Two finite-volume unstructured mesh models for large-scale ocean modeling

S. Danilov*

Alfred Wegener Institute for Polar and Marine Research, 27515 Bremerhaven, Germany

Abstract

Two approaches pertaining to modeling large-scale ocean circulation on unstructured meshes are described. Both use the finite-volume ideology, unstructured surface triangular mesh and geopotential vertical coordinate, and promise better numerical efficiency than $P_1 - P_1$ finite element models. The first one is formulated on median-dual control volumes for all variables and presents a finite-volume implementation of $P_1 - P_1$ finite-element discretization (A-grid). The second one differs by the cell-centered placement of horizontal velocities (quasi-B-grid). Two practical tasks have to be solved to ensure their stable performance in long-term simulations. For triangular A-grids, it is the stabilization against pressure modes triggered by the stepwise bottom topography. The proposed solution preserves volume and tracers by introducing a composite representation for the horizontal velocity (with an elementwise-constant velocity correction). The quasi-B-grid setup is free of pressure modes but requires efficient filtering and dissipation in the momentum equation because of its too large velocity space. Implementations of momentum advection and viscosity that serve this goal are proposed. Both setups show stable performance and similar numerical efficiency, as exemplified by simulations of a baroclinic channel flow and circulation in the North Atlantic.

Key words: Unstructured meshes, Finite volumes, large-scale ocean circulation

1. Introduction

There are many ways unstructured meshes can be helpful in large-scale ocean modeling, most obviously by providing a local focus in a global con-

*Corresponding author.

Email address: Sergey.Danilov@awi.de (S. Danilov)

figuration without nesting and open boundaries. Other appealing features like mesh adaptivity are potentially of interest in some broader context (for a review, see, e. g., Piggott et al. (2008)).

The experience gained thus far with the Finite-Element Sea-ice Ocean circulation Model (FESOM) (Wang et al. (2008), Timmermann et al. (2009)) indicates that unstructured meshes present a sensible approach to modeling ocean circulation in configurations requiring a regional focus in an otherwise global context; the approach becomes rather efficient on meshes with a large refinement factor (≤ 20). It has also shown that a strong gain in numerical efficiency is desirable in order to be practical in situations where less refinement is needed. Discretizations based on finite volume (FV) method promise better computational efficiency (see, e. g. Blazek (2001)) and thus it seems natural to follow their ideology. There also are good examples to follow, one suggested by FVCOM (Chen et al. , 2003), and others proposed by the atmospheric modeling community (see, e. g., Szmelter and Smolarkiewicz (2010) and Ringler et al. (2010)).

There are more subtle issues as well. Continuous Galerkin (CG) finite-element (FE) discretizations (used by FESOM and several other models) face difficulties when solving for hydrostatic pressure and vertical velocity in hydrostatic codes. These elementary first-order problems lead to matrices with zeros at diagonals. The horizontal connections further complicate the solution by entangling all mesh nodes. Similarly, inversion of global matrices is needed if vertical diffusion or viscosity is treated implicitly. Although these difficulties can partly be alleviated by switching to vertically discontinuous elements (as in White et al. (2008)), only a fully discontinuous representation reintroduces 'locality' to the discretized operators. However, the respective Discontinuous Galerkin (DG) methods prove to be more costly than the CG methods. The FV method once again emerges as a promising alternative.

This article aims at presenting two FV unstructured-mesh approaches, one using median-dual (vertex-centered) control volumes for all variables, and the other one, using cell-centered horizontal velocities, but preserving median-dual control volumes for scalar variables. A standard set of primitive equations is solved under the Boussinesq, hydrostatic and other traditional approximations. Both setups assume z-coordinate in vertical, as is common in large-scale ocean modeling. Since all variables are at mesh vertices in the horizontal plane in the first case, it will be referred to as the A-grid approach. This placement is shared with FESOM, and the A-grid is just its FV implementation. The other, cell–vertex approach will be referred to as the quasi-B-grid to emphasize staggering of variables. It shares the placement of variables with FVCOM. Judged by the ratio of velocity to scalar degrees of freedom, it is closer to the C-D grids, yet its velocities are not at edges. In the framework of FE method, the quasi-B-grid corresponds to $P_0 - P_1$ element.

These variable placements are well explored on the level of shallow water equations (see, e. g., Le Roux et al. (2007) and Le Roux and Pouliot (2008) for the analysis of $P_1 - P_1$ and $P_0 - P_1$ pairs)[1] and boast long lists of applications, too numerous to be discussed here. Our interest to these variable placements was partly motivated by their known behavior.

The other aspect is that these two grids imply different ratios between degrees of freedom in the horizontal velocity and scalars. This has implications for their performance in tasks of large-scale ocean modeling. The A-grids offer the least expensive configuration on triangular meshes with the balanced (2:1) ratio. This may be beneficial in strongly nonlinear regimes because same scales are resolved by velocities and scalars. However, just as $P_1 - P_1$ FE setups, A-grids may support pressure modes. Quasi-B-grids present an alternative without pressure modes, but introduce too many velocities. This leads to spurious inertial modes, and, more importantly, may result in strong generation of small-scale velocity variance through the momentum advection. Note that the velocity space is excessively large for many triangular discretizations proposed in the literature. Note also that many of them support spurious modes (Le Roux et al. (2007)).

The implications of these 'geometrical' features depend on typical dynamics, and the specific goal of this paper is to present solutions that work well on large scales for A- and quasi-B-grids. It turns out that the stepwise bottom of z-coordinate meshes triggers pressure modes on A-grids, and we propose a stabilization technique similar to that of FESOM which is compatible with volume and tracer conservation. The main problem of quasi-B-grids indeed proves to be their tendency to noise in eddy-resolving regimes. Its solution lies in filtering the momentum advection. The algorithms proposed below tackle this problem too. Augmented with these solutions the A- and quasi-B-grids show rather similar performance, but assume different tuning

[1]As concerns linear waves, the difference between FE and FV implementations is roughly equivalent to mass matrix lumping, which does not compromise wave dispersion (Le Roux et al. (2009)).

strategy.

Among many (sometimes sophisticated) ways of discretizing the primitive equations on unstructured meshes, those based on low-order elements are frequently preferred as they warrant geometrical flexibility at a reasonable numerical cost. Since many of them have to deal with issues introduced by the geometry of variable placement, their robust functioning depends on specific algorithms (like those mentioned above for A- and quasi-B-grids). Current challenge, in our opinion, lies in providing fast and reliable frameworks enabling real-world simulations which will feedback on the model development. It is hoped that the proposed approaches will contribute in this direction.

The material is organized as follows. Section 2 explains geometrical issues. The next sections 3 and 4 present discretizations of the two setups in some detail. Section 4 concentrates only on the momentum equation part. Since the arrangement of scalar variables is the same as on the A-grid, the discretization is similar too and is not repeated. Numerical examples illustrating functionality of two setups (baroclinic instability in a channel and circulation in the North Atlantic) are presented in section 5. Section 6 presents a short discussion and section 7 concludes. The analysis assumes plane geometry for simplicity, the spherical geometry is used in reality.

2. Placement of variables

The horizontal and vertical placement of variables is illustrated in left and right panels of Fig. 1 respectively. On an A-grid all variables are located at nodes (vertices) in the horizontal plane. We will be referring to them as nodal fields, with understanding that the name pertains only to the horizontal placement. Similarly, an elemental field is that with variables at centroids when viewed from above. On quasi-B grids the horizontal velocity is elemental, but scalar quantities and vertical component of velocity are nodal, same as on an A-grid. Note that an alternative A-grid setup is possible with all variables at centroids. It is not considered here as we would like to keep the scalar parts of A and quasi-B-grid setups as similar as possible. We use z-levels, and arrange the horizontal velocities, temperature, salinity and pressure at mid-levels, while the vertical velocity is at full levels. Let $\overline{z}_n$ denote the depth of levels, with $\overline{z}_1 = 0$ and $\overline{z}_{N_L} = -H_{max}$, where N_L is the maximum number of levels and H_{max} is the maximum depth. The depth of mid-levels is $Z_n = (\overline{z}_n + \overline{z}_{n+1})/2$, $n = 1 : N_L - 1$. The field variables will be

