defined at the side centers and whole levels. The vertical velocities are defined at the element centers and whole levels as they are solved with a finite-volume method. The linear shape functions are used for elevations and velocities; we note, however, that for velocities, shape functions are only used for interpolation at the feet of characteristic lines (Section 3.3.1). Note that the shape functions used here are different from those in a lowest-order Raviart–Thomas element (Walters, 2005), in that the elevations are not constant within an element but continuous across elements. The locations where salinities and temperatures are defined depend on the method used to solve the transport equations; they are defined at the prism centers if the FVUM is used (Section 3.4), and at both nodes *and* side centers, at whole levels, if the ELM is used (Section 3.3.1).

3.2. Barotropic module

SELFE solves the barotropic Eqs. (1)–(3) first, as the transport and turbulent closure equations lag one time step behind (in other words, the baroclinic pressure gradient term in the momentum equation is treated explicitly in SELFE). The transport and turbulent closure equations will be discussed in Section 3.4. Due to the hydrostatic approximation, the vertical velocity w is solved from Eq. (1) after the horizontal velocity is found. To solve the coupled Eqs. (2) and (3), we first discretize them and the vertical boundary conditions Eqs. (12) and (13) semi-implicitly in time as:

$$\frac{\eta^{n+1}-\eta^n}{\Delta t} + \theta\nabla\cdot\int_{-h}^{\eta}\mathbf{u}^{n+1}\,dz + (1-\theta)\nabla\cdot\int_{-h}^{\eta}\mathbf{u}^n\,dz = 0 \tag{20}$$

$$\frac{\mathbf{u}^{n+1}-\mathbf{u}_*}{\Delta t} = \mathbf{f}^n - g\theta\nabla\eta^{n+1} - g(1-\theta)\nabla\eta^n + \frac{\partial}{\partial z}\left(\nu^n\frac{\partial\mathbf{u}^{n+1}}{\partial z}\right), \tag{21}$$

$$\begin{cases} \nu^n\frac{\partial\mathbf{u}^{n+1}}{\partial z} = \tau_w^{n+1}, & \text{at } z=\eta^n; \\ \nu^n\frac{\partial\mathbf{u}^{n+1}}{\partial z} = \chi^n\mathbf{u}_b^{n+1}, & \text{at } z=-h, \end{cases} \tag{22}$$

where superscripts denote the time step, $0 \leqslant \theta \leqslant 1$ is the implicitness factor, $\mathbf{u}_*(x,y,z,t^n)$ is the back-tracked value calculated with ELM (Section 3.3.1), and $\chi^n = C_D|\mathbf{u}_b^n|$. The elevations in the second and third terms of Eq. (20) are treated explicitly, which effectively amounts to a linearization procedure.

A Galerkin weighted residual statement in the weak form for Eq. (20) reads:

$$\int_\Omega \phi_i\frac{\eta^{n+1}-\eta^n}{\Delta t}\,d\Omega + \theta\left[-\int_\Omega\nabla\phi_i\cdot\mathbf{U}^{n+1}\,d\Omega + \int_{\Gamma_v}\phi_i\widehat{U}_n^{n+1}\,d\Gamma_v + \int_{\bar{\Gamma}_v}\phi_i U_n^{n+1}\,d\bar{\Gamma}_v\right] + (1-\theta)\left[-\int_\Omega\nabla\phi_i\cdot\mathbf{U}^n\,d\Omega + \int_\Gamma\phi_i U_n^n\,d\Gamma\right] = 0, \quad (i=1,\ldots,N_p) \tag{23}$$

where N_p is the total number of nodes, $\Gamma \equiv \Gamma_v + \overline{\Gamma}_v$ is the boundary of the entire domain Ω, with Γ_v corresponding to the boundary segments where natural boundary conditions are specified, $\mathbf{U} = \int_{-h}^{\eta}\mathbf{u}\,dz$ is the depth-integrated velocity, U_n is its normal component along the boundary, and $\widehat{U}_n$ is the boundary condition. In SELFE, linear shape functions are used; thus, ϕ_i are conventional "hat" functions.

Integrating the momentum Eq. (21) along the vertical direction leads to:

$$\mathbf{U}^{n+1} = \mathbf{G}^n - g\theta H^n\Delta t\nabla\eta^{n+1} - \chi^n\Delta t\mathbf{u}_b^{n+1} \tag{24}$$

with

$$\mathbf{G}^n = \mathbf{U}_* + \Delta t\left[\mathbf{F}^n + \tau_w^{n+1} - g(1-\theta)H^n\nabla\eta^n\right], \quad H^n = h+\eta^n, \quad \mathbf{F}^n = \int_{-h}^{\eta^n}\mathbf{f}\,dz, \quad \mathbf{U}_* = \int_{-h}^{\eta^n}\mathbf{u}_*\,dz \tag{25}$$

Note that Eq. (24) involves no vertical discretization as it is merely an analytical integration of Eq. (21).

To eliminate the unknown $\mathbf{u}_b^{n+1}$ in Eq. (24), we invoke the discretized momentum equation, as applied to the top of the bottom cell:

$$\frac{\mathbf{u}_b^{n+1}-\mathbf{u}_{*b}}{\Delta t} = \mathbf{f}_b^n - g\theta\nabla\eta^{n+1} - g(1-\theta)\nabla\eta^n + \frac{\partial}{\partial z}\left(\nu^n\frac{\partial\mathbf{u}^{n+1}}{\partial z}\right), \quad \text{at } z=\delta_b-h. \tag{26}$$

However, since the viscosity term vanishes inside the bottom boundary layer (Eq. (18)), the bottom velocity can be formally solved as:

$$\mathbf{u}_b^{n+1} = \hat{\mathbf{f}}_b^n - g\theta\Delta t\nabla\eta^{n+1}, \tag{27}$$

where

$$\hat{\mathbf{f}}_b^n = \mathbf{u}_{*b} + \mathbf{f}_b^n\Delta t - g\Delta t(1-\theta)\nabla\eta^n. \tag{28}$$

Note that although the vertical viscosity is not explicitly present in Eq. (27), it is indirectly involved through terms $\mathbf{u}_{*b}$ and the Coriolis term in $\mathbf{f}_b^n$. Substituting Eq. (27) into Eq. (24) results in:

$$\mathbf{U}^{n+1} = \widehat{\mathbf{G}}^n - g\theta\widehat{H}^n\Delta t\nabla\eta^{n+1}, \tag{29}$$

where

$$\widehat{\mathbf{G}}^n = \mathbf{G}^n - \chi^n\Delta t\hat{\mathbf{f}}_b^n, \quad \widehat{H}^n = H^n - \chi^n\Delta t. \tag{30}$$

It is interesting to note from Eq. (30) that the bottom friction reduces the total depth by an amount that is proportional to the drag coefficient and the bottom velocity.

For simplicity the Coriolis terms are treated explicitly in SELFE. It is well known that the explicit treatment of the Coriolis terms is stable but introduces damping (Wicker and Skamarock, 1998). SELFE could have instead been formulated to treat the Coriolis terms implicitly, in which case, the two components of $\mathbf{U}$ would become coupled in Eq. (29), but could still be solved simultaneously from this equation.

Since SELFE uses linear shape functions for the elevations, the two components of the horizontal velocity, u and v, are solved from the momentum equation independently from each other after the elevations are found. This approach has important implications as far as the Coliolis is concerned, and is different from that used in ELCIRC. As a matter of fact, special treatment must be made to find the tangential velocity components in UnTRIM-like models after the normal velocities are found, as discussed in Zhang et al. (2004) and Ham et al. (2005).

Finally, substitution of Eq. (29) into Eq. (23) leads to an equation for elevations alone:

$$\int_\Omega \left[\phi_i \eta^{n+1} + g\theta^2\Delta t^2 \widehat{H}^n \nabla\phi_i \cdot \nabla\eta^{n+1}\right] \mathrm{d}\Omega - g\theta^2\Delta t^2 \times \int_{\bar{\Gamma}_v} \phi_i \widehat{H}^n \frac{\partial \eta^{n+1}}{\partial n} \mathrm{d}\bar{\Gamma}_v + \theta\Delta t \int_{\Gamma_v} \phi_i \widehat{U}_n^{n+1} \mathrm{d}\Gamma_v = I^n \tag{31}$$

where I^n consists of some explicit terms:

$$I^n = \int_\Omega \left[\phi_i \eta^n + (1-\theta)\Delta t \nabla\phi_i \cdot \mathbf{U}^n + \theta\Delta t \nabla\phi_i \cdot \widehat{\mathbf{G}}^n\right] \mathrm{d}\Omega - (1-\theta)\Delta t \int_\Gamma \phi_i U_n^n \mathrm{d}\Gamma - \theta\Delta t \int_{\bar{\Gamma}_v} \phi_i \mathbf{n} \cdot \widehat{\mathbf{G}}^n \, \mathrm{d}\bar{\Gamma}_v \tag{32}$$

Following standard finite-element procedures, and using appropriate essential and natural boundary conditions, SELFE solves Eq. (31) to determine the elevations at all nodes. For example, the integrals on $\bar{\Gamma}_v$ need not be evaluated if the essential boundary conditions are imposed by eliminating corresponding rows and columns of the matrix. Natural boundary conditions are used to evaluate the integral on Γ_v on the left-hand side of Eq. (31). If a Flather-type radiation condition (Flather, 1987) needs to be applied, it can be done in the following fashion:

$$\widehat{\mathbf{U}}_n^{n+1} - \overline{U}_n = \sqrt{g/H}(\eta^{n+1} - \bar{\eta}), \tag{33}$$

where $\overline{U}_n$ and $\bar{\eta}$ are specified incoming current. The matrix resulting from Eq. (31) is sparse and symmetric. It is also positive-definite if a mild restriction is placed on the friction-reduced depth in the form of $\widehat{H}^n \geqslant 0$. Numerical experiments (not shown) indicated that even this restriction can be relaxed for many practical applications that include shallow areas. The matrix can be efficiently solved using a pre-conditioned Conjugate Gradient method (Casulli and Cattani, 1994).

After the elevations are found, SELFE solves the momentum Eq. (3) along each vertical column at side centers. A semi-implicit Galerkin finite-element method is used, with the pressure gradient and the vertical viscosity terms being treated implicitly, and other terms treated explicitly:

$$\int_{-h}^{\eta} \gamma_k \left[\mathbf{u} - \Delta t \frac{\partial}{\partial z}\left(\nu \frac{\partial \mathbf{u}}{\partial z}\right)\right]_{j,k}^{n+1} \mathrm{d}z = \int_{-h}^{\eta} \gamma_k \left\{\mathbf{u}_* + \Delta t\left[\mathbf{f}_{j,k}^n - g\theta\nabla\eta_j^{n+1} - g(1-\theta)\nabla\eta_j^n\right]\right\} \mathrm{d}z, \tag{34}$$

where $\gamma_k(z)$ is the hat function in the vertical dimension. The two terms that are treated implicitly would have imposed the most severe stability constraints. The explicit treatment of the baroclinic pressure gradient and the horizontal viscosity terms, however, does impose mild stability constraints (Section 3.5).

After the velocities at all sides are found, the velocity at a node, which is needed in ELM, is evaluated by a weighted average of all surrounding sides in its ball, aided by proper interpolation in the vertical. The procedure to average the velocities (or alternatively calculating the velocity at a node based on a least-square fit from all surrounding sides) introduces numerical diffusion of the same order as the ELM (see Section 3.3.1). This is because the velocities at nodes are not used anywhere else in the model except in ELM tracking and interpolation. As an alternative to the averaging procedure, the velocity at a node is computed within each element from the three sides using the linear shape function and is kept discontinuous between elements. This approach leads to parasitic oscillations, but a Shapiro filter (Shapiro, 1970) can be used to suppress the noise, with minimum distortion of physical features. Our preliminary results indicate that the filter approach induces less numerical diffusion.

The vertical velocity serves as a diagnostic variable for local volume conservation,[3] but is a physically important quantity, especially when a steep slope is present (Zhang et al., 2004). To solve the vertical velocity, we apply a finite-volume method to a typical prism, as depicted in Fig. 3c, assuming that w is constant within an element i, and obtain:

$$\widehat{S}_{k+1}(\bar{u}_{k+1}^{n+1} n_{k+1}^x + \bar{v}_{k+1}^{n+1} n_{k+1}^y + w_{i,k+1}^{n+1} n_{k+1}^z) - \widehat{S}_k(\bar{u}_k^{n+1} n_k^x + \bar{v}_k^{n+1} n_k^y + w_{i,k}^{n+1} n_k^z) + \sum_{m=1}^{3} \widehat{P}_{js(i,m)}(\hat{q}_{js(i,m),k}^{n+1} + \hat{q}_{js(i,m),k+1}^{n+1})/2 = 0, \quad (k = k^b, \ldots, N_z - 1) \tag{35}$$

where $\widehat{S}$ and $\widehat{P}$ are the areas of the five prism surfaces (Fig. 3c), (n^x, n^y, n^z), are the normal vector (pointing upward), $\bar{u}$ and $\bar{v}$ the averaged horizontal velocities at the top and bottom surfaces, and $\hat{q}$ is the outward normal velocity at each side center. The vertical velocity is then

[3] Although other definitions of volume/mass exist, we define volume/mass in the finite-volume sense throughout this paper and measure conservation based on this definition.

solved from the bottom to the surface, in conjunction with the bottom boundary condition $(u, v, w) \cdot \mathbf{n} = 0$. The closure error between the calculated w at the free surface and the surface kinematic boundary condition is an indication of the local volume conservation error (Luettich et al., 2002). Because the primitive form of the continuity equation is solved in the model, this closure error is in general negligible.