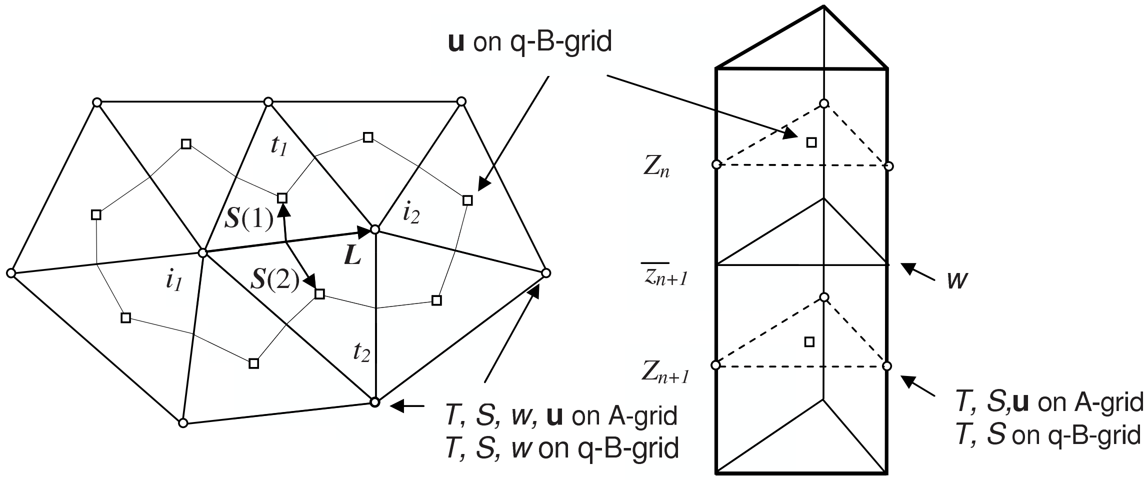

Figure 1: Schematics of mesh geometry. Left panel: In the horizontal plane, the scalar quantities and vertical velocities are located at mesh nodes (circles). The horizontal velocities are at nodes on A-grid and on centroids (squares) on quasi-B-grid. An edge is characterized by its two nodes i_1 and i_2, two neighboring triangles t_1 and t_2, the edge vector $\mathbf{L}$ directed to i_2 (t_1 on the left) and two cross-vectors $\mathbf{S}(1:2)$ directed to centroids. The median-dual control cells in the horizontal plane are formed by connecting mid-edges with centroids (thin lines). Control cells for the horizontal velocities on quasi-B-grid coincide with triangles. Three-dimensional control volumes are prisms based on respective control cells with top and bottom faces on the level surfaces $\overline{z}_n$. Right panel: In the vertical plane, the temperature, salinity, pressure and horizontal velocities are at mid-levels Z_n. The vertical velocity is at full levels $\overline{z}_n$.

distinguished by two indices, for example, T_{ni} is the value of temperature at Z_n and below the surface node i.

With each surface node i we associate a median-dual surface control cell that is built from segments connecting centroids of neighboring triangles with centers of edges containing node i. A triangle is referred to as neighboring if it contains node i. Most of operations in FV codes are edge-based. An edge j is characterized by its two nodes (i_1, i_2), the edge vector pointing to node i_2, $\mathbf{L}_j = (x_{i_2} - x_{i_1}, y_{i_2} - y_{i_1})$, two triangles sharing the edge (t_1, t_2), where t_1 is to the left of $\mathbf{L}_j$, and two cross-vectors drawn from the edge center to element centroids, $\mathbf{S}_j(1:2) = (\mathbf{x}_1, \mathbf{x}_2)$, as illustrated in Fig. 1. For boundary edges the second triangle is absent.

Since the elevation is defined at nodes, it would be natural to define the bottom topography in the same way, i. e. associate it with the scalar control cells. This however, leads to problems with respect to pressure gradient computation on A-grids. Indeed, in this case all velocity points are wet, and we have to write momentum equations for each of them. Except for the flat bottom case, there are deep locations where the neighborhoods used to compute pressure and elevation gradients are different, which is inconsistent.

Note that this difficulty would not exist on quasi-B grids because velocity locations with reduced number of neighbors are then always on vertical walls where the no-slip boundary conditions are applied. Note also that the problem is specific to z-coordinate meshes.

The alternative is to define the bottom topography on triangles, which is compatible with both A and quasi-B grids. We therefore follow it. The elementwise-constant depth of ocean may take any of $\overline{z}_n$ values for $n \geq 2$.

A 3D control volume is a prism based on respective surface control cell (median-dual for A-grid, and both median-dual and triangular for quasi-B-grid) and bounded by level surfaces at its top and bottom. Because of z-coordinate and elementwise-constant bottom topography, the deep median-dual control volumes can partly be occupied with land. For that reason it is convenient to introduce the array containing actual 'liquid' horizontal areas of scalar control volumes, A_{ni}, in addition to the array A_t of triangle areas. The area A_{ni} is related to mid-level Z_n and node i. The vertical advective flux through the upper face of control volume (n, i) involves this area, and through the lower face, $A_{(n+1)i}$. Also for convenience we introduce, for each node i, maximum and minimum numbers of levels over neighboring triangles, N_i^{max} and N_i^{min}, respectively (see Fig. 2).

Such 'partial' control volumes do not create complications for scalar quantities because vertical rigid walls contribute with zero fluxes. The A-grid horizontal velocities turn to lie at bottom singularities and the only safe option is to fix them assuming no-slip boundary conditions, as illustrated in Fig. 2. In this case the horizontal velocity is non-zero only in full control volumes, i. e., in layers from 1 to $N_i^{min} - 1$. The vertical velocity is not constrained in that way because it must react to convergence (divergence) of volume fluxes through the 'liquid' vertical faces of control volumes.

On quasi-B-grid the horizontal velocity locations are always 'wet' and thus both free-slip and non-slip boundary conditions are allowed.

Admittedly, because of boundary conditions in z-coordinate setups A-grids are disadvantageous in narrow straits. More importantly, in shallow regions with rough topography they may over-constrain the solution and trigger a noisy response in the vertical velocity and elevation. It is mainly this induced noise that makes stabilization (see further) indispensable on z-coordinate meshes.

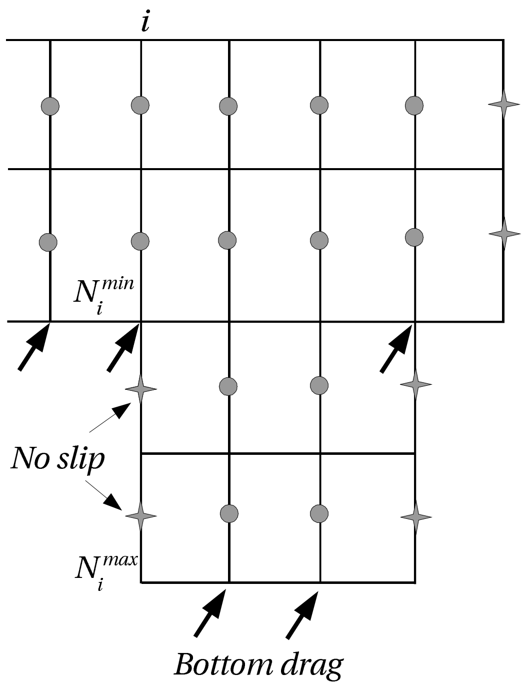

Figure 2: Schematics explaining boundary conditions on the horizontal velocity on A-grid. The horizontal velocities at vertical wall edges are set to zero (four-stars). The 'partial' control volumes hosting these locations are skipped in horizontal velocity computations, so that one always deals with full control volumes in layers from $n = 1$ to $n = N_i^{min} - 1$. Arrows show locations where the bottom drag is applied. The vertical velocity is zero only at bottom locations, but is allowed at vertical walls to accommodate volume fluxes through faces of control volumes.

3. Triangular A-grid

The A-grid setup was inspired by the work by Szmelter and Smolarkiewicz (2010) on the edge-based (median-dual) unstructured mesh discretization in geospherical framework and the fact that it corresponds to FESOM (Wang et al. , 2008) reformulated in the finite-volume language. An immediate advantage of FV discretization as compared to the CG FE one of FESOM is the simplicity of computations of the vertical velocity and hydrostatic pressure and the implicit integration of vertical diffusion and viscosity.

A triangular A-grid, similarly to a regular quadrilateral one, may suffer from pressure noise (elevation noise in hydrostatic codes). Its formal reason is the null space of the discretized gradient operator. Despite the true null space is present very rarely on meshes of variable resolution, the pressure noise is generally observed if the geopotential (z) vertical coordinate is used for the reasons mentioned above. In this respect the situation resembles that on regular B-grids (see, e. g., Killworth et al. (1991)), but the problem is more expressed on triangular A-grids and stabilization is generally necessary. Its basic idea is close to the recipe for B-grids by Killworth et al. (1991), but the implementation is different, as we seek a way that preserves the volume balance.

Our presentation of A-grid setup starts from the case without stabilization, shared except for detail with the quasi-B-grid setup, and is complemented with the implementation of stabilization.