As in UnTRIM-like models, one of the prominent features of SELFE is its natural treatment of wetting and drying in shallow areas. A node is considered wet (or dry) whenever the total depth H at the node is above (or below) the specified minimum depth h_0. With an appropriate wetting and drying algorithm, SELFE has been rigorously benchmarked against analytical solutions for wave run-up on a beach, and successfully applied to coastal inundation by tsunamis ~~(Zhang et al., in preparation)~~.

3.3. Treatment of explicit terms

3.3.1. Advection

SELFE treats the advection in the momentum and transport equations with ELM. In this method, a fluid particle is tracked from its position at step $n+1$ backwards in time and in 3D space to find its originating position at step n, followed by an interpolation at the foot of the characteristic line to evaluate the variable of interest. In SELFE, backtracking is done using either an Euler scheme or a more accurate, but expensive, fifth-order embedded Runge–Kutta scheme (Press et al., 1992); the latter method is needed for problems that pose special challenges for volume conservation, since tracking errors affect the volume conservation (Oliveira and Baptista, 1998).

The order of interpolation in ELM determines whether the leading-order truncation error is diffusion- or dispersion-dominant (Oliveira and Baptista, 1995). For velocity, linear interpolation is used,[4] i.e., the interpolation is done using the velocity information at the three nodes of the element that contains the foot of the characteristic line. In one dimension, this ELM introduces a diffusion-like leading-order truncation error in the following form (Baptista et al., 2005; Casulli and Cattani, 1994):

$$\begin{aligned}\varepsilon_1 &= \frac{v'_*}{2\Delta t}(x_{i+1} - x_*)(x_* - x_i)\\ &= \frac{\Delta x^2}{2\Delta t} v'_*[Cu](1 - [Cu])|\varepsilon_1| \leqslant \frac{\Delta x^2}{8\Delta t}|v''_*|, \end{aligned} \tag{36}$$

where v is the analytical solution for velocity, $x = x^*$ is the location of the foot of the characteristic line, $[x_{i+1}, x_{i+1}]$ is the interval encompassing the foot, and $[C_u]$ is the fractional part of the Courant number $C_u = v\Delta t/\Delta x$. Since the velocities at the two nodes of the interval are not exact but have errors from the averaging procedure described in Section 3.2, an additional truncation error occurs, the leading-order term of which is

$$\varepsilon_2 = \frac{\Delta x^2}{8\Delta t} v''_*. \tag{37}$$

Therefore the two errors ε_1 and ε_2 are of the same order of magnitude.

We note from Eq. (36) that the numerical diffusion is controlled by the Courant number, and is null when $[C_u] = 0$. As will be demonstrated in Section 4.1.1 (also in Zhang et al., 2004), numerical accuracy is the best when $C_u \geqslant 1$. Should the numerical diffusion become excessive due to a small Courant number, a larger time step or smaller grid size needs to be used. The numerical diffusion can also be reduced by using the discontinuous nodal velocity for ELM in conjunction with a Shapiro filter, as mentioned in Section 3.2.

The advection in the transport equations can also be treated using ELM. In this case, the linear interpolation often is excessively diffusive and therefore the quadratic interpolation or an element-splitting procedure, like the one suggested in Zhang et al. (2004), is used to reduce the numerical diffusion. The latter amounts to using a finer grid for the transport equations. The quadratic interpolation can be achieved by using the quadratic triangular element (Lapidus and Pinder, 1982, p. 116), and the quadratic function in the vertical direction, with an upwind bias.

3.3.2. Horizontal viscosity

Many circulation models rely on explicitly specified horizontal viscosity/diffusivity to eliminate the spurious oscillations; others (e.g., Marshall et al., 1997) use filters to suppress the sub-grid noise. Diffusion, either explicitly or implicitly implemented, achieves the same goal of eliminating the spurious oscillations.

The horizontal diffusion in the transport equations Eqs. (4) and (5) is neglected in SELFE, because the inherent numerical diffusion in ELM or FVUM is sufficient to suppress high-frequency spurious oscillations (Zhang et al., 2004). SELFE is also often run without horizontal viscosity because of the inherent numerical diffusion in ELM, or the effectiveness of Shapiro filter in eliminating numerical oscillations.

When needed, the horizontal viscosity can be calculated in the following fashion. The corresponding part in I^n is

$$\begin{aligned}&\int_\Omega \nabla\phi_i \cdot \widehat{\mathbf{G}}'\,\mathrm{d}\Omega = \Delta t \nabla\phi_i\\ &\quad\cdot \int_\Omega \left[\int_{-h}^{\eta} \nabla\cdot(\mu\nabla\mathbf{u})\,\mathrm{d}z - \chi\Delta t\nabla\cdot(\mu\nabla\mathbf{u}_\mathrm{b})\right]\mathrm{d}\Omega = \Delta t\nabla\phi_i\\ &\quad\cdot\left[\int_S \mu\mathbf{n}\cdot\nabla\mathbf{u}\,\mathrm{d}S - \bar{\chi}\Delta t\int_{S'}\mu\mathbf{n}\cdot\nabla\mathbf{u}_\mathrm{b}\,\mathrm{d}S'\right]\end{aligned} \tag{38}$$

where all terms are evaluated at time step n, $\bar{\chi}$ denotes the average of χ, S' is the boundary of the ball of node i, and S

[4] Alternative higher-order interpolation scheme based on Kriging (Le Roux et al., 1997) has also been implemented. This scheme is only marginally more expensive as the Kriging matrix depends on geometry only and needs to be inverted once, but sometimes offers significant improvement in accuracy.

is the exterior surface of the volume spanned by the ball from the bottom to the surface. To evaluate the gradient $\nabla\mathbf{u}$, which is defined in the original Z plane, the chain rule is used. For example:

$$\left.\frac{\partial\mathbf{u}}{\partial x}\right|_z = \left.\frac{\partial\mathbf{u}}{\partial x}\right|_\sigma - \frac{\partial\mathbf{u}}{\partial z}\frac{\partial z}{\partial x}. \tag{39}$$

The horizontal viscosity, μ, is conventionally specified as a constant or calculated from Smagorinsky parameterization (Smagorinsky, 1963). However, since the leading-order truncation error in the advection is given by Eq. (36), SELFE uses the following alternative parameterization for μ:

$$\mu = \gamma\frac{A}{\Delta t}, \tag{40}$$

where A is the local element area, and γ is a dimensionless constant. For stability reasons, $\gamma \leqslant 0.5$ (see Eq. (45)).

3.3.3. Baroclinic pressure gradient

All circulation models using terrain-following coordinates suffer from the so-called hydrostatic inconsistency, which stems from the fact that terrain-following coordinates do not conform to the geo-potentials (Gary, 1973; Blumberg and Mellor, 1987; Haney, 1991; Shchepetkin and McWilliams, 2003). As a result, the baroclinic pressure gradient is evaluated as the difference between two large components that tend to cancel each other, leading to large round-off errors. Hydrostatic inconsistency can also be viewed as the result of evaluating pressures at a grid point effectively using extrapolation when steep bathymetric slope is present (Shchepetkin and McWilliams, 2003). Many remedies have been proposed to mitigate this problem, including evaluating the pressure gradient in the Z coordinate (Fortunato and Baptista, 1996) or using higher-order schemes (Song and Haidvogel, 1994; Shchepetkin and McWilliams, 2003). While such remedies seemed successful in dealing with a typical benchmark test of a tall and isolated seamount in a stably stratified fluid (with a density profile $\rho(z)$) with zero viscosity and diffusivity (Song and Haidvogel, 1994), there are 2 limitations to this idealized test: (1) the test sheds no light on realistic situations where mixing is present, and (2) more importantly, the density profile used in the test violates the bottom boundary condition $\partial\rho/\partial z = 0$, except in the trivial case of density reaching a constant value below the shallowest depth (which is not what was used in the test). Therefore, theoretically, the analytical solution of a perpetually motionless state assumed in such a test is dubious at best. In any case, although it may be somewhat mitigated by using Z coordinate or higher-order integration schemes, the hydrostatic inconsistency cannot be completely eliminated in a terrain-following coordinate model (Pietrzak et al., 2002).

In SELFE, the use of a hybrid coordinate system in the vertical direction effectively alleviates hydrostatic inconsistency because the Z levels used in the deeper part of the vertical grid do conform to geo-potentials and therefore at least a part of the baroclinic pressure gradient calculation is not subject to the hydrostatic inconsistency. For the upper part of the water column, where the S coordinates are used, SELFE computes the gradient either (1) in density Jacobian form (with the vertical density gradients calculated using a cubic-spline fit), or (2) in Z space (with extra attention paid to the bottom and the surface, to ensure the use of interpolation instead of extrapolation; see Fortunato and Baptista, 1996). Together with higher-order integration schemes as suggested by Song and Haidvogel (1994), the two approaches generally yield comparable results.

3.4. Baroclinic module

The core part of SELFE is the barotropic module as described in Sections 3.2 and 3.3. To complete the model, SELFE solves two more sets of equations: transport and turbulence closure equations.

The advection in the transport equations is usually a dominant process. SELFE treats the advection in the transport equations with either an ELM or FVUM. If the ELM is used, the transport equations are solved at nodes and side centers along each vertical column using a finite-element method, with the lumping of the mass matrix to minimize numerical dispersion (in the form of under- or overshoots). As discussed in Section 3.3.1, the order of interpolation used in ELM is important since linear interpolation leads to excessive numerical diffusion. To reduce the numerical diffusion, element-splitting or quadratic interpolation is used in ELM (Zhang et al., 2004).

Despite its efficiency, one of the main drawbacks of the ELM approach is its disregard for mass conservation (Oliveira and Baptista, 1998). On the other hand, FVUM guarantees mass conservation. In FVUM, the scalar variables (salinity or temperature) are defined at the center of a prism, (i,k), which has five exterior faces (top and bottom with areas $\widehat{S}_{i,k}$ and $\widehat{S}_{i,k-1}$, and three vertical faces with areas $\widehat{P}_{jsj,k}$; see Eq. (35) and Fig. 3c). The discretized temperature equation reads:

$$\begin{aligned}
&T_{i,k}^{n+1}V_{i,k}^n + \Delta t(u_n)_{i,k}^{n+1}\widehat{S}_{i,k}T_{up(i,k)}^{n+1} + \Delta t(u_n)_{i,k-1}^{n+1}\hat{S}_{i,k-1}T_{up(i,k-1)}^{n+1} \\
&\quad = \Delta t A_i\left[\kappa_{i,k}^n\frac{T_{i,k+1}^{n+1}-T_{i,k}^{n+1}}{\Delta z_{i,k+1/2}^n} - \kappa_{i,k-1}^n\frac{T_{i,k}^{n+1}-T_{i,k-1}^{n+1}}{\Delta z_{i,k-1/2}^n}\right] \\
&\quad + V_{i,k}^n\left(T_{i,k}^n + \frac{\dot{Q}}{\rho_0 C_p}\Delta t\right) - \Delta t\sum_{l=1}^{3} q_l^{n+1}T_{up(jsj,k)}^n, \\
&\quad (k = k^b+1,\ldots,N_z),
\end{aligned} \tag{41}$$

where "up()" indicates upwinding, $V_{i,k}$ is the volume of the prism, u_n is the outward normal velocity, $jsj = js(i,l)$ are three sides, and $q_l^{n+1} = \widehat{P}_{jsj,k}(u_n)_{jsj,k}^{n+1}$ are 3 horizontal advective fluxes. The salinity equation is similarly discretized. Note that Eq. (41) reduces to Eq. (35) when $T = \text{const.}$ and $\dot{Q} = 0$.

The stability condition for the upwind scheme, the Courant number restriction, is given by:

$$\Delta t \leqslant \frac{V_{i,k}}{\sum_{j\in S^+}|q_j|}, \tag{42}$$

where S^+ indicates all outflow *horizontal* faces. Were the vertical advective fluxes on the left-hand side of Eq. (41) treated explicitly, the denominator in Eq. (42) would include the outflow faces for the top and bottom faces as well (Sweby, 1984; Casulli and Zanolli, 2005). But since the advective fluxes at the top and bottom faces are treated implicitly, S^+ excludes the top and bottom faces, and thus the more stringent stability constraints associated with the vertical advective fluxes are by-passed. The Courant number restriction (Eq. (42)) may still be too severe, and in this case the sub-division of a time step is necessary. Despite the fact that Eq. (41) does not conform to the depth-integrated continuity Eq. (31), the FVUM guarantees mass conservation and the maximum principle (i.e., the solution is bounded by the maximum and minimum of the initial and boundary conditions; Casulli and Zanolli, 2005), and thus is usually preferred over the ELM approach. To further reduce the numerical diffusion, we have recently implemented a higher-order finite-volume TVD scheme in SELFE (Sweby, 1984).

SELFE solves the turbulence closure equations (Eqs. (6) and (7)) along each vertical column at each node with a finite-element method. The vertical mixing terms and the dissipation term in these equations are treated implicitly, but the production and buoyancy terms are treated either implicitly or explicitly, depending on the sign of their total contribution (Zhang et al., 2004). The advection terms in the turbulence closure equations are small compared to other terms, and are therefore neglected in SELFE.