3.1. Unstabilized solution algorithm

The horizontal momentum equation is discretized with respect to time as

$$\mathbf{u}^{k+1} - \mathbf{u}^k + g\Delta t\nabla(\theta\eta^{k+1} + (1-\theta)\eta^k) = \Delta t\mathbf{R}^{k+1/2}, \tag{1}$$

where

$$\mathbf{R} = -\nabla p - \nabla\cdot(\mathbf{uu}) - \partial_z(w\mathbf{u}) - \mathbf{f}\times\mathbf{u} + \nabla\cdot\sigma + \partial_z(A_v\partial_z\mathbf{u})$$

is the right hand side (rhs) vector. Here k labels time steps of length Δt, the rhs is estimated at mid-step with an appropriate explicit algorithm, e. g., the second or third order Adams-Bashforth method (the implicit stepping of vertical viscosity introduces modifications mentioned below). The rest of notation is standard: $\mathbf{u} = (u, v)$ is the horizontal velocity, $\mathbf{v} = (\mathbf{u}, w)$ the full 3D velocity, $\mathbf{f}$ the Coriolis vector, η the elevation, $p = \int_z^0 g\rho dz/\rho_0$ the normalized pressure due to fluid below $z = 0$, g the gravity acceleration, ρ the density and ρ_0 its reference value, A_v the vertical viscosity coefficient, θ the implicitness parameter, and $\nabla = (\partial_x, \partial_y)$. The horizontal viscosity is given in terms of viscous stress tensor σ with components $\sigma_{\alpha\beta} = 2A_h(e_{\alpha\beta} - (1/2)\delta_{\alpha\beta}e_{ll})$, where A_h is the horizontal viscosity coefficient, α, β and l are x or y, $\delta_{\alpha\beta}$ is the Kronecker tensor, $e_{\alpha\beta} = (1/2)(\partial_\alpha u_\beta + \partial_\beta u_\alpha)$ is the symmetrized tensor of horizontal velocity derivatives, and summation is implied over repeating indices.

We split the momentum equation (1) into a predictor step,

$$\mathbf{u}^* - \mathbf{u}^k = \Delta t\mathbf{R}^{k+1/2} - g\Delta t\nabla\eta^k, \tag{2}$$

and the corrector step,

$$\mathbf{u}^{k+1} - \mathbf{u}^* = -g\Delta t\theta(\eta^{k+1} - \eta^k). \tag{3}$$

The predictor velocity $\mathbf{u}^*$ can immediately be determined based on values from the previous time step, but the full velocity cannot, because the elevation on the new time level is not known.

In order to find it write first the elevation (vertically integrated continuity) equation

$$\eta^{k+1} - \eta^k = -\Delta t\nabla \cdot \int_{-H}^{0} (\alpha\mathbf{u}^{k+1} + (1-\alpha)\mathbf{u}^k)dz, \tag{4}$$

and insert $\mathbf{u}^{k+1}$ expressed from (3) to obtain an equation containing only the elevation. Here α is the implicitness parameter in the elevation equation. The approximation of linear free surface (zero upper limit in the integral) is used here for simplicity.

However, to be consistent on the discrete level, the substitution has to be made after discretizing equations in space. We will now explain how to do it.

Equations (2), (3) and (4) are integrated over control volumes. By virtue of Gauss theorem their flux divergence terms reduce to sums of fluxes through the faces of control volumes. On an A-grid the momentum advection term becomes

$$\int_{ni} (\nabla\cdot(\mathbf{uu}) + \partial_z(w\mathbf{u}))d\Omega = w_{ni}\mathbf{u}_{(n-1/2)i}A_{ni} - w_{(n+1)i}\mathbf{u}_{(n+1/2)i}A_{(n+1)i} + \sum_s \mathbf{n}\mathbf{u}_s\mathbf{u}_s l_s h_n.$$

Here $h_n = \overline{z}_n - \overline{z}_{n+1}$ is the layer thickness, the sum is over the segments s (faces in reality, but the surface edge/segment structure is used to address them) building the boundary of the control cell i, $\mathbf{n}_s$ are their outer normals, l_s are the segment lengths, $\mathbf{u}_s$, $\mathbf{u}_{(n-1/2)i}$ and $\mathbf{u}_{(n+1/2)i}$ are, respectively, the velocity estimates on segment s and the top and bottom faces. Similar approach is used to compute all other fluxes, with the difference that incomplete prisms are taken into account for scalar quantities. In all cases appropriate estimates of the advected quantities have to be supplied.

As an aside note that the convenience of FV approach hinges on using the edge structure to assemble sums of horizontal fluxes. For example, returning to the momentum advection, the contribution from edge j and layer n into the control volume around the first node i_1 of edge j is

$$((\mathbf{S}_j(1) - \mathbf{S}_j(2)) \times \mathbf{u}_{nj}) \cdot \mathbf{e}_z\mathbf{u}_{nj}h_n.$$

Here $\mathbf{u}_{nj} = (1/2)(\mathbf{u}_{ni_1} + \mathbf{u}_{ni_2})$ is the velocity estimate at edge j (both segments associated with edge use the same edge velocity), $\mathbf{e}_z$ the unit vertical vector, i_2 the second node of edge j, and the contribution to the control volume around i_2 differs in sign.

We employ centered estimate of velocity at mid-edges in computations of volume flux. This, in fact, defines the discretized divergence operator.

Computation of discretized gradient operator requires a comment. Following the edge scheme, the contribution to the area-integrated pressure gradient at node i_1 of edge j in layer n is

$$(1/2)(\mathbf{S}_j(1) - \mathbf{S}_j(2)) \times \mathbf{e}_z (p_{ni_1} + p_{ni_2}) h_n.$$

It is taken with opposite sign for the other node.

Alternatively, one may follow the FE way, first computing gradients on elements (triangular prisms) and then combining element-area-weighted gradients to get nodal quantities,

$$(\nabla p)_{ni} A_{ni} = \sum_t (\nabla p)_{nt} A_t / 3,$$

where t lists neighboring triangles, and A_t is the area of triangle t. Elemental gradients are computed by three nodal values assuming linear representation.

Because of stabilization (explained further) we will need gradients on nodes and elements, and the second form becomes more convenient. Although the two implementations of nodal gradient operator are identical in planar geometry (and prove to be minus transpose of the divergence operator), only the edge implementation preserves this property on A-grids in spherical geometry. We admit the incurring small inconsistency.

Written in terms of discretized variables, equations (2-4) take the form

$$\mathbf{u}^*_{ni} - \mathbf{u}^k_{ni} = \Delta t \mathbf{R}^{k+1/2}_{ni} - g\Delta t \sum_j G^n_{ij} \eta^k_j, \tag{5}$$

$$\mathbf{u}^{k+1}_{ni} - \mathbf{u}^*_{ni} = -\Delta t \theta g \sum_j G^n_{ij} (\eta^{k+1}_j - \eta^k_j), \tag{6}$$

and

$$\eta^{k+1}_i - \eta^k_i = -\Delta t \sum_{n,j} D^n_{ij} (\alpha \mathbf{u}^{k+1}_{nj} + (1-\alpha)\mathbf{u}^k_{nj}) h_n. \tag{7}$$

Here we introduced the gradient G^n_{ij} and divergence D^n_{ij} operator matrices for layer n for brevity. The gradient operator is defined at nodes and acts on elevation on neighboring nodes. The divergence operator is defined at nodes too, but acts on nodal velocities. Index j lists all neighbors of node i at layer n, and n lists all layers down to $N^{max}_i - 1$.

Substituting $\mathbf{u}^{k+1}_{ni}$ expressed from (6) into (7), one gets the equation governing the elevation

$$\eta^{k+1}_i - \eta^k_i - g\alpha\theta(\Delta t)^2 \sum_{n,j} D^n_{ij} h_n \sum_l G^n_{jl}(\eta^{k+1}_l - \eta^k_l) = \Delta t R_\eta, \qquad (8)$$

where

$$R_\eta = -\sum_{n,j} D^n_{ij}(\alpha \mathbf{u}^*_{nj} + (1-\alpha)\mathbf{u}^k_{nj})h_n.$$

The predictor velocity is estimated first, and equation (8) is then solved for the elevation. The velocity is corrected afterwards by (6). The concern with pressure (elevation) noise on A-grids is explained by the fact that G^n_{ij} possesses a null-space in certain cases. The operator occurring in (8) is obtained by summing over layers, and it is thus improbable that it will be rank deficient on variable topography. Ironically, the pressure noise is the strongest just in such cases and is seldom seen on flat bottom. It is thus enforced through the stepwise z-coordinate bottom and the structure of differential operator in (8) which connects node i not only with neighboring nodes, but also with their neighbors. The stencil of this operator, written more concisely as $\mathrm{H} = \sum_n \mathrm{D}^n \mathrm{G}^n h_n$ turns out to be too wide to effectively penalize local discontinuities. The operator is the depth-weighted Laplacian, so the idea of stabilization is to replace it partly or fully with the Laplacian defined on immediate neighborhood stencil as discussed further.