3.5. Numerical stability

Assuming a uniform grid and constant coefficients, a stability analysis of SELFE closely follows that in Casulli and Cattani (1994), because of similar matrix structures shared between the two models. It can be shown that SELFE is stable for $1/2 \leqslant \theta \leqslant 1$, with the highest degree of accuracy achieved at $\theta = 1/2$ (Casulli and Cattani, 1994). The explicit treatment of the baroclinic and horizontal viscosity terms does impose stability constraints for the time step and grid size. A stability condition for the baroclinic term in SELFE is given by (Zhang et al., 2004):

$$\frac{\Delta t\sqrt{g'h}}{\Delta_{xy}} \leqslant 1, \tag{43}$$

where

$$g' = g\frac{\Delta\rho}{\rho_0} \tag{44}$$

is the reduced gravity due to stratification. A stability condition for the horizontal viscosity term is tied to the local diffusion number (Casulli and Cheng, 1992):

$$\frac{\mu\Delta t}{\Delta_{xy}^2} \leqslant \frac{1}{2}. \tag{45}$$

Note that the constraints Eqs. (43) and (45) are much milder than the CFL condition. In particular, the condition in Eq. (45) no longer applies when no horizontal viscosity is used ($\mu = 0$). Also the internal wave speed as appeared in Eq. (43) is at least an order of magnitude smaller than the surface wave speed. Therefore exceptionally large time steps can be used in SELFE; for example, a time step of 90s was used in forecasting the Columbia River estuary and plume with a grid size as small as 80 m.

As indicated in Section 3.4, an additional stability condition arises if the FVUM is used to solve the transport equations (cf. Eq. (42)). Therefore a smaller time step is usually used to solve the transport equations.

4. Numerical benchmarks

In this section, we present results from SELFE for five benchmark tests of increasing complexity, and compare the results to those from ELCIRC to show the improvements over ELCIRC.

4.1. Wave run-up on a quarter-annulus domain

4.1.1. One-dimensional convergence test

We first test the barotropic module of SELFE and conduct a convergence study with a simple problem that has a linearized analytical solution (Lynch and Gray, 1978). In this test, the domain is a quarter annulus with a linear bottom slope and a depth varying from 25.05 m at the outer boundary to 10.02 m at the inner boundary (Fig. 4). Note that the problem is essentially 1D in the sense that the analytical solution varies in the radial direction only. An M_2 tide of 0.3048 m is imposed at the outer boundary, and all the other boundaries are closed. The bottom is frictionless, and the vertical viscosity and Coriolis force are not included in the test. Although negligible once the steady state is reached, non-linearity was found to be important in setting up the wave motion from rest, and was therefore retained in the simulation. Since the ELM formulation requires an inflow condition at the open boundary, the analytical velocity is imposed there.

Two S layers were used in SELFE runs, and one Z layer was used in ELCIRC runs. However, results were found to be insensitive to the choices of vertical grid, and differences between the ELCIRC and SELFE results shown below are attributed to two limitations of ELCIRC identified earlier (low-order shape functions; orthogonality requirements). A family of horizontal grids, all symmetric with respect to the 45° line (Fig. 4a), was used in the study. The implicitness factor was set at 0.6. Results from the last 4 days of the 7-

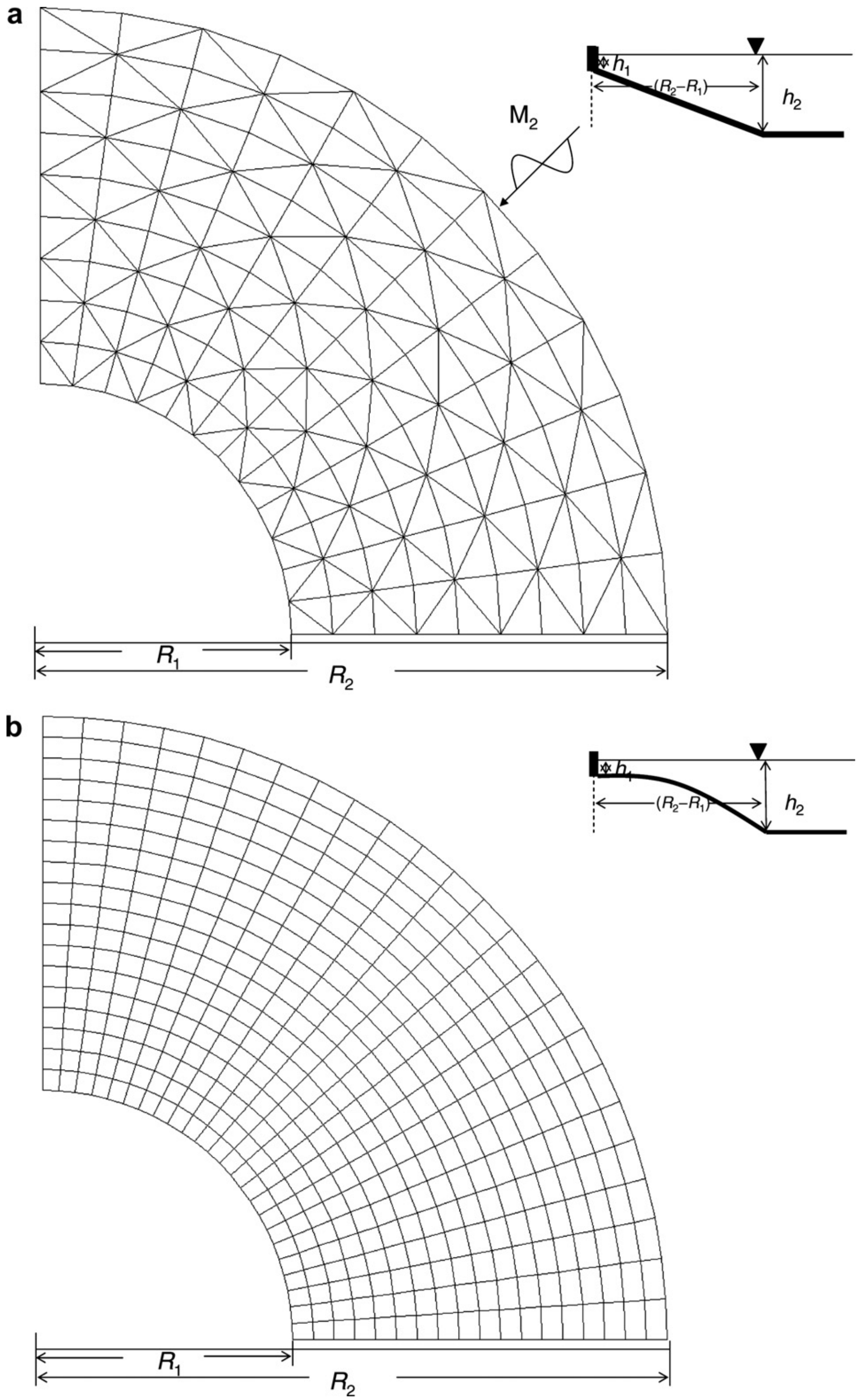


Fig. 4. (a) Quarter-annulus domain with a horizontal discretization and the bathymetry shown. The inner and outer radii are $R_1 = 60{,}960$ m, $R_2 = 152{,}400$ m. (b) An orthogonal grid generated by JANET used for the 3D test. The deviation from orthogonality is very small; for example, the maximum ratio of the distance between the two circumcenters of the two triangles split from each quadrangle, and the equivalent radius of each element is only about 9×10^{-6}.

day run were harmonically analyzed, and the averaged root-mean-square (RMS) errors for amplitudes and phases over the entire domain were examined for convergence with respect to the time step and grid size used in the test.

The convergence with respect to the time step (with the grid resolution in the radial direction being fixed at $\Delta r = 10160$ m) is typical of any ELM-based method including SELFE and ELCIRC (Fig. 5). The RMS errors actu-

Please cite this article in press as: Zhang, Y., Baptista, A.M., SELFE: A semi-implicit Eulerian–Lagrangian finite-element model ..., Ocean Modell. (2008), doi:10.1016/j.ocemod.2007.11.005

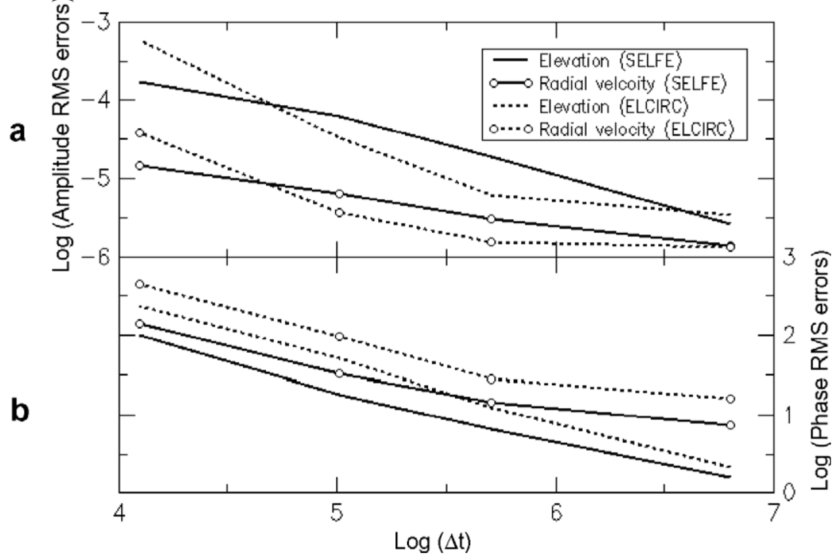


Fig. 5. RMS errors of (a) amplitudes and (b) phases for elevation and radial velocity as a function of time step used. No horizontal viscosity is used ($\gamma = 0$).

ally decrease with an increase in the time step because the reduced numerical diffusion in ELM at larger steps more than compensates for the increased truncation errors in time. For most time steps tested, SELFE has larger errors in amplitude while ELCIRC has larger errors in phase. ELCIRC also responds dramatically to the change in time step for $\Delta t < 150$ s.

Note that there is a non-linear feedback loop between the Lagrangian solution of advection and the solution of the continuity equation and the Eulerian part of the momentum equation, because velocities used in the tracking of characteristic lines are part of the solution. This explains why the errors in Fig. 5 do not strictly conform to Δt^{-1} as suggested by Eq. (36). Note also that the average Courant number, dominated by the surface wave speed in this test, is between 0.08 (for $\Delta t = 60$ s) and 1.25 (for $\Delta t = 900$ s). Hence, larger time steps (with the Courant number on the order of 1 or larger) can and *need to* be used in SELFE or ELCIRC to achieve better accuracy. This peculiar behavior of ELM does not invalidate the method itself since convergence is guaranteed as $\Delta t \rightarrow 0$, with the Courant number being fixed (in other words, with $\Delta x \rightarrow 0$ as well; Baptista, 1987).

Fig. 6 shows the convergence of the two models with respect to the grid resolution, the time step being fixed at $\Delta t = 300$ s. The grid is uniformly refined in both radial

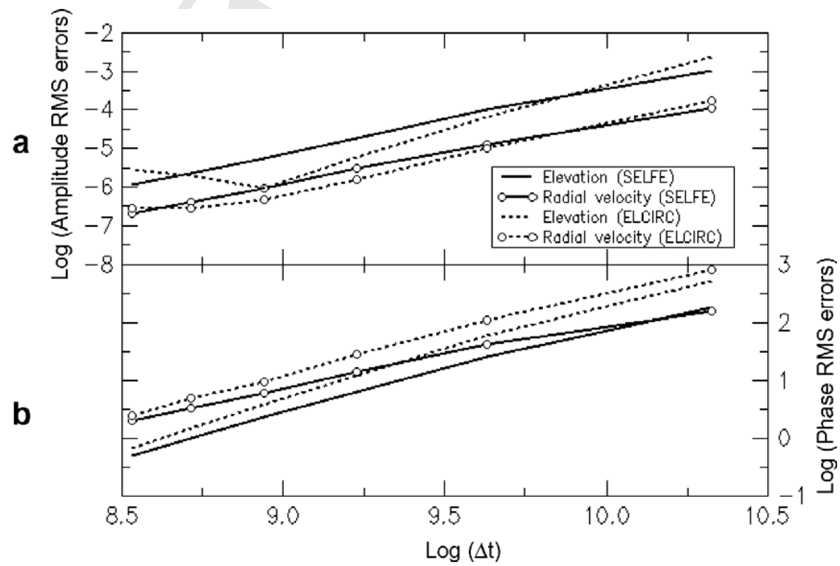


Fig. 6. RMS errors of (a) amplitudes and (b) phases for elevation and radial velocity as a function of grid size, with $\Delta t = 300$ s. No horizontal viscosity is used ($\gamma = 0$). Note the non-convergence of ELCIRC in (a). For SELFE, the linear regression slopes (the rate of convergence) are: 1.68 (elevation amplitude), 1.55 (radial velocity amplitude), 1.43 (elevation phase), and 1.36 (radial velocity phase).