When the vertical viscosity is large, it is treated implicitly. In that case $\partial_z A_v \partial_z \mathbf{u}$ is included on the left hand side of (2) and hence (5), while the corrector equations are left without changes. The second-order time accuracy is formally retained because the right hand side of predictor equation (5) includes the estimate of elevation gradient at time level k. The velocity nodes become vertically connected in the predictor equation. A three-diagonal system of linear equations is solved for each horizontal location to disentangle them. In contrast, a full 3D system must be solved for CG FE case because of existing horizontal connections.

3.2. Stabilization

The idea of stabilization is borrowed from FESOM (see Wang et al. (2008) and Danilov et al. (2008)). We modify the predictor and corrector steps in the following way. The predictor step becomes

$$\mathbf{u}^*_{ni} - \mathbf{u}^k_{ni} = \Delta t \mathbf{R}^{k+1/2}_{ni} - g\gamma\Delta t \sum_j G^n_{ij}\eta^k_j, \tag{9}$$

i.e. $\mathbf{u}^*$ is now slightly offset (for γ close, but less than 1) from a 'good' prediction (j here lists neighboring nodes). This difference is compensated in the correction step, but in the space of velocities defined at centroids,

$$\tilde{\mathbf{u}}^{k+1}_{ni} = \mathbf{u}^*_{ni}, \tag{10}$$

$$\overline{\mathbf{u}}^{k+1}_{nt} = -\Delta t\theta g \sum_j \overline{G}^n_{tj}(\eta^{k+1}_j - \eta^k_j) + \Delta t(\gamma - 1)g \sum_j \overline{G}^n_{tj}\eta^k_j, \tag{11}$$

and j here indexes nodes of triangle t. Let us explain this notation. In the second case the velocity is computed at centroids t instead of nodes, and the operator $\overline{G}^n_{tj}$ returns elemental gradients. This is the composite representation of the horizontal velocity, with the largest part (tilde) in the nodal space and the correction (overline) in the elemental space. Although their sum is undefined, the volume or tracer fluxes driven by them can be added. We therefore demand that the continuity be satisfied by the velocity field in the composite representation. The volume flux through vertical faces of control volumes is, for every face, the sum of two contributions, one from the nodal velocity part at mid-edge locations ($\tilde{\mathbf{u}}$) and the other one, from the elemental part at centroids ($\overline{\mathbf{u}}$). Technically the modification reduces to just summing both velocity contributions for each face.

When the elevation η^{k+1} at a new time step is found, both (10) and (11) are known and are used to compute the vertical velocity and advect the scalars. This ensures internal consistency and warrants conservation.

This modification replaces the operator H with $\mathrm{L} = \sum_n \overline{\mathrm{D}}^n \overline{\mathrm{G}}^n h_n$, where $\overline{\mathrm{D}}^n$ is the divergence operator complementary to $\overline{\mathrm{G}}^n$ (acting on velocities at centroids). In contrast to H, L is computed on the nearest neighborhood stencil. At the end of full model time step, when tracers are already updated, the velocity $\overline{\mathbf{u}}^{k+1}$ is projected to nodal locations, and one recovers full nodal $\mathbf{u}^{k+1}$. It is only used to compute the rhs of momentum equations on the next time step.

In practice, $\gamma = 0.97$ is sufficient in most cases. The role of small deviation from one becomes transparent if we consider a quasistationary limit when $\eta^{k+1} \approx \eta^k$. It is only this difference which keeps $\mathbf{u}^*$ slightly offset from $\mathbf{u}^{k+1}$, so that the stabilization continues to work.

On the positive side, the operator part in the resulting equation on elevation contains only L. This reduces the CPU time needed to solve for the elevation (the number of nonzero elements in rows of L is more than twice smaller than in rows of H). Similarly to the nonstabilized case, the implicit treatment of vertical viscosity can be added to the predictor step, because the difference between $\mathbf{u}^*$ and $\mathbf{u}^{k+1}$ remains small.

3.3. Vertical velocity, pressure

Computation of vertical velocity and hydrostatic pressure follow the standard implementation of hydrostatic models. Here the FV method offers major advantages over the CG FE approach because horizontal connections of CG formulation are absent.

To ensure consistency between w and η the horizontal volume fluxes are accounted in the same way as for η, using the composite representation of velocity. The computation proceeds upward from the bottom at $n = N_i^{max}$ where $w_{ni} = 0$ (recall that w is at full levels) by collecting volume fluxes through the vertical walls of control volumes:

$$A_{(n-1)i} w_{(n-1)i} = A_{ni} w_{ni} + \sum_s \mathbf{n}_s \mathbf{u}_{(n-1)s} l_s h_{n-1},$$

where s implies summation over water segments bounding the control cell i in layer $n-1$, $\mathbf{u}_{ns} = \overline{\mathbf{u}}_{nt}^{k+1} + \tilde{\mathbf{u}}_{nj}^{k+1}$ with j and t indexing the edge and triangle associated with segment s, and the edge value of nodal velocity field is obtained by averaging over the edge nodes, $\tilde{\mathbf{u}}_{nj}^{k+1} = (1/2)(\tilde{\mathbf{u}}_{ni_1}^{k+1} + \tilde{\mathbf{u}}_{ni_2}^{k+1})$.

Computations of pressure p begin from the unperturbed surface by taking $p_{1i} = -g\rho_{1i} Z_1/\rho_0$ (atmospheric pressure can be added to this value if needed). Pressure in the layer $n > 1$ is obtained as $p_{ni} = g\rho_{(n-1)i}(Z_{n-1} - \overline{z}_n)/\rho_0 + g\rho_{ni}(\overline{z}_n - Z_n)/\rho_0 + p_{(n-1)i}$.

3.4. Temperature and salinity

We use asynchronous time stepping assuming that the velocity time step is offset by $\Delta t/2$ from that of temperature and salinity. As a result, velocity is now centered for a time step between k and $k+1$ for T and S (time is incremented as $t = \Delta t(1/2+k)$ in tracer equations). The transport (advection-diffusion) equations are discretized by integrating over control volumes and expressing the flux divergence in terms of fluxes leaving the volume. The horizontal velocity in the advection term is taken in the composite form, as for w above, to maintain consistency with the volume fluxes. The contribution from layer n and edge j in $(\int \nabla(\mathbf{u}T)d\Omega)_{ni_1}$ becomes

$$(\mathbf{u}_{ns} \times \mathbf{S}_j(1)) \cdot \mathbf{e}_z T_{ns} h_n,$$

from the left segment, and similarly from the right, but with the minus sign. It remains to provide an estimate of tracer quantity T_{ns} at segments. This step relies on reconstructions of either temperature field or its gradients. Several advection schemes exemplifying different approaches have been implemented. Here we just sketch them, their details will be reported elsewhere.

3.4.1. Methods based on tracer reconstruction

If $\mathcal{T}_{ni}(x,y) = T_0 + a_x x + a_y y + a_{xx}x^2 + a_{xy}xy + a_{yy}y^2 + ...$ is a horizontal reconstruction for control volume (n,i), it should satisfy the constraint $\int_{ni} \mathcal{T}_{ni} d\Omega = T_{ni}A_{ni}$ (otherwise time derivative will include information on neighbors). Here x, y are components of vector $\mathbf{r}_i$ drawn from vertex i. Relatedly, this statement is taken into account as a strong constraint. Together with the weak constraint $\sum_{j(i)} |\int_{nj} \mathcal{T}_{ni} d\Omega - Tnj|^2 = \min$ it is used to compute the coefficients of reconstruction (see, e. g., Ollivier-Gooh and Van Altena (2002), and Ouvrard et al. (2009)). Here $j(i)$ is the list of vertices close to i. A recent implementation of the second-order and fourth-order reconstruction schemes on hexagonal meshes is presented in Skamarock and Menchaca (2010). On median-dual control volumes the nearest neighbors are sufficient for the first or (on good quality meshes) second order reconstruction. A much simpler linear reconstruction $\mathcal{T}_{ni}(x,y) = T_{ni} + (\nabla T)_{ni}\mathbf{r}_i$ is sometimes used, but it is biased if the mesh is not uniform. The linear reconstruction upwind (LRU) scheme (similar to that used in FVCOM) and the Miura scheme (Miura (2007)), as implemented by us, are based on biased linear reconstruction. They are least expensive in terms of CPU time and provide second-order accuracy on quasi-uniform meshes. The LRU scheme is stepped with the second-order Adams Bashforth method and the Miura scheme is the direct time-space one. They are augmented by the quadratic reconstruction upwind direct space-time scheme (QRU) which uses the reconstruction algorithm of Ouvrard et al. (2009).

The Miura scheme was originally formulated for hexagonal elements, but it is not specific to the element type. Its idea is to trace the fluid volume that will be advected through a given vertical face (segment) over time interval Δt, and associate T_{ns} with the mean T over this volume. It is just the estimate at the centroid of this volume. Four quadrature points are used for the QRU which exploits the same idea.