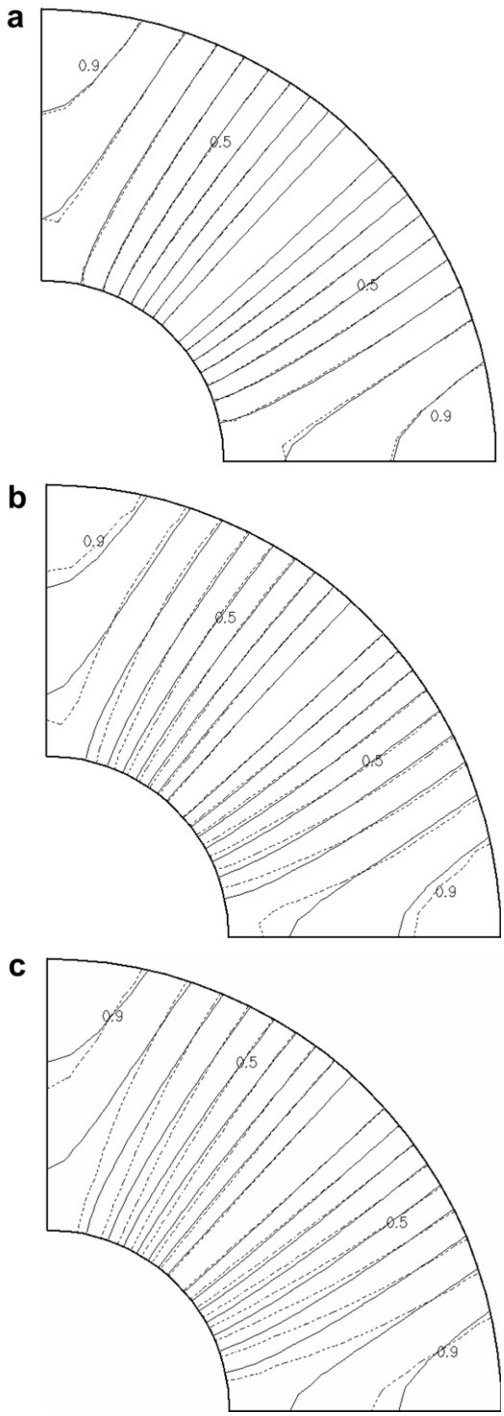


Fig. 7. 3D quarter-annulus test. (a) Isolines of amplitudes (normalized by 0.1 m) computed from analytical solution (solid lines) and SELFE (dashed lines). (b) Isolines of amplitudes (normalized by 0.1 m) computed from analytical solution (solid lines) and ELCIRC5.01 (dashed lines). (c) Isolines of amplitudes (normalized by 0.1 m) computed from analytical solution (solid lines) and ELCIRC5.01 with an orthogonal grid (see Fig. 4b) (dashed lines). The increment between adjacent isolines is 0.1. The mismatch on the outer boundary in (b) and (c) is due to the way elevation boundary condition is imposed in ELCIRC (see text). The average errors for (a), (b) and (c) are 0.003 and 0.013, 0.024.

and azimuthal directions, and so the aspect ratio of each element is kept unchanged. As $\Delta r \to 0$, the Courant number becomes very large, and so the adverse effect of the numerical diffusion on accuracy mentioned in the previous paragraph is not an issue here; convergence is expected. The convergence rate for SELFE is lower than second order for this family of non-uniform grids. The phase errors in SELFE are smaller than those in ELCIRC (Fig. 6b), but the rate of decrease in the amplitude errors is larger in ELCIRC than in SELFE, and SELFE has larger amplitude errors for $\Delta r < 17.5$ km (Fig. 6a). However, ELCIRC results for the amplitude clearly show divergence as $\Delta r \to 0$. The root cause for this divergence is that the family of grids used here is non-orthogonal. Due to the finite-difference framework used in UnTRIM-like models, grid orthogonality is a requirement for those models, which we confirm here for ELCIRC. SELFE, on the other hand, does not suffer from such a restriction and runs on general unstructured grids, a flexibility that is invaluable in practical applications.

4.1.2. Three-dimensional test

Lynch and Officer (1985) derived an analytical solution for a corresponding 3D case to the simple 1D problem shown in Section 4.1.1. In this sub-section this solution is used to quantitatively gauge the performance of ELCIRC and SELFE in a more complex setting than the 1D problem.

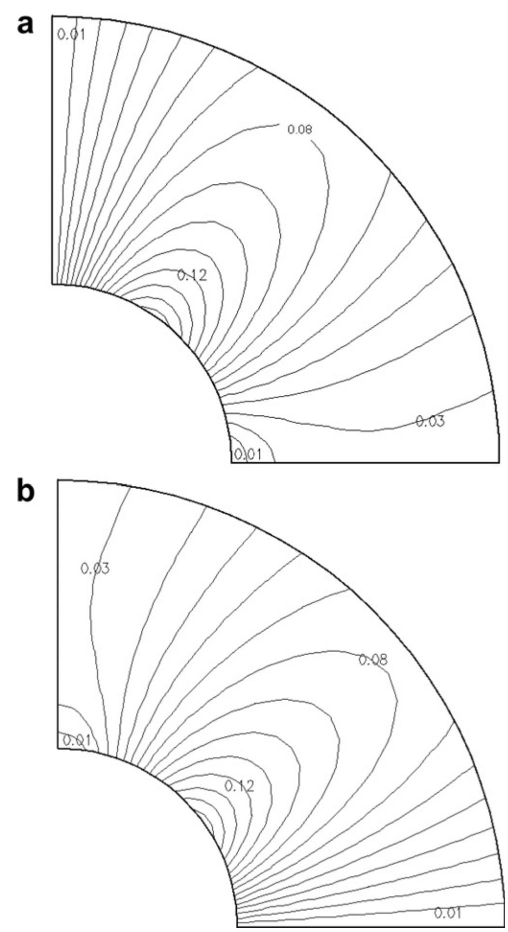


Fig. 8. Analytical solution of the amplitudes of surface (a) u and (b) v (in m/s).

In this test, we set the horizontal extent of the domain exactly the same as in the 1D case. The bottom depth changes quadratically along the radial direction, and the bottom is no-slip. The Coriolis and horizontal viscosity are both neglected, but the vertical viscosity is proportional to h^2. The choices of the bottom depth and the vertical viscosity are necessary for the analytical solution to exist. An M_2 tide with an amplitude of $0.1\cos(2\vartheta)$ is imposed at the outer boundary; therefore, there is a 180° phase change at $\vartheta = 45°$. Other details of the setup can be found in (Lynch and Werner, 1991).

For both models, an 18×24 symmetric grid was used in the horizontal direction. Eleven equally spaced σ levels were used in the SELFE simulation, while 20 equally spaced Z levels with 1 m resolution were used in the ELCIRC simulation. A large bottom drag coefficient of 1.0 was used to approximate the no-slip bottom. A time step of 2.5 min was used for both models, which were run for a total of 10 days, with a 2-day spin-up.

Fig. 7a and b shows a comparison of elevation amplitudes (normalized by 0.1 m appeared in the boundary condition) computed from a harmonic analysis of the

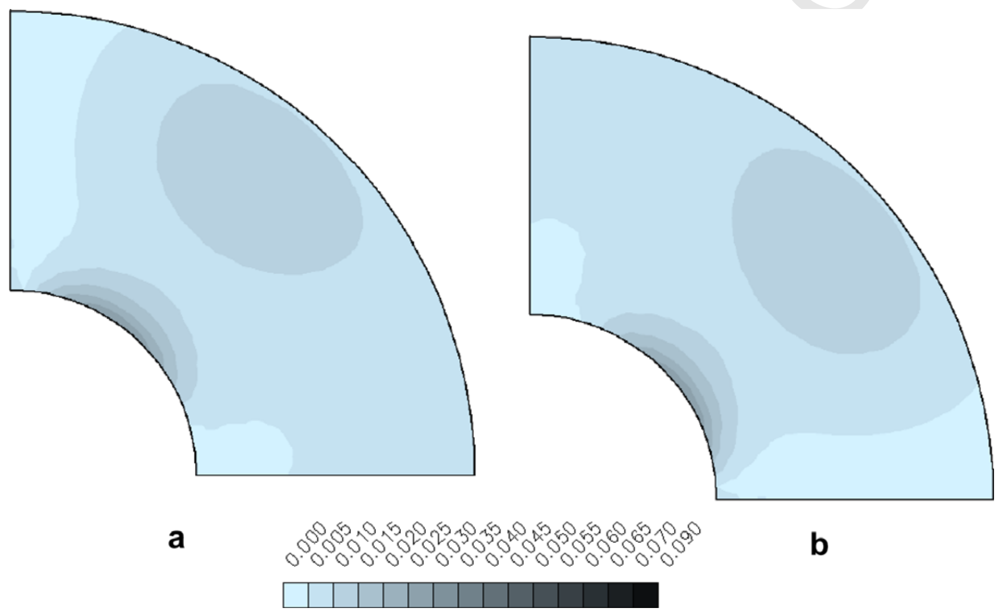


Fig. 9. Errors in amplitudes of surface (a) u and (b) v, calculated from SELFE. The average errors are 8 mm/s for both u and v.

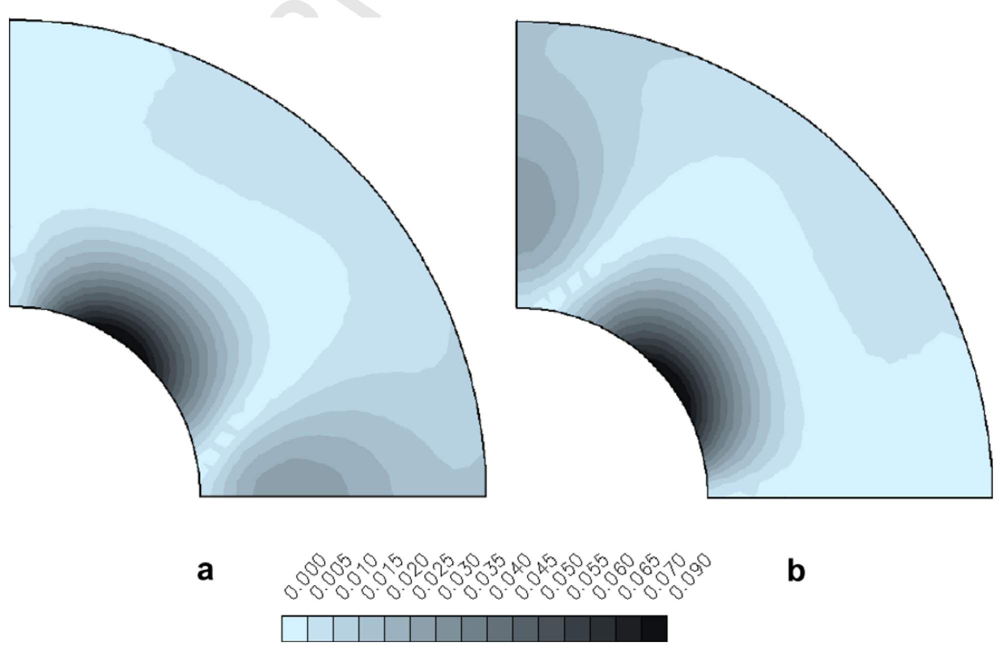


Fig. 10. Errors in amplitudes of surface (a) u and (b) v, calculated from ELCIRC. The average errors are 14 mm/s for both u and v.

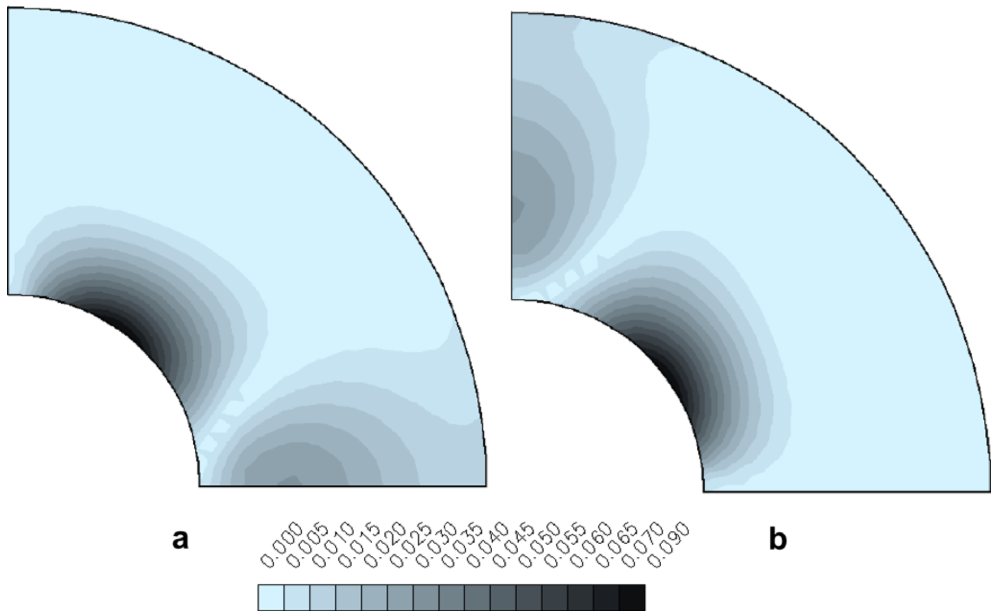


Fig. 11. Errors in amplitudes of surface (a) u and (b) v, calculated from ELCIRC on an orthogonal grid (Fig. 4b). The average errors are 12 mm/s for both u and v.

analytical and numerical solutions for the last 6 days. The mismatch between ELCIRC and analytical elevations at the outer boundary, especially in regions of larger amplitudes, is due to the staggering scheme used in ELCIRC; in ELCIRC the elevation boundary condition is imposed at element centers not nodes and the elevations at nodes are computed from elevations at element centers using an averaging scheme (Zhang et al., 2004). The averaging scheme has resulted in larger errors in this 3D case than the simple 1D case in Section 4.1.1, and therefore a finer grid seems to be needed in ELCIRC to overcome this problem. The SELFE solution is clearly more accurate than the ELCIRC solution, which is reflected in the normalized average errors in the entire domain (0.003 m vs. 0.013 m). The analytical amplitudes of the two horizontal components of the surface velocity are plotted in Fig. 8. The largest velocity occurs near the interior wall due to the strong traverse flow in the azimuthal direction. The errors in the SELFE and ELCIRC velocity amplitudes are shown in Figs. 9 and 10, respectively. Not surprisingly, the largest discrepancy for both models occurs in the region where the velocity is largest. In addition, the ELCIRC solution also has large errors on the two side walls (Fig. 10). The average error for either u and v in SELFE is 8 mm/s as compared to 14 mm/s in ELCIRC. Comparison of time series at various locations (not shown) indicate that SELFE accurately captures the main features of the velocity field at all tidal stages during the 10-day run, while ELCRC has large errors during the reversal stages.

To assess the role of non-orthogonality in the accuracy of ELCIRC results, we built an alternative grid with the commercial software JANET (from Smile Consulting, Germany). The grid, as shown in Fig. 4b, consists of quadrangles that are combined from the original triangles, and has a very small deviation from strict orthogonality (see Fig. 4b). With this new grid, while the velocity amplitudes are slightly improved from the previous ELCIRC results (Figs. 10 and 11), the elevation amplitudes are 85% worse (Fig. 7b and c). Therefore the inferior results obtained in ELCIRC can-not be attributed to orthogonality, and are more likely due to the low-order shape functions and/or the vertical coordinates.

4.2. Volume conservation test on a river segment

Local volume conservation has long been questioned for finite-element models[5] and indeed some early finite-element circulation models suffered from serious volume conservation problems due to the use of the Generalized Wave-Continuity Equation (GWCE) (ADCIRC; QUODDY). GWCE is a blend of continuity and momentum equations, used to fend off unphysical parasitic oscillations that originate from the non-staggering scheme used in the Finite-Element Method (Westerink et al., 2004). The choice of the relative weight between the two equations used in GWCE leads to either a loss in volume conservation or parasitic oscillation. Even with a relatively "primitive" form of GWCE (i.e., with a very small weight), significant volume imbalance can occur.

SELFE solves the primitive form of the continuity equation, and thus has a much improved conservation property relative to GWCE-based models, despite the fact that volume conservation is not enforced explicitly. Numerous

[5] We used the terms "volume/mass conservation" in the most traditional sense, i.e., change in volume/mass in a region should be accounted for by the boundary fluxes and internal sources and sinks. Definition of volume/mass in SELFE is given on ~~page 20~~.

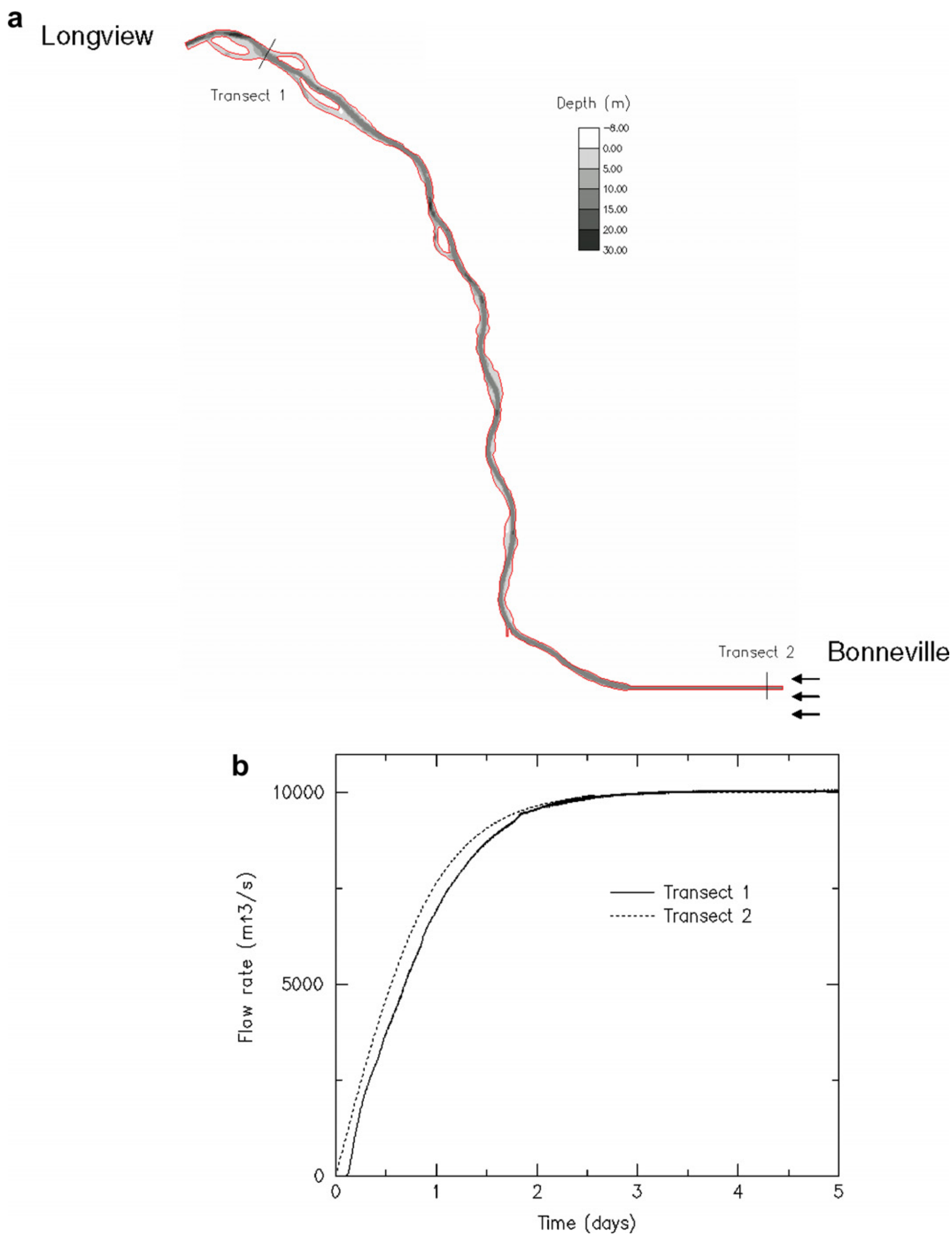


Fig. 12. (a) A river segment used to test volume conservation. Steady flow is imposed upstream of transect 2, and the flow rates are measured at both transects. (b) Time history of flow rates as measured at the two transects. ELCIRC results are indistinguishable at this scale.

tests that we have completed so far confirmed this finding, and here we present the results for the same test previously used for ELCIRC (Zhang et al., 2004).

The domain was based on the upper stretch of the Columbia River from Bonneville Dam to Longview, with channels and flats retained (Fig. 12a). The ELCIRC set-up for this problem can be found in Zhang et al. (2004). In the SELFE run, eleven S levels were deployed in the vertical grid – with $h_c = 5$ m, $\theta_b = 0.9$, and $\theta_f = 8$. A steady discharge of 10,000 m^3/s was imposed at Bonneville Dam, and the elevation at Longview was clamped at $\eta = 0$. The bottom drag coefficient (C_D) was set at 0.0025, and the horizontal viscosity coefficient, at $\gamma = 0.05$ (we found that the conservation is not sensitive to the choices of C_D or γ). The turbulence closure scheme of GLS as k-kl was used to compute the vertical viscosity and the vertical diffusivity, but the latter was not used in this barotropic case. The Coriolis factor was set to the latitude of the Columbia River (46°N). A time step of 300 s was used. The flow rates were measured at the two transects indicated in Fig. 12a throughout the run, and if volume conservation is perfect, the two rates should converge to 10,000 m^3/s after a steady state is established.

SELFE results indicate that the flow reaches a steady state soon after the ramp-up period of 2 days (Fig. 12b). The flow rates at the two transects converge to the imposed value at Bonneville with no more than 0.6% error, which is slightly worse than the ELCIRC error (0.002%). Furthermore, the volume conservation error remains at this low level and does not increase in time. The error in ELCIRC is mainly due to round-off errors as volume conservation is enforced by the finite-volume method used therein. Obviously the volume conservation property of SELFE is important as mass conservation in many transport processes will depend on a non-divergent flow field.

4.3. Adjustment under gravity

To evaluate the performance of the baroclinic model, we investigate the simple problem of adjustment under gravity, or exchange flow in a rectangular box (Lynch and Davies, 1995). Initially the box contains two fluids of different densities (ρ_1 and ρ_2) at rest, each occupying half of the domain. Gravity force will cause the heavier fluid (ρ_2) to sink and lighter fluid to rise, and internal waves are generated at the interface, and the speed of the internal waves can be estimated as $\sqrt{gh(\rho_2-\rho_1)/\rho_1}$.

In the SELFE run, the domain of 64 km × 20 km × 20 m was evenly discretized in x and y with $\Delta x = \Delta y = 500$ m (with each rectangle split into two triangles), and 23 (traditional) σ levels were used in the vertical grid, with slightly higher resolution ($\sim$0.5 m) near the bottom. There is no open boundary in this problem. The bottom was treated as frictionless, and the viscosities, diffusivity and Coriolis factor were all neglected ($\nu = \mu = f = \kappa = 0$). A time step of 300 s was used. The more accurate fifth-order Runge–Kutta method was employed for the advection in the momentum equation. To facilitate comparison with ROMS results (http://marine.rutgers.edu/poasfJan.2005), the initial salinity distribution in the x-direction was approximated by

$$S = \frac{6.25}{2}\left[1 - \tanh\frac{x - 32\,000}{1000}\right], \tag{46}$$

and the temperature was fixed at 4 °C throughout the run. The choices of the initial salinity and temperature were made to emulate the initial condition used in ROMS (i.e., the initial density varying from 1000 to 1005 kg/m^3). The positions where the 1002.5 kg/m^3 isopycnal intersects the surface and bottom were used to compute the average internal wave speed, which was compared with the linearized analytic value given in the previous paragraph as well as ROMS and ELCIRC results (Zhang et al., 2004).

The two different approaches to solve the transport equations (ELM vs. FVUM) were found to give similar results for the internal wave speed, and mass conservation errors were negligible with either approach. The salinity profile at the end of the 12-hour simulation, generated using ELM, is shown in Fig. 13 (cf. Fig. 10 in Zhang et al., 2004). The internal wave speed calculated by SELFE is 90.3% of the analytical value, which compares well with the 90–94% range obtained with ROMS with various parameter choices; on the other hand, the best ELCIRC result, with the same horizontal resolution (but with rectangular elements) and initial vertical resolution and the same choices of parameters (diffusivity, bottom friction etc) and fifth-order Runge–Kutta tracking, is only 85%. This again confirms the better accuracy of SELFE over ELCIRC. For this case, the grid orthogonality is not an issue for ELCIRC (as uniform rectangles are used), and the vertical representations are essentially equivalent for both models. Therefore the root cause for the difference is likely the lower-order shape functions used in ELCIRC.

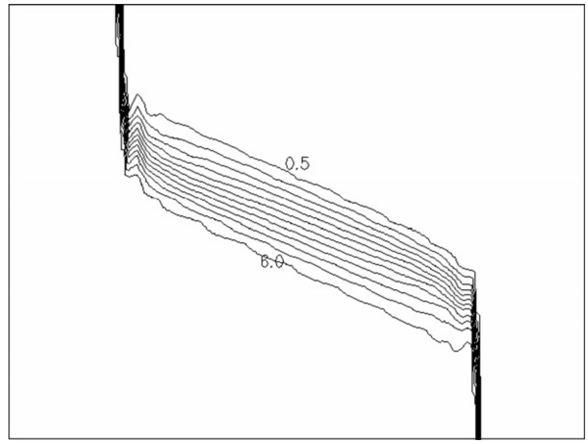


Fig. 13. Isolines of salinity at $t = 12$hr, in increment of 0.5PSU.

4.4. Unforced Columbia River plume

Given the important influence of freshwater plumes on physical, chemical, and biological processes along coastal oceans, many researchers have studied freshwater plumes using various models. Physically, the freshwater plume is essentially a surface trapped process with strong stratification occurring near the surface. Kourafalou et al. (1996ab), Garvine (1999), and Garcia-Berdeal et al. (2002) used terrain-following coordinate models to investigate the plumes on idealized geometry and bathymetry with very mild bottom slope. Whitney and Garvine (2006) studied the Delaware Bay outflow with ECOM3D, a descendant of POM (Blumberg and Mellor, 1987), and compared numerical results with observation. In their study the bottom slope is very mild ($\sim$0.03°) and the plume extent is very small (only about 15 km offshore; cf. their Fig. 6) as the estuary is well mixed by tides even during freshet month of April (with average freshwater discharge of 1100 m^3/s). The mild slopes found in these studies effectively masked the difficulty for terrain-following coordinate models by alleviating the hydrostatic inconsistency.