3.4.2. Method based on gradient reconstruction

The technology suggested by Abalakin et al. (2002) mimics the MUSCL approach and seeks to reconstruct the gradients by combining the centered estimate with estimates from upwind triangles. The approach warrants second order on general meshes and becomes higher order if meshes are uniform. We write

$$T_{ns} = T_{ni_1} + (\nabla T)_{ns}\mathbf{L}_j/2$$

or

$$T_{ns} = T_{ni_2} - (\nabla T)_{ns}\mathbf{L}_j/2,$$

depending on which node is upwind. Further,

$$(\nabla T)_{ns}\mathbf{L}_j = (1-\beta)(T_{ni_2} - T_{ni_1}) + \beta(\nabla T)^u_{nj}\mathbf{L}_j,$$

where $(\nabla T)^u_{ns}$ is the gradient on triangle that is upwind to edge j, and β is a parameter. $\beta = 1/3$ ensures the third-order behavior on uniform meshes. The order can be raised to fourth if the upwind estimate for T_{ns} is replaced by the centered one. Even higher orders are possible, but estimate of gradient becomes more cumbersome. The third/fourth order scheme is similar to that suggested by Skamarock and Gassmann (2011), with the difference that their formulation is suited for the Barth control volumes (obtained by connecting circumcenters), and that by Abalakin et al. (2002) is valid also for median-dual control volumes. The third-order scheme is implemented in the code (abbreviated with MUSCL further). It is also augmented with the FCT algorithm (MUSCL-FCT). In that case the first-order upwind is used as a low-order method. In parallel implementation these schemes require an additional layer of halo elements, which may influence scalability. Without the FCT limiting the scheme is less expensive in terms of CPU time than the QRU scheme. With the FCT limiting, it becomes more expensive.

Our two-dimensional tests show that the QRU, MUSCL and MUSCL-FCT are less dissipative than the Miura scheme. We expect that the performance in terms of convergence is similar to that reported by Skamarock and Menchaca (2010) and Skamarock and Gassmann (2011) for hexagonal meshes (the placement of scalar variables is the same in their and our cases).

3.4.3. Vertical advection

Quadratic upwind reconstruction is used in most cases in the vertical direction. We replace it with the linear reconstruction at the surface and bottom when necessary. In the case of Miura and QRU schemes the estimate is performed at locations shifted by $-w_{ni}\Delta t/2$ from $z = \overline{z}_n$, in other cases — directly at $\overline{z}_n$. Quadratic reconstruction is known to be suboptimal on uniform meshes (a linear combination of quadratic and linear reconstruction can lead to a more accurate estimate of flux divergence, see e. g. Webb et al. (1998)), but we keep it here because in practice the vertical discretization is seldom uniform.

3.4.4. Diffusive fluxes

Computation of diffusive fluxes needs some generalization in the case of diffusivity tensors, which we skip here for brevity.

When a vertical mixing scheme is operating, the vertical diffusion is treated implicitly as a separate substep. We split the full time step for the temperature T (salinity is treated in the same way)

$$T^{k+1} - T^k = \Delta t \partial_z K_v \partial_z T^{k+1} + \Delta t R_T^{k+1/2}$$

into an explicit

$$T^* - T^k = \Delta t R_T^{k+1/2}$$

and implicit

$$T^{k+1} - \Delta t \partial_z K_v \partial_z T^{k+1} = T^*$$

parts. Here K_v is the vertical diffusivity coefficient, and R_T takes into account advection and horizontal diffusion. The implicit part reduces, for every surface location, to a three-diagonal matrix system for $N_L - 1$ or less vertically aligned nodes, which is easily solved. Notice, that by adding explicit and implicit parts one recovers the original equation, so the split does not introduce errors. The second-order accuracy in time will be achieved if vertical diffusion is treated semi-implicitly. We do not do it because K_v is supplied by parameterization and its accuracy is unknown. The test cases reported below use the vertical mixing scheme by Pacanowsky and Philander (1981).

4. Cell–vertex (quasi-B-grid) setup

This setup uses the same placement of variables as FVCOM (Chen et al. , 2003), but is formulated on z-levels and differs in the implementation of time stepping, advection and dissipation. Distinct from the A-grid case, the horizontal velocity is now at centroids (in the horizontal plane) and triangular prisms serve as control volumes for the momentum. The velocity points are always inside full control volumes so that both no-slip and free-slip boundary conditions are supported. This and the absence of pressure modes are the major advantages of quasi-B-grids. Additionally, the geostrophic balance can be maintained on the discrete level.

Note that there is an almost exact analog of this variable arrangement on hexagons, called the ZM grid (Ringler and Randall , 2002a,b). The difference lies in using scalar control volumes obtained by connecting circumcenters instead of median-dual ones.

The main practical difficulty of working with quasi-B-grids is their large velocity space. It supports spurious modes that correspond to inertial oscillations at the Coriolis frequency (Le Roux et al. , 2007). The modes prove to be a minor issue on their own, as any viscous dissipation will damp them. Much more annoying is the generation of small scales through the advection of momentum in typical eddying regimes encountered in large-scale ocean modeling.

The point of concern here has already been raised by Ringler and Randall (2002b) who showed that the velocity representation on ZM grids resolves wave numbers that are absent in the representation of scalar fields. The small-scale part of the horizontal velocity field may alias the field of horizontal divergence computed at scalar locations. Correspondingly, the small-scale components in the horizontal velocity field have to be effectively filtered. We stress that the extent to which they hamper the performance depends on applications, but noise in the vertical velocity is often seen in eddy-resolving simulations. Filtering can be implemented either through viscous operators or the treatment of momentum advection.

In summary, the success of using quasi-B-grid FV discretization for simulating large-scale ocean circulation relies on tuning viscosity and momentum advection. Below we explain how to do it.

Because the quasi-B-grids do not suffer from pressure modes, the time stepping of dynamical part is organized as for unstabilized A-grids with the difference that operators $\overline{G}_{ti}^{n}$ and $\overline{D}_{it}^{n}$ appear now in equations (5-7), with

i and t being indices of nodes and elements respectively, and momentum equations are formulated at elements. Discretization of momentum advection and viscosity is different and is discussed further. Since the arrangement of vertical velocity, elevation, pressure, temperature and salinity is shared with the case of triangular A-grid, this part of code follows the A-grid setup, with obvious modifications to account for the horizontal velocities on elements.

4.1. Linear reconstruction and viscosity operator

We need horizontal gradients of horizontal velocity to perform its linear reconstruction and estimate viscous fluxes. This is done by the least square fit of four velocities (in the control volume and its three neighbors). The reconstruction coefficients are stored for each triangle.

Some of neighbors can be absent in deep layers on z-topography. Instead of modifying the scheme we employ the concept of ghost element across the respective face and compute velocity there either as $\mathbf{u}_{nj} = -\mathbf{u}_{nt}$ for no-slip, or reflect only the component normal to the edge for the free-slip, $\mathbf{u}_{nj} = -\mathbf{u}_{nt} + 2(\mathbf{u}_{nt}\mathbf{L}_{jt})\mathbf{L}_{jt}/|\mathbf{L}_{jt}|^2$. Here j is the index of ghost triangle, and $\mathbf{L}_{jt}$ is the edge vector associated with the edge between triangles j and t. In this case the gradient coefficients can be used through the whole depth. On lateral walls the ghost triangles are physically absent, and their centroids are assumed to be mirror images of the centroid of t with respect to the boundary edges.

Since velocity gradients are available, the viscous stress tensor is known on elements too. The viscous flux at the vertical faces is computed as average of estimates from the two elements sharing the face. No averaging is performed if the face is at the rigid wall.

The biharmonic diffusivity operator is build by repeating twice the procedures involved in the construction of the harmonic (Laplacian) viscosity. When $\nabla\sigma$ is available, we apply the same least square fit procedure as used for velocities to find its gradients, and then compute the divergence of 'biharmonic stresses'.

Scaling the viscosity coefficients with areas (as $A_t^{1/2}$ and $A_t^{3/2}$ for harmonic and biharmonic viscosities respectively) is sufficient to stabilize flows on coarse meshes. It frequently fails on fine meshes in configurations with strong baroclinicity, which tend to develop a grid-scale mode in the vertical velocity field. The idea is to select the coefficient A_h of harmonic horizontal viscosity so that it penalizes the places where the vertical velocity is changing too sharply (which indicates that small-scale noise in the horizontal velocity field is developing). It is well served by the modified Leith viscosity used in MITgcm (see Fox-Kemper and Menemenlis (2008)). We select

$$A_h = C_{ML}|\nabla\nabla\cdot\mathbf{u}|_{nt}A_t^{3/2},$$

where A_t is the area of respective triangle t, and C_{ML} is the constant of modified Leith parameterization. Our implementation uses the w field because $(w_{ni} - A_{(n+1)i}w_{(n+1)i}/A_{ni})/h_n$ provides the estimate of divergence at node i in layer n. Its gradient on triangles is obtained by applying the rule used for scalar quantities. Taking C_{ML} from 0.25 to 1 typically helps to maintain the code stability by enforcing smoothness of w. We also keep the Smagorinsky viscosity as an additional option. Its implementation is standard (velocity gradients are known) and is not repeated here.