For many large rivers like Columbia River, the freshwater plume extends hundreds of kilometers offshore into regions with steep bottom slope. For example, the Astoria Canyon, which is situated $\sim$23 km from the mouth and well within the reach of the Columbia River plume most of the time, has an average slope of 3°, with the adjacent shelf break having roughly the same steepness (cf. Fig. 14b). Therefore the test presented in this sub-section is a very severe test for any terrain-following coordinate model. In fact, the plume predicted by the "pure *S*" SELFE model disintegrates no later than 2 weeks, even if a large number of vertical levels are used. The plume is stabilized with the addition of *Z* layers below the *S* layers. Together with the fact that the authors have successfully studied the freshwater plume of the Columbia River using the *Z* coordinate model ELCIRC (Zhang et al., 2004; Baptista et al., 2005), we conclude that the hydrostatic inconsistency is the root cause for the failure of the pure *S* model of SELFE for this problem.

We study the "unforced" river plume of the Columbia River, i.e., without tides, ambient currents or wind stirring. It is more challenging to study the "unforced" plume than the real "forced" plume, as the tides, ambient currents and wind would mask numerical instabilities. The unforced plume also gives an indication of the residual plume under mild winds and reveals some natural tendencies and key features of the plume as indicated in Fong and Geyer (2002). Therefore the study shown in this sub-section serves

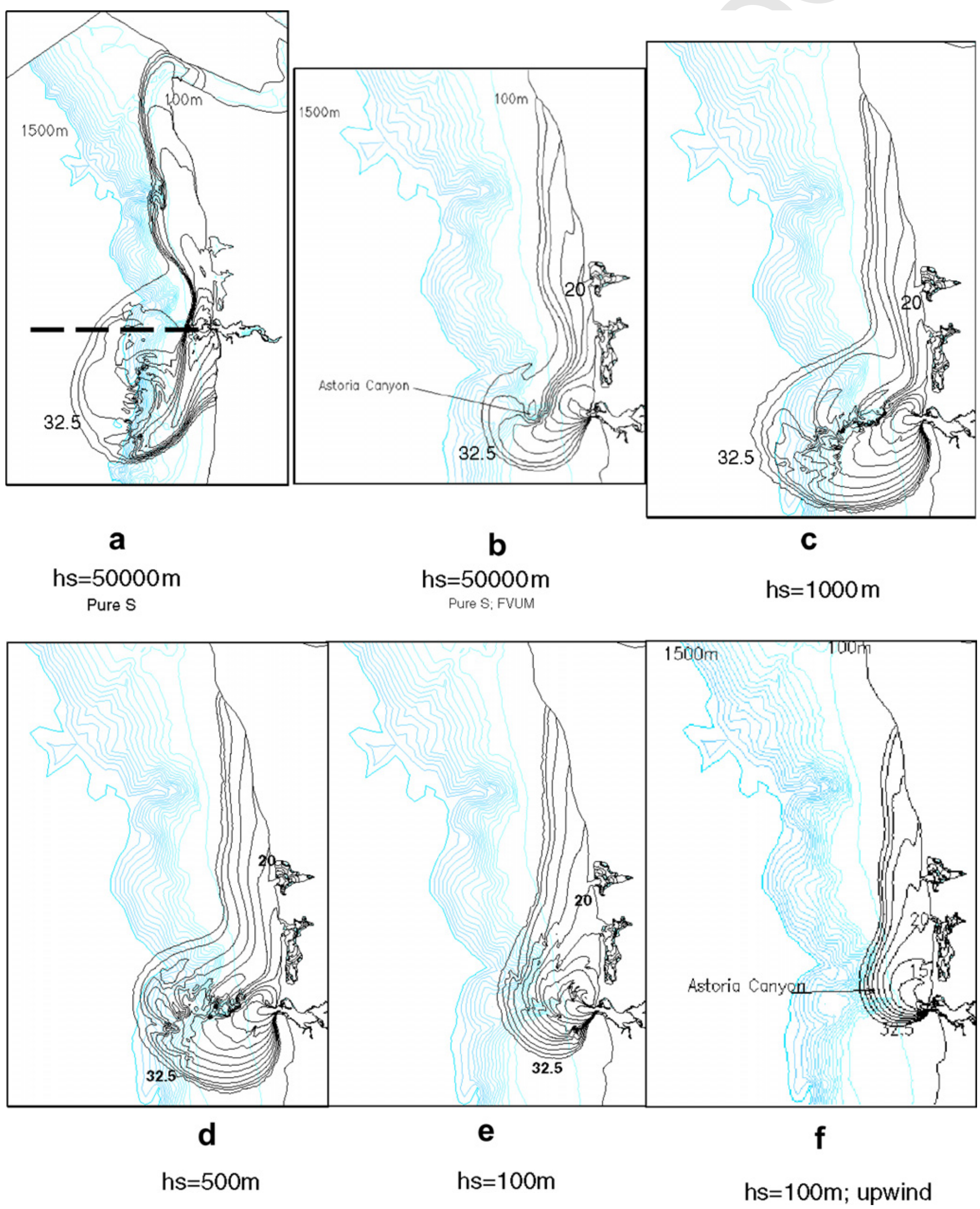
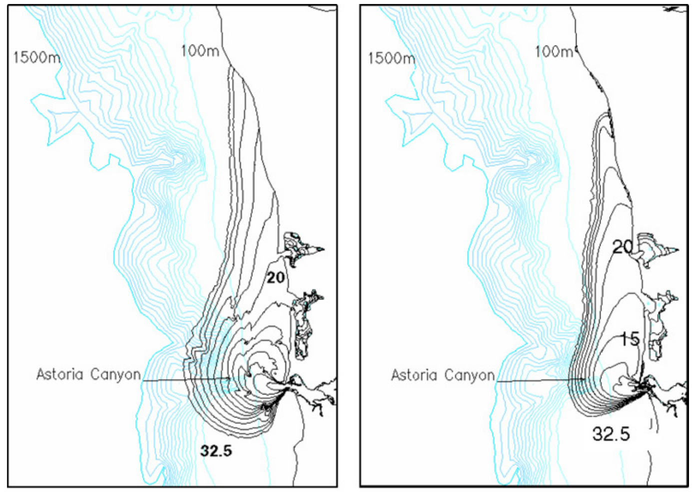


Fig. 14. Surface plume at $t = 2$ weeks. The salinity contour is from 0 to 32.5 PSU in increments of 2.5 PSU; the background bathymetry contour is from 0 to 1500 m in increments of 100 m. (a) SELFE with $h_s = 50{,}000$ m and ELM (pure S model; different horizontal scale from other plots to see the full extent of the plume); (b) SELFE with $h_s = 50{,}000$ m and FVUM; (c) SELFE with $h_s = 1000$ m and ELM; (d) SELFE with $h_s = 500$ m and ELM; (e) SELFE with $h_s = 100$ m and ELM; (f) SELFE with $h_s = 100$ m and FVUM; (g) SELFE with $h_s = 40$ m and ELM; (h) ELCIRC. The broken line in Fig. 14a indicates the transect used in Fig. 15.



g hs=40m **h** ELCIRC

Fig 14. (*continued*)

as a crucial first step in successfully simulating the real Columbia River plume.

As pointed out by many authors (Fong and Geyer, 2002; Isobe, 2005; Horner-Devine et al., 2006), the unforced plume is only quasi-steady in the sense that the bulge will grow at a rate proportional to $(tR_0)^{1/4}$ approximately, where R_0 is the inflow Rossby number, because the coastal jet in the direction of Kelvin wave propagation cannot carry away the entire freshwater outflow. However, it is important to differentiate between the quasi-steady nature of the unforced plume and the disintegration of the plume due to numerical instability discussed below. As a matter of fact, none of the numerical studies mentioned at the beginning of this sub-section or field observation (Hickey et al., 1998; Baptista et al., 2005) published so far have indicated such a mode of disintegration.

Realistic, unsmoothed[6] bathymetry was used in this study. The long-term average discharge of the Columbia River, 7000 m^3/s, was the only external forcing applied to the system and was imposed at ~88 km upstream from the mouth which is beyond the maximum salt intrusion. The ambient ocean salinity and temperature were set at 33 PSU and 10 °C. A total of 37,146 triangles was used in the horizontal grid, with a higher resolution (<500 m) concentrated in the near-plume and estuary regions. The closure scheme of $k{-}kl$ (Umlauf and Burchard, 2003; Baptista et al., 2005) was used. The spacing constants were chosen as: $h_c = 30$ m, $\theta_b = 0.7$, $\theta_f = 10$, and the time step was set at 90 s, which translated to large Courant numbers in the estuary and near-field plume (Baptista et al., 2005).

The results for the tests detailed in the next paragraph are qualitatively similar between ELM and FVUM transport schemes, and therefore majority of tests were conducted using ELM. However it is important to note that the plume predicted by the FVUM scheme is generally smaller than that predicted from the ELM scheme (Fig. 14e and f), and the FVUM scheme has been found to be more accurate than the ELM scheme when compared with the real Columbia River plume, because the former is mass conservative.

To assess the sensitivity of the plume to the choice of the vertical grid, and to illustrate the origin and growth of the numerical instability, we present results from the following seven runs (different numbers of *S* and *Z* levels were used to ensure that the transition of the vertical grid is smooth between *S* and *Z* layers):

1. $h_s = 50{,}000$ m, 0 *Z* layers, 60 *S* layers; ELM;
2. $h_s = 50{,}000$ m, 0 *Z* layers, 60 *S* layers; FVUM;
3. $h_s = 1000$ m, 5 *Z* layers, 40 *S* layers; ELM;
4. $h_s = 500$ m, 8 *Z* layers, 36 *S* layers; ELM;
5. $h_s = 100$ m, 17 *Z* layers, 36 *S* layers; ELM;
6. $h_s = 100$ m, 17 *Z* layers, 36 *S* layers; FVUM;
7. $h_s = 40$ m, 23 *Z* layers, 30 *S* layers; ELM.

Table 1
Parameters used in the ELCIRC and SELFE run 5 for the plume test (Section 4.4)

Model	ELCIRC	SELFE
Horizontal grid	Hybrid triangular/ quadrangular elements	Triangular elements
Vertical grid	61 *Z* layers	17 *Z* + 36 *S* layers
Momentum advection	Eulerian tracking + linear interpolation ELM	Eulerian tracking + linear interpolation ELM
Transport advection	Eulerian tracking + linear interpolation ELM	Eulerian tracking + linear interpolation ELM
Turbulence closure	GLS as *k*–*kl*	GLS as *k*–*kl*

[6] Bathymetry smoothing is commonly used in terrain-following models, but we found that the pure *S* coordinates in SELFE cannot produce a stable plume even with heavy bathymetry smoothing.

Please cite this article in press as: Zhang, Y., Baptista, A.M., SELFE: A semi-implicit Eulerian–Lagrangian finite-element model ..., Ocean Modell. (2008), doi:10.1016/j.ocemod.2007.11.005

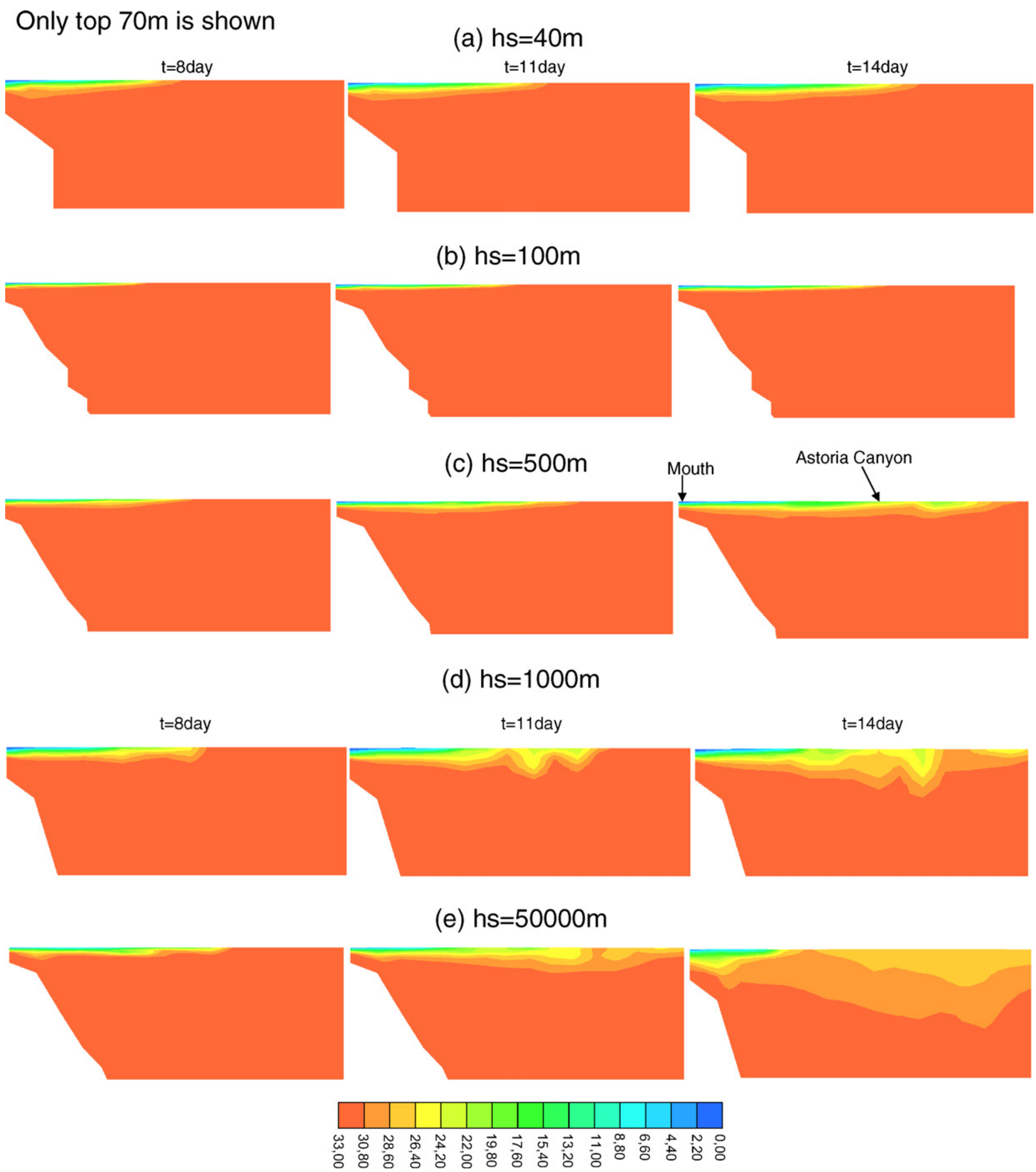


Fig. 15. Evolution of vertical profiles of salinity for the top 70 m along the transect in Fig. 14a. (a–e) $h_s = 40$, 100, 500, 1000, 50,000 m with ELM transport.