4.2. Momentum advection

We describe here several discretizations of momentum advection. They include the linear upwind reconstruction scheme on velocity control volumes (MA), the scheme based on velocity reprojection (MB), the scheme based on scalar control volumes (MC) and the vector-invariant scheme (MD). In a general case, they still need the modified Leith viscosity for stable performance, but the scheme MC is least demanding.

4.2.1. Linear upwind reconstruction

The MA scheme is, perhaps, the most straightforward way to proceed and corresponds to that of Chen et al. (2003). Having the horizontal velocity gradients on triangles t_1 and t_2 of edge j one can linearly reconstruct the horizontal velocity to the mid-edge position in the horizontal plane:

$$\mathbf{u}_{nj,l} = \mathbf{u}_{nt_1} - \mathbf{S}_j(1)\cdot(\nabla\mathbf{u})_{nt_1},$$

on the left triangle (t_1) and

$$\mathbf{u}_{nj,r} = \mathbf{u}_{nt_2} - \mathbf{S}_j(2)\cdot(\nabla\mathbf{u})_{nt_2}$$

on the right one (t_2). For each face, an estimate, symmetrized over two volumes sharing the face is formed, $\mathbf{u}_{nj} = (1/2)(\mathbf{u}_{nj,l} + \mathbf{u}_{nj,r})$, and used to compute the normal velocity on the face. Depending on its sign, the linear reconstruction from the upwind control volume is used to compute the horizontal momentum flux,

$$\int_{nt} \nabla \cdot (\mathbf{uu}) d\Omega = \sum_j \mathbf{u}_{nj}\mathbf{n}_j |\mathbf{L_j}| (\mathbf{u}_{nj} + (1/2)\mathrm{sign}(\mathbf{u}_{nj}\mathbf{n}_j)(\mathbf{u}_{nj,l} - \mathbf{u}_{nj,r}))h_n$$

Here j indexes three edges of triangle t, and the normal is directed to the right triangle of edge j.

Vertical fluxes of horizontal momentum are computed using quadratic upwind reconstruction of horizontal velocity.

Although this scheme introduces dissipation, it is insufficient to effectively suppress small scales, and additional viscous damping is necessary. This results in low levels of turbulent kinetic energy in experiments on baroclinic instability reported in section 5.1.

4.2.2. Momentum advection reprojection

There are two ways of discretizing the flux form of momentum advection that are simultaneously less dissipative and provide certain filtering, which is a desirable feature. The first one (MB) introduces a nodal velocity field as an element-area-weighted estimate of elemental velocities:

$$A_{ni}\mathbf{u}_{ni} = \sum_t \mathbf{u}_{nt} A_t/3,$$

where t lists neighboring triangles of node i. The next step uses the nodal velocities to estimate the momentum fluxes through the faces of velocity control volumes:

$$(\int \nabla \cdot (\mathbf{uu}) d\Omega)_{nt} = \sum_j \mathbf{u}_{nj} \cdot \mathbf{n}_j \mathbf{u}_{nj} |\mathbf{L}_j| h_n,$$

where $\mathbf{u}_{nj} = (\mathbf{u}_{ni_1} + \mathbf{u}_{ni_2})/2$ is the mean velocity on the face associated with layer n and edge j, i_1 and i_2 are the nodes of edge j and summation is over three edges (faces) of triangle t.

The second way (MC) is seemingly more consistent. One selects scalar control volumes to compute full (horizontal and vertical) momentum advection at nodal locations. In the same manner as on A-grid, the contribution of layer n and edge j to $(\int \nabla \cdot (\mathbf{uu}) d\Omega)_{ni_1}$ at the edge node i_1 becomes

$$(\mathbf{u}_{nt_1}(\mathbf{S}_j(1) \times \mathbf{u}_{nt_1}) - \mathbf{u}_{nt_2}(\mathbf{S}_j(2) \times \mathbf{u}_{nt_2}))h_n.$$

It enters with opposite sign to the control volume around node i_2. Computations of the vertical advection use nodal estimate of horizontal velocities and quadratic upwind reconstruction. On the next step, the nodal estimates of momentum advection are averaged to elements. We employ this scheme most frequently.

4.2.3. Vector-invariant form

There is one more possibility (MD) that implies some horizontal smoothing too. It comes from the vector-invariant form of momentum advection:

$$(\mathbf{u}\cdot\nabla)\mathbf{u} + w\partial_z\mathbf{u} = \omega\mathbf{e}_z \times \mathbf{u} + (1/2)\nabla\mathbf{u}^2 + w\partial_z\mathbf{u},$$

where $\omega = \operatorname{curl}\mathbf{u}$. The relative vorticity ω has to be defined at nodal locations where it can be estimated by making use of Stokes' theorem and computing circulation along the boundary of scalar control volume. Then a value of ω averaged to centroids is used to estimate the first term in the formula above. We need the kinetic energy $K = \mathbf{u}^2/2$ at vertices to obtain its gradient on elements. The rule of computing it is dictated by the need to preserve the kinetic energy balance. It can be shown that the rule

$$K_{ni}A_{ni} = \sum_t \mathbf{u}_{nt}^2 A_t/3,$$

is the consistent one (t lists neighboring triangles of node i). Moreover, the energy conservation also imposes limitations on the implementation of the vertical part. The conservation is warranted if we write $w\partial_z\mathbf{u} = \partial_z(w\mathbf{u}) - \mathbf{u}\partial_z w$. The first term here is computed as the difference of fluxes through the top and bottom faces of triangular prism nt and in the second one $\partial_z w$ is taken as the mean on triangle t, $\sum_{i(t)}(\partial_z w)_{ni}/3$. Centered approximation for the horizontal velocity on the top and bottom face is used. Although we do not show it here, the vector-invariant discretization on median-dual control volumes shares the properties of discretization in Ringler and Randall (2002a) (energy and enstrophy conservation for the shallow water equations).

The vector invariant form is sensitive to observing the rules formulated above and is incompatible with upwinding in vertical fluxes.

4.2.4. Comments on momentum advection

Schemes MB and MC require the least explicit dissipation, followed by MD and then MA, in a baroclinic instability test reported further. They are therefore recommended. They, however, do not conserve energy. There are additional issues as well.

Although $(\mathbf{u}\nabla+w\partial_z)\mathbf{u} = \nabla\cdot(\mathbf{uu})+\partial_z(w\mathbf{u})$ in the continuous case because $\nabla\mathbf{u} + \partial_z w = 0$, this equality is violated in the discretized equations because $\nabla\mathbf{u} + \partial_z w = 0$ is valid only in a particular sense. This implies that the discretizations of vector invariant and flux forms of momentum advection are irreducible to each other. The differences between discretizations may lead to noticeable effects on the ocean circulation on large time scales, especially in the vicinity of topography (cf. Le Sommer et al. (2009)). One should be aware of this fact, its implications require a thorough study.

5. Performance comparison

Since the variable placements used here are not new, their general performance is well understood. In particular, Wang et al. (2008) and Danilov et al. (2008) present some test cases with FESOM, and Chen et al. (2003) with FVCOM, and there are numerous other publications which will not be discussed here. In general, because of similar scalar parts and filtering of momentum advection on quasi-B-grids one does not expect to see strong differences in their performance. We therefore focus on two cases that illustrate, to an extent, manifestations of 'geometrical' issues discussed above in situations relevant to large-scale modeling. They do not propose the metrics to judge on model results, but highlight the points we consider worth of attention.

The first one involves baroclinic instability in a zonally re-entrant channel. It highlights consequences of the large size of velocity space on quasi-B-grids. The other configuration deals with the circulation in the North Atlantic basin. It illustrates the impact of realistic topography represented with z levels, in which case the quasi-B-grids face less difficulties if properly tuned.

5.1. Baroclinic instability in a zonally re-entrant channel

The domain occupies a latitude belt between 30° N and 45° N and is 20 degrees long in zonal direction. The resolution is 1/6 by 1/7 degree and there are 24 levels spaced unevenly down to the depth of 1600 m. Triangulation is done by splitting quadrilaterals of original rectangular mesh into triangles. The initial state is characterized by linear meridional and vertical temperature gradients of -5×10^{-6} and 8.2×10^{-3} °C/m respectively, the largest surface temperature is 25°C and salinity is (and stays) uniform with

35 psu. Full nonlinear equation of state is used. The flow is forced by relaxing temperature to its initial distributions in 1.5 degree wide southern and northern relaxation zones. The relaxation coefficient decreases linearly from 1/(3 days) to zero within these zones. In all cases the background vertical viscosity and diffusivity are 10^{-3} and 10^{-5} m^2/s respectively. The Pacanowsky–Philander vertical mixing scheme with maximum diffusivity of 0.01 m^2/s is operating on temperature. The horizontal diffusivity is 30 m^2/s which is presumably below the implicit diffusivity introduced by the upwind transport schemes used here. The A-grid case is stable with $A_h = 100$ m^2/s (actual horizontal viscosity and diffusivity are scaled on each triangle with factor $(A/A_0)^{1/2}$, where scaling area $A_0 = 2 \times 10^8$ m^2). In the quasi-B-grid case dissipation should each time be carefully adjusted to fit the particular momentum advection scheme (see below). The bottom drag coefficient is $C_d = 0.0025$ in both cases.