The first two runs are effectively pure S runs because h_s is larger than the maximum depth in the domain. We also conducted a reference run with ELCIRC, using 61 Z layers (see Baptista et al. (2005) for layer arrangement). Table 1 summarizes the main differences between run 5 (the best ELM-based SELFE run) and the ELCIRC run. Comparisons of surface salinity profiles at the end of 2 weeks, across all SELFE and ELCIRC runs, are shown in Fig. 14. The horizontal scale is the same across all figures except in the two pure S runs (Fig. 14a and b), where a different scale was necessary to show the disintegrated plume.

It is clear from these figures that the plume is very sensitive to the choice of the vertical grid, and in particular, to the demarcation depth h_s; no stable plume is obtained with $h_s \geqslant 500$ m (Fig. 14a–d). The instability starts as the plume first reaches the Astoria Canyon, and thereafter the isohalines become noisy and patchy and the vertical advection is greatly exaggerated in this region. The balance of force inside the plume bulge (see the discussions near the end of this sub-section) breaks down and part of the freshwater is directed southwestward, and eventually spins off to form whirlpools of freshwater offshore. The stability increases as

Please cite this article in press as: Zhang, Y., Baptista, A.M., SELFE: A semi-implicit Eulerian–Lagrangian finite-element model ..., Ocean Modell. (2008), doi:10.1016/j.ocemod.2007.11.005

h_s is reduced, and a stable plume is obtained when $h_s < 500$ m (Fig. 14e–g). Extension of the runs to 4 weeks suggests similar results: the unstable plumes continue to disintegrate while the stable plumes remain stable; the bulge in the stable plumes will continue to grow but remain circular. Comparison between Fig. 14e and f indicates that the FVUM scheme has produced a tighter freshwater bulge and a "saltier" plume than the ELM scheme. The plumes in Figs. 14e and g are very similar to each other; comparison with observation (after the external forcings being added) indicates that the vertical grid used in run 6 yields the best accuracy.

The cause for the disintegration becomes more apparent if we examine the salinity profiles along a vertical transect from the mouth to the Astoria Canyon (Fig. 14a). Fig. 15 shows the evolution of the isohalines along such a transect in the top 70 m. For $h_s = 100$ m or 40 m, the plume thins out rapidly in the cross-shore direction, which is the expected behavior of the plume (Fong and Geyer, 2002) (Fig. 15a and b). For $h_s \geqslant 500$ m, the plume thins out smoothly until it encounters the steeper slope in Astoria Canyon and adjacent shelf break (Fig. 15c–e). Numerical instability in the form of wave-like wiggles starts to develop there and quickly grows over time and spreads to deeper depths. The larger h_s is the sooner and deeper the wiggles develop. As h_s is increased, more S coordinate lines are drawn towards the bottom in the deeper depth, which in turn leads to larger truncation errors in the evaluation of the baroclinic pressure gradient. Different methods for evaluating the baroclinic pressure gradient, using either density Jacobian form or the Z-space approach in conjunction with the third-order integration rule, yielded essentially the same results. We hypothesize that other sophisticated methods will unlikely yield substantially better results. The origin of this instability is, in all likelihood, hydrostatic inconsistency. The deeper Z layers effectively act as a "stabilizer" in preventing the wiggles from growing because the Z coordinates do not suffer from the hydrostatic inconsistency. Therefore the hybrid SZ coordinates used in SELFE are crucial in obtaining a stable plume.

The stable plumes as in Fig. 14e–h show a nearly circular bulge with anti-cyclonic turning of the freshwater immediately outside the mouth and a narrow coastal jet propagating northward in the form of a Kelvin wave. The southward extent of the plume is limited. All these features are consistent with the results from previous idealized studies (Kourafalou et al., 1996a,b; Garvine, 1999; Garcia-Berdeal et al., 2002). Due to the large river discharge and a narrow mouth in Columbia River, the Rossby number (∼0.4) is high and therefore the bulge is nearly circular instead of semi-circular (Fong and Geyer, 2002). As shown by Fong and Geyer (2002) and Horner-Devine et al. (2006), the coastal jet is nearly geostrophic, and inside the bulge centrifugal and Coriolis forces balance the pressure gradient ("gradient-wind" balance). The noisy isohalines inside the bulge as discussed above initially increase the pressure gradient, which in turn enhances the centrifugal acceleration. As a result, parts of the plume are detached from the main bulge and spun off as freshwater whirlpools.

SELFE runs ∼3.2 times faster than real time on a 2.2 GHz AMD Opteron processor for run 6, with approximately 2.4 million active prism faces. The horizontal and vertical grids used in run 6 are essentially the same as those used for forecast and multi-year hindcasts. By contrast, ELCIRC with 61 Z layers runs ∼5 times faster than real time, i.e., 1.56 times faster than SELFE. Although the relative efficiency of SELFE and ELCIRC depends on the details of the problem and grid and parameter choices, we find SELFE to be moderately more expensive than ELCIRC in most cases, mainly because of the evaluation of the finite-element integrals.

5. Concluding remarks

The development of SELFE as a model to predict baroclinic circulations in bays and estuaries was driven by the challenges in modeling cross-scale "river-to-ocean" circulation. The use of semi-implicit time stepping, Eulerian–Lagrangian treatment of advection, and a formal finite-element framework has enabled SELFE to efficiently and robustly simulate complex circulations as found in the Columbia River. The strong computational performance of SELFE is supported by the findings presented in this paper.

Since 2005, SELFE has become an integral part of the Columbia River observing system CORIE (Baptista, 2006), replacing ELCIRC (Zhang et al., 2004) as the default model for generating CORIE daily forecasts (e.g., Fig. 2) and long-term simulation databases (e.g., Fig. 1) of Columbia River estuarine and plume circulation. SELFE is also at the core of a rapid-deployment national estuarine forecasting system (NEFS), under development as a pilot project for the US Integrated Ocean Observing System. Results from SELFE in the context of CORIE and NEFS will be described in separate publications. SELFE is an open-source code, available at http://www.ccalmr.ogi.edu/CORIE/modeling/selfe/.

~~6. Uncited references~~

~~Fortunato et al. (1994), Gross et al. (2002) and Lynch et al. (1996).~~ Q1

Acknowledgements

The development and testing of SELFE has greatly benefited from the contributions of many colleagues. Within the CORIE team, Paul Turner and Charles Seaton provided the software tools that allowed the analysis of SELFE results during development and application, and Michael Wilkin led in collecting the CORIE oceanographic data that provided groundtruthing for the development of SELFE (e.g., Figs. 1 and 2). Dr. Edmundo Casillas of the National Oceanic and Atmospheric Administration

(NOAA) provided the original application. Thanks are due to early beta-testers of SELFE, in particular to Dr. Mike Foreman of the Institute of Ocean Sciences, Canada, and OHSU students Ryan Kilgren and Nate Hyde. The National Science Foundation (ACI-0121475; OCE-0239072; OCE-0424602), Bonneville Power Administration (19126; 23677; 28143) and National Oceanic and Atmospheric Administration (AB133F04CN0033) provided financial support for this research. Any statements, opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views or policies of the federal sponsors, and no official endorsement should be inferred.

Appendix A. Terrain-following coordinates in SELFE

As discussed in Section 3.1, the terrain-following generalized S–coordinate system (Song and Haidvogel, 1994) is used in the upper part of the water column. The transformation from S to Z is given by:

$$\begin{cases} z = \eta(1+\sigma) + h_c\sigma + (\tilde{h} - h_c)C(\sigma) \\ \quad (-1 \leqslant \sigma \leqslant 0) \\ C(\sigma) = (1-\theta_{\mathrm{b}})\frac{\sinh(\theta_{\mathrm{f}}\sigma)}{\sinh\theta_{\mathrm{f}}} + \theta_{\mathrm{b}}\frac{\tanh[\theta_{\mathrm{f}}(\sigma+1/2)] - \tanh(\theta_{\mathrm{f}}/2)}{2\tanh(\theta_{\mathrm{f}}/2)} \\ \quad (0 \leqslant \theta_{\mathrm{b}} \leqslant 1; 0 < \theta_{\mathrm{f}} \leqslant 20) \end{cases} \tag{47}$$

where $\tilde{h} = \min(h, h_{\mathrm{s}})$ is a "restricted" depth, h_c is a positive constant dictating the thickness of the bottom or surface layer that needs to be resolved, and θ_{b} and θ_{f} are constants that control the vertical resolution near the bottom and surface. As $\theta_{\mathrm{f}} \to 0$, the S coordinates reduce to the traditional σ-coordinates:

$$z = \widetilde{H}\sigma + \eta, \tag{48}$$

where $\widetilde{H} = \tilde{h} + \eta$ is the restricted total water depth. For $\theta_{\mathrm{f}} >> 1$, more resolution is skewed towards the boundaries, and the transformation becomes more non-linear. If $\theta_{\mathrm{b}} \to 0$, only the surface is resolved, not the bottom, while if $\theta_{\mathrm{b}} \to 1$, both are resolved. The latter case is particularly important in coastal and oceanic applications, where both bottom and surface processes are important (which is the main reason that the S coordinates were chosen over the σ coordinates in SELFE). Unfortunately, the S coordinate system becomes invalid in shallow depths; a sufficient condition for a valid S transformation can be derived from $z'(\sigma) > 0$ as:

$$\begin{cases} \widetilde{H} > h_0 \\ \tilde{h} > h_c \\ \eta > -h_c - (\tilde{h} - h_c)\frac{\theta_{\mathrm{f}}}{\sinh\theta_{\mathrm{f}}} \end{cases} \tag{49}$$

The first condition in this equation simply requires that the spot is "wet", with h_0 being a specified minimum depth of water. For a wet location, the S coordinate system becomes degenerate when $\tilde{h} \leqslant h_c$ (i.e., the depth is too shallow) or $\eta \leqslant -h_c - (\tilde{h} - h_c)\frac{\theta_{\mathrm{f}}}{\sinh\theta_{\mathrm{f}}}$ (i.e., the surface falls below a certain threshold). In either case, the transformation Eq. (47) becomes non-monotonic, and the S coordinates need to be replaced by the traditional σ coordinates, which are uniformly valid at all depths. Where a transition from S to σ is warranted, we use the following strategies to make the transition smooth:

1. $\tilde{h} \leqslant h_c$. In SELFE, the transformation Eq. (47) is replaced by:

$$z = \widetilde{H}\sigma + \eta, \tag{50}$$

From Eq. (47), as $\tilde{h} \to h_c^+$, the S coordinates approach σ coordinates, and therefore, the transition from shallow to deep regions is smooth.

2. $\tilde{h} > h_c$, but $\eta \leqslant -h_c - (\tilde{h} - h_c)\frac{\theta_{\mathrm{f}}}{\sinh\theta_{\mathrm{f}}}$. In this case, the "nearest valid set" is:

$$\begin{cases} \hat{\sigma} = \frac{\hat{z} - \hat{\eta}}{\tilde{h} + \hat{\eta}} \\ \hat{\eta} = \beta\left[-h_c - (\tilde{h} - h_c)\frac{\theta_{\mathrm{f}}}{\sinh\theta_{\mathrm{f}}}\right] \\ \hat{z} = \hat{\eta}(1+\sigma) + h_c\sigma + (\tilde{h} - h_c)C(\sigma) \end{cases}, \tag{51}$$

where $\beta = 0.98$ is a safety factor. In practice, this case will unlikely be encountered if a sufficiently large h_c (e.g., $h_c > 5$ m) is used; a large h_c is recommended for most SELFE applications.

References

Baptista, A.M., 1987. Solution of advection-dominated transport by Eulerian–Lagrangian Methods using the backwards method of characteristics. Ph.D. Dissertation, MIT, Cambridge.

Baptista, A.M., 2006. CORIE: the first decade of a coastal-margin collaborative observatory. In: Oceans'06, MTS/ IEEE, Boston, MA.

Baptista, A.M., Zhang, Y.L., Chawla, A., Zulauf, M., Seaton, C., Myers III, E.P., Kindle, J., Wilkin, M., Burla, M., Turner, P.J., 2005. A cross-scale model for 3D baroclinic circulation in estuary–plume–shelf systems: II. Application to the Columbia River. Cont. Shelf Res. 25, 935–972.