A small sinusoidal perturbation of temperature is added to zonally uniform initial temperature distribution to trigger the baroclinic instability, which fully develops within the first model year. We performed multiple runs with different scalar advection schemes (A- and quasi-B-grids) and also momentum advection (quasi-B-grid) to identify their influence on the mean kinetic energy levels. Each case is integrated for at least three years. The basin-mean kinetic energy (dominated by the turbulent part) shows marked fluctuations, so that mean levels can be identified only approximately. In order to learn about the 'true' energy levels, reference simulations have been performed on a mesh with approximately doubled resolution (8.5 km) using the quasi-B-grid code with the least possible dissipation. They show fluctuations of smaller amplitude and give the mean reference kinetic energy of approximately 0.11 m^2/s.

The left and middle panels of Fig. 3 show, respectively, snapshots of elevation and temperature simulated on A-grid using the Miura advection and stabilization with $\gamma = 0.97$. The elevation pattern is free of pressure modes, while that of temperature shows filaments characteristic of well-developed baroclinic instability. The setup also runs without stabilization in this case (the bottom is flat) demonstrating very similar levels of kinetic energy and absence of pressure modes.

The right panel shows the temperature snapshot from quasi-B-grid simulations with MUSCL temperature advection and MC momentum advection. The temperature fronts are noticeably sharper compared to those of Miura scheme, which is indicative of smaller implicit dissipation.

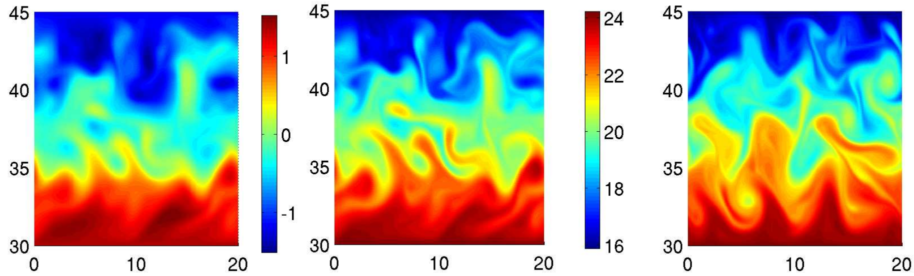

Figure 3: Snapshots of elevation (m) (left) and temperature (°C) (middle and right) (at approximately 100 m depth) in zonally reentrant channel. Left and middle panels: A-grid, the Miura advection; right: quasi-B-grid, MUSCL advection and MC momentum advection.

However, despite this and the fact that the QRU and MUSCL schemes are less dissipative in 2D tests than the Miura scheme (or the LRU scheme which is very similar in performance), we found no obvious increase in kinetic energy levels. This is also true of MUSCL-FCT scheme. We therefore do not consider the impact of these schemes on the energy level any further. The analysis of their other aspects is outside the scope of this paper.

The stabilization on A-grids introduces a bias in the energy transfer because of two representations for the horizontal velocity. We diagnose it as the difference between $\int \mathbf{u}\nabla p d\Omega$ (nodal velocity) and $-\int p\nabla\mathbf{u} d\Omega = \int p\partial_z w d\Omega$ (composite velocity) which makes up about 5% of the energy transfer on the mean. It is not negligible, but the effect on the kinetic energy cannot be distinguished on the background of natural fluctuations if one compares outputs of stabilized and unstabilized setups. In the case considered, $\int \mathbf{u}\nabla p d\Omega$ is always negative (the kinetic energy is supplied through the release of available potential energy which is replenished by the relaxation to 'climatology'), and the bias term does not change sign. It works to reduce the energy transfer.

For the quasi-B-grid we first consider two cases: (i) the momentum advection is computed on scalar control volumes (scheme MC above), and viscosity operator is biharmonic, with $A_{bh} = 0.8 \times 10^{10}$ m^4/s scaled as $(A/A_0)^{3/2}$; (ii) the momentum advection is on velocity control volumes (scheme MA) with biharmonic and modified Leith viscosities. Dissipation in (i) is at minimum compatible with stable performance. The case (ii) was first run with $A_{bh} = 3 \times 10^{10}$ m^2/s and $C_{mL} = 1$ for three years, and continued then with reduced dissipation ($A_{bh} = 10^{10}$ m^2/s and $C_{mL} = 0.5, 0.25$ and 0; the last two

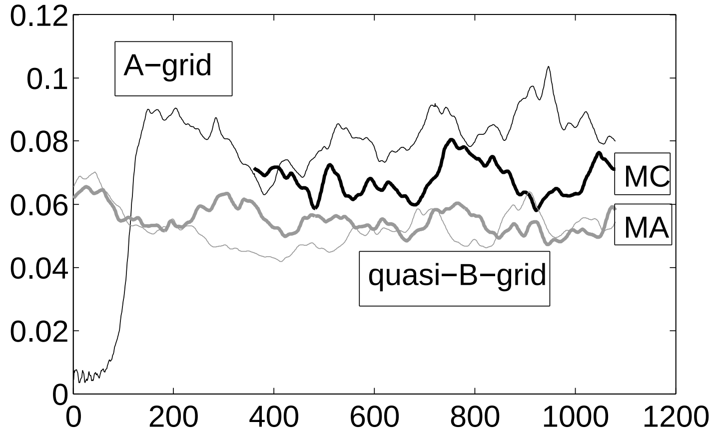

Figure 4: Doubled kinetic energy per unit mass (m^2/s^2) as a function of time (days) in channel experiments. The two gray curves correspond to MA momentum advection on quasi-B-grid with weak dissipation (thick) with $A_{bh} = 10^{10}$ m^2/s and $C_{mL} = 0.5$ and strong dissipation (thin) with $A_{bh} = 3.0 \times 10^{10}$ m^2/s and $C_{mL} = 1.0$. They show similar energy levels, pointing at the dominance of dissipation due to upwinding. Simulations with MC momentum advection (black thick curve) reach higher energy levels but even they are below the result for A-grid (thin black curve). Initial evolution phase is very similar in all cases and is retained only for A-grid. The reference value of 0.11 m^2/s is not achieved, but A-grid simulations are the closest to it.

variants are losing stability with time). Figure 4 illustrates that the case (ii) reaches lower energy level than case (i) (gray curves vs. thick black). It does not show strong sensitivity to the magnitude of dissipative coefficients, as can be concluded from the behavior of two gray curves for strong (thin) and weak (thick) dissipation in Fig. 4, which implies that dissipation is mostly set by upwinding in the MA scheme.

However, the presence of modified Leith viscosity is crucial, and if it is insufficient one sees the development of numerical noise well emphasized in patterns of vertical velocity, as illustrated by Fig. 5. Its bottom left panel represents a snapshot from case (ii) for $C_{mL} = 0.5$, which should be compared to a 'normal' pattern of case (i) shown in the upper left panel. Maxima and minima of vertical velocity are in fact an order of magnitude stronger in the lower left panel. The grid-scale band structure becomes even more expressed for smaller C_{mL} ending in unstable behavior. Schemes MB and MC of momentum advection work with $C_{mL} = 0$ in the channel case, but MB requires slightly higher biharmonic viscosity ($A_{bh} = 1.0 \times 10^{10}$ m^4/s) than MC (we do not illustrate it here).

Two right panels compare the runs with MC (top) and MD ($C_{mL} = 0.5$, bottom) momentum advection and MUSCL temperature advection. The temperature distribution has sharper fronts in this case, so the w pattern is less smooth. Despite non-zero C_{mL}, the MD case shows some tendency to developing a grid-scale pattern. Apart from that, it reproduces the same energy levels as MC.