Barron, C.N., Kara, A.B., Martin, P.J., Rhodes, R.C., Smedstad, L.F., 2006. Formulation, implementation and examination of vertical coordinate choices in the Global Navy Coastal Ocean Model (NCOM). Ocean Modell. 11, 347–375.

Blumberg, A.F., Mellor, G.L., 1987. A description of a three-dimensional coastal ocean circulation model. In: Heaps, N. (Ed.), Three-Dimensional Coastal Ocean Models, . In: Coastal and Estuarine Studies, vol. 4. AGU, Washington, DC, pp. 1–16.

Canuto, V.M., Howard, A., Cheng, Y., Dubovikov, M.S., 2001. Ocean turbulence I: one-point closure model. Momentum and heat vertical diffusivities. J. Phys. Oceanogr. 31, 1413–1426.

Casulli, V., Cattani, E., 1994. Stability, accuracy and efficiency of a semi-implicit method for 3D shallow water flow. Comput. Math. Appl. 27, 99–112.

Casulli, V., Cheng, R.T., 1992. Semi-implicit finite difference methods for three-dimensional shallow water flow. Int. J. Numer. Methods Fluids 15, 629–648.

Casulli, V., Walters, R.A., 2000. An unstructured grid, three-dimensional model based on the shallow water equations. Int. J. Numer. Methods Fluids 32, 331–348.

Casulli, V., Zanolli, P., 2005. High resolution methods for multidimensional advection–diffusion problems in free-surface hydrodynamics. Ocean Modell. 10, 137–151.

Chen, C., Liu, H., Beardsley, R.C., 2003. An unstructured grid, finite-volume, three-dimensional, primitive equations ocean model: application to coastal ocean and estuaries. J. Atmos. Oceanic Technol. 20, 159–186.

Cheng, H.P., Yeh, G.T., Cheng, J.R., 2000. Modeling bay/estuary hydrodynamics and contaminant/sediment transport. Adv. Environ. Res. 4, 187–209.

Clark, H.L., Isern, A., 2003. The OOI and The IOOS – can they be differentiated? An NSF perspective. Oceanography 16, 20–21.

Danilov, S., Kivman, G., Schroter, J., 2004. A finite element ocean model: principles and evaluation. Ocean Modell. 6, 125–150.

Flather, R.A., 1987. A tidal model of Northeast Pacific. Atmosphere–Ocean 25, 22–45.

Fong, D.A., Geyer, R.W., 2002. The alongshore transport of freshwater in a surface-trapped river plume. J. Phys. Oceanogr. 32, 957–972.

Foreman, M.G.G., Stucchi, D., Zhang, Y.L., Baptista, A.M., 2006. Estuarine and tidal currents in the Broughton Archipelago. Atmosphere–Ocean 44 (1), 47–63.

~~Fortunato, A.B., Baptista, A.M., 1994. Localized sigma coordinates for the vertical structure of hydrodynamic models. In: Spaulding, M.L. et al. (Eds.), Estuarine and Coastal Modeling III. American Society of Civil Engineers, pp. 323–335.~~

Fortunato, A.B., Baptista, A.M., 1996. Evaluation of horizontal gradients in sigma-coordinate shallow water models. Atmosphere–Ocean 34, 489–514.

Fringer, O.B., Gerritsen, M., Street, R.L., 2006. An unstructured-grid, finite-volume, nonhydrostatic, parallel coastal ocean simulator. Ocean Modell., 139–173.

Galperin, B., Kantha, L.H., Hassid, S., Rosati, A., 1988. A quasi-equilibrium turbulent energy model for geophysical flows. J. Atmos. Sci. 45, 55–62.

Garcia-Berdeal, I., Hickey, B.M., Kawase, M., 2002. Influence of wind stress and ambient flow on a high discharge river plume. J. Geophys. Res. 107, 3130.

Garvine, R.W., 1999. Penetration of buoyant coastal discharge onto the continental shelf: a numerical model experiment. J. Phys. Oceanogr. 29, 1892–1909.

Gary, J.M., 1973. Estimate of truncation error in transformed coordinate primitive equation atmospheric models. J. Atmos. Sci. 30, 223–233.

~~Gross, E.S., Bonaventura, L., Rosatti, G., 2002. Consistency with continuity in conservative advection schemes for free-surface models. Int. J. Numer. Methods Fluids 38, 307–327.~~

Ham, D.A., Pietrzak, J., J., Stelling, G.S., 2005. A scalable unstructured grid 3-dimensional finite volume model for the shallow water equations. Ocean Modell. 10, 153–169.

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

Hickey, B.M., Banas, N.S., 2003. Oceanography of the US Pacific Northwest Coastal Ocean and Estuaries with Application to Coastal Ecology. Estuaries 26, 1010–1031.

Hickey, B.M., Pietrafesa, L.J., Jay, D.A., Boicourt, W.C., 1998. The Columbia River plume study: subtidal variability in the velocity and salinity fields. J. Geophys. Res. 103, 10339–13368.

Horner-Devine, A.R., Fong, D.A., Monismith, S.G., Maxworthy, T., 2006. Laboratory experiments simulating a coastal river inflow. J. Fluid Mech. 555, 203–232.

Iskandarani, M., Haidvogel, D.B., Levin, J.C., 2003. A three-dimensional spectral element model for the solution of the hydrostatic primitive equations. J. Comput. Phys. 186, 397–425.

Isobe, A., 2005. Ballooning of river-plume bulge and its stabilization by tidal currents. J. Phys. Oceanogr. 35, 2337–2351.

Jay, D.A., Smith, J.D., 1990a. Circulation, density distribution and neap-spring transitions in the columbia river estuary. Prog. Oceanogr. 25, 81–112.

Jay, D.A., Smith, J.D., 1990b. Residual circulation in shallow estuaries: I. highly stratified, narrow estuaries. J. Geophys. Res. 95, 711–731.

Kantha, L.H., Clayson, C.A., 1994. An improved mixed layer model for geophysical applications. J. Geophys. Res. 99 (25), 235–266.

Kourafalou, V.H., Oey, L.-Y., Wang, J.D., Lee, T.N., 1996a. The fate of river discharge on the continental shelf, part I: modeling the river plume and shelf coastal current. J. Phys. Res. 101, 3415–3434.

Kourafalou, V.H., Lee, T.N., Oey, L.-Y., Wang, J.D., 1996b. The fate of river discharge on the continental shelf, part I: modeling the river plume and shelf coastal current. J. Geophys. Res. 101, 3435–3456.

Labeur, R.J., Pietrzak, J., 2005. A fully three dimensional unstructured grid non-hydrostatic finite element coastal model. Ocean Modell. 10, 51–67.

Lapidus, L., Pinder, G.F., 1982. Numerical Solution of Partial Differential Equations in Science and Engineering. Wiley-Interscience.

Leupi, C., Altinakar, M.S., 2005. Finite element modeling of free-surface flows with non hydrostatic pressure and *k*–epsilon turbulence model. Int. J. Numer. Methods Fluids 49, 149–170.

Luettich, R.A., Westerink, J.J., Scheffner, N.W. 1991. ADCIRC: an advanced three-dimensional circulation model for shelves, coasts and estuaries. Coast. Engrg. Res. Ct., US Army Engs. Wtrways. Experiment Station, Vicksburg, MS Report 1: Theory and Methodology of ADCIRC-2DDI and ADCIRC-3DL.

Luettich, R.A., Muccino, J.C., Foreman, M.G.G., 2002. Considerations in the calculation of vertical velocity in three-dimensional circulation models. J. Atmos. Ocean. Technol. 19, 2063–2076.

Lynch, D.R., Davies, A.M., 1995. Quantitative Skill Assessment for Coastal Ocean Models. American Geophysical Union.

Lynch, D.R., Gray, W.G., 1978. Analytic solutions for computer flow model testing. ASCE J. Hydraul. Div. 104, 1409–1428.

Lynch, D.R., Officer, C.B., 1985. Analytical test cases for three-dimensional hydrodynamic models. Int. J. Numer. Methods Fluids 5, 529–543.

Lynch, D.R., Werner, F.E., 1991. Three-dimensional hydrodynamics on finite elements. Part II: non-linear time-stepping model. Int. J. Numer. Methods Fluids 12, 507–533.

~~Lynch, D.R., Ip, J.T., Naimie, C.E., Werner, F.E., 1996. Comprehensive coastal circulation model with application to the Gulf of Maine. Cont. Shelf Res. 16, 875–906.~~

Marshall, J., Hill, C., Perelman, L., Adcroft, A., 1997. Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling. J. Geophys. Res. 102 (C3), 5733–5752.

Martin, D.L., 2003. The National Oceanographic Partnership Program, Ocean US, and real movement towards an integrated and sustained ocean observing system. Oceanography 16, 13–19.

Mellor, G.L., Yamada, T., 1982. Development of a turbulence closure model for geophysical fluid problems. Rev. Geophys. 20, 851–875.

Miglio, E., Quarteroni, A., Saleri, F., 1999. Finite element approximation of quasi-3D shallow water equations. Comput. Methods Appl. Mech. Eng. 174 (3), 355–369.

Oliveira, A., Baptista, A.M., 1995. A comparison of integration and interpolation Eulerian–Lagrangian methods. Int. J. Numer. Methods Fluids 21, 183–204.

Oliveira, A., Baptista, A.M., 1998. On the role of tracking on Eulerian–Lagrangian solutions of the transport equation. Adv. Water Res. 21, 539–554.

Pietrzak, J., Jakobson, J.B., Burchard, H., Vested, H.J., Petersen, O., 2002. A three-dimensional hydrostatic model for coastal and ocean modelling using a generalised topography following coordinate system. Ocean Modell. 4, 173–205.

Pinto, L., Fortunato, A.B., Oliveira, A., Baptista, A.M., 2004. Haline stratification in the Guadiana estuary: II. Numerical modeling. Recursos Hidricos 25, 99–110.

Pond, S., Pickard, G.L., 1998. Introductory Dynamical Oceanography. Butterworth-Heinmann.

Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T., 1992. Numerical Recipes in C – The Art of Scientific Computing, second ed. Cambridge University Press, Cambridge/ New York.

Rodi, W., 1984. Turbulence models and their applications in hydraulics: a state of the art review. International Association for Hydraulics Research, Delft, The Netherlands.

Roux, Le D.Y., Lin, C.A., Staniforth, A., 1997. An accurate interpolating scheme for semi-Lagrangian advection on an unstructured mesh for ocean modelling. Tellus 49A, 119–138.

Shapiro, R., 1970. Smoothing, filtering and boundary effects. Rev. Geophys. Space Phys. 8 (2), 359–387.

Shchepetkin, A.F., McWilliams, J.C., 2003. A method for computing horizontal pressure-gradient force in an oceanic model with a non-aligned vertical coordinate. J. Geophys. Res. 108, 1–34.

Shchepetkin, A.F., McWilliams, J.C., 2005. The regional oceanic modeling system (ROMS): a split-explicit, free-surface, topography-following-coordinate, oceanic model. Ocean Modell. 9, 347–404.

Smagorinsky, J., 1963. General circulation experiments with the primitive equations: I. The basic experiment. Monthly Weather Rev. 91, 99–164.

Song, Y., Haidvogel, D., 1994. A semi-implicit ocean circulation model using a generalized topography-following coordinate system. J. Comput. Phys. 115, 228–244.

Sweby, P.K., 1984. High resolution schemes using flux limiters for hypobolic conservation laws. SIAM J. Numer. Anal. 21 (5), 995–1011.

Umlauf, L., Burchard, H., 2003. A generic length-scale equation for geophysical turbulence models. J. Mar. Res. 6, 235–265.

Walters, R.A., 2005. Coastal ocean models: two useful finite element methods. Cont. Shelf Res. 25, 775–793.

Westerink, J.J., Feyen, J.C., Atkinson, J.H., Luettich, R.A., Dawson, C.N., Powell, M.P., Dunion, J.P., Roberts, H.J., Kubatko, E.J., Pourtaheri, H. 2004. A new generation hurricane storm surge model for southern Louisiana. http://www.nd.edu/~adcirc/pubs/westerinketal_bams_ref1935b.pdf.

Whitney, M.M., Garvine, R.W., 2006. Simulating the Delaware Bay buoyant outflow: comparison with observations. J. Phys. Oceanogr. 36, 3–21.

Wicker, L.J., Skamarock, W.C., 1998. A time-splitting scheme for the elastic equations incorporating second-order Runge–Kutta time differencing. Monthly Weather Rev. 126, 1992–1999.

Wilcox, D.C., 1998. Reassessment of scale determining equation for advance turbulence models. AIAA J. 26, 1299–1310.

Zeng, X., Zhao, M., Dickinson, R.E., 1998. Intercomparison of bulk aerodynamic algorithms for the computation of sea surface fluxes using TOGA COARE and TAO data. J. Clim. 11, 2628–2644.

Zhang, Y.L., Baptista, A.M., Myers, E.P., 2004. A cross-scale model for 3D baroclinic circulation in estuary–plume–shelf systems: I. Formulation and skill assessment. Cont. Shelf Res. 24, 2187–2214.

Zhang, Y.L., ~~Priest, G.R.,~~ Baptista, A.M., ~~in preparation.~~ ~~Tsunami inundation modeling with a new unstructured finite-element model.~~ ~~Science of Tsunami Hazards.~~ Q2

Please cite this article in press as: Zhang, Y., Baptista, A.M., SELFE: A semi-implicit Eulerian–Lagrangian finite-element model ..., Ocean Modell. (2008), doi:10.1016/j.ocemod.2007.11.005