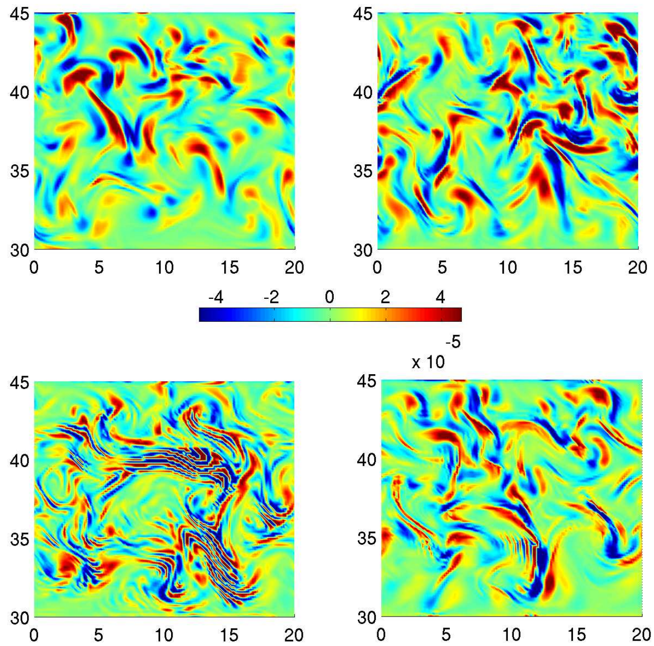

Figure 5: Snapshots of vertical velocity (m/s) at approximately 100 m depth in quasi-B-grid runs with different advection of momentum. Left column: MC scheme (top); MA scheme, $C_{mL} = 0.5$ (bottom). Right column: MC (top); MD, $C_{mL} = 0.5$ (bottom). Temperature advection is with the Miura (left column) or MUSCL (right column) schemes.

This noise is the main difficulty of the quasi-B-grid approach in eddy resolving regimes. In fact the grid-scale pattern in w just visualizes a mode in the horizontal velocity field. It manifests itself through fluctuations of direction of neighboring velocity vectors. While one may attribute its development to the vulnerability of the quasi-B-grid discretization to spurious inertial modes, it is invariably present only when the momentum advection is strong. It seems plausible to conclude that the problem is at least triggered by aliasing of the resolved dynamics through small scales. Indeed, schemes MB and MC are less susceptible to the noise because of explicit averaging (of velocity or the momentum advection). It remains to see why the MD scheme, which works on the same stencil as MB and MC still needs the modified Leith viscosity.

Note that even the simulations with the MC scheme on quasi-B-grid do not reach the kinetic energy level of A-grid simulations (see Fig. 4) with a rather high harmonic viscosity. Namely in an attempt to minimize dissipation we run the quasi-B-grid cases with biharmonic background viscosity, and use the harmonic one only as the modified Leith contribution. The w pattern of A-grid runs in channel flow is always smooth.

5.2. North-Atlantic configuration

The mesh employed here is fully unstructured and uses resolution of about 20 km over the Gulf Stream area and a part of Caribbean basin, and is about 100 km otherwise except for coastlines where the resolution is also refined. There are 26 vertical levels, with layer thickness from 10 m at the top to 500 m in deep ocean (with the deepest level at -5500 m). The bottom topography is derived from the ETOPO5 database averaged to a regular quarter degree mesh. Strong relaxation to climatology is used in buffer zones attached to open boundaries (the southern one at 28° S, the northern one at 80° N and the eastern one completing the north-east corner of the domain) and in the vicinity of Gibraltar. The surface forcing is implemented as relaxation to monthly mean temperature and salinity of the World Ocean Atlas 2001 (www.nodc.noaa.gov/OC5/WOA01/pr_woa01.htm), and wind forcing relies on monthly mean NCAR/NCEP reanalysis winds (Kalnay et al. (1996)) from 1990 on. The Miura advection scheme is used as most economical.

The intention here is only to demonstrate main practical difficulties of the A-grid setup seen in the presence of real topography. The A-grid code is run with stabilization ($\gamma = 0.97$), the background horizontal viscosity $A_h = 200$ m^2/s and horizontal diffusivity $K_h = 100$ m^2/s, both scaled as $(A/A_0)^{1/2}$. It develops rather strong equatorial currents within the first year of integration. In order to keep them in reasonable bounds the modified Leith viscosity is switched on with $C_{mL} = 0.5$ and additionally, the horizontal viscosity is multiplied with a factor linearly increasing from 1 to 2 in a 7 degree zone around the equator.

The momentum advection is computed on scalar volumes on quasi-B-grid. It uses the same biharmonic viscosity as in channel runs, the modified Leith viscosity is also added with $C_{mL} = 0.35$.

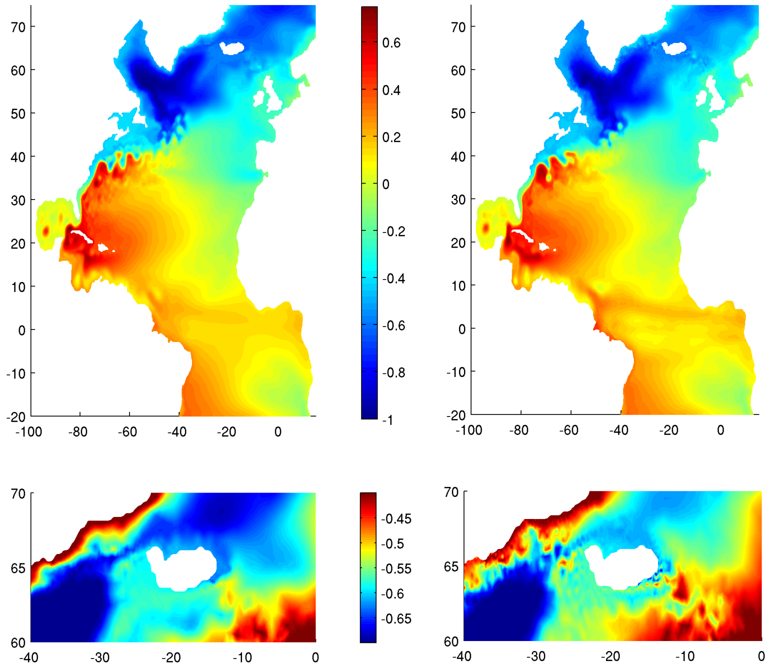

Figure 6: Snapshots of simulated elevation (m) in the North Atlantic on completing 1 year of integration in quasi-B-grid (left) and A-grid (right) setups. While the pattern is very similar in both cases, the A-grid develops noise in the shallow regions (the periphery of the Labrador Sea and the vicinity of Iceland; there are many other places along the western coast yet they cannot be discerned in the figure). Bottom panels zoom into the area around Iceland to visualize the noise on A-grid. In most cases it can be eliminated by refining the mesh.

Figure 6 compares instantaneous sea surface height fields after one year of integrations. They are similar in general, but differ in detail over the fine mesh part as dynamics there are to some extent stochastic. Both setups run stable, and we selected the output just after one year only to emphasize similarity which is less apparent at later time. Considering the elevation field in the vicinity of Iceland (see bottom panels) we note a pronounced noise on the A-grid. It is also present close to the coast in other areas where topography varies too fast for the mesh resolution used (it is hardly discernible without zooming-in). No problem of that kind is seen for the quasi-B-grid setup. Relatedly, the velocity field in the A-grid setup is noisy over such areas (not shown), which may affect dynamics of temperature and salinity if diffusion is insufficient to control their smoothness. The emergence of noise can also be attributed to dynamics being 'overspecified' by a too large number of no-slip boundary conditions imposed on vertical edges of z-coordinate bottom topography. The noise can slightly be reduced by increasing stabilization, but real improvement can only be achieved by increasing resolution locally or smoothing topography in such regions (note that noise is absent over the well-resolved Gulf Stream area). Although the potential danger of such situations on A-grids can be envisaged, the details are not known a priori, which in practice implies probing multiple meshes and topography implementations.

6. Discussion

Our intention here was to describe two unstructured mesh setups that can be applied for large-scale ocean modeling. Both use finite-volumes as the discretization ideology and share, up to some detail, the scalar part. The choice was partly motivated by the already existing practical applications. The A-grid setup derives from FESOM (Wang et al. (2008), Sidorenko et al. (2011)) and represents in effect, its finite volume reformulation which follows the ideas of Szmelter and Smolarkiewicz (2010). The other choice was inspired by the success of FVCOM (Chen et al. , 2003) and the understanding that this type of discretization is well suited to model geostrophically balanced flows.

Apart from that, the setups correspond to two configurations with balanced (A-grid) and unbalanced (quasi-B-grid) ratios of velocity to scalar degrees of freedom, which has implications as concerns the measures needed to maintain their stability.

There are two simple ideas behind this development. The first one is the numerical efficiency, and the second one is algorithmic simplicity in the hydrostatic case. The first one hinges on practical observation that FV codes are as a rule more numerically efficient than their FE counterparts, and our comparison with FESOM shows that indeed a speedup of 2 to 3 times is easily achieved for both approaches discussed here (it is difficult to be more precise as actual results depend on options used). The second one bears on the fact that the FV discretization enables solving for the hydrostatic pressure and vertical velocity in a natural way.

There is a comprehensive body of research on wave dispersion for various types of unstructured-mesh discretizations in the linearized shallow-water framework (see, i.e., Le Roux et al. (2007) and Le Roux and Pouliot (2008)). Although it is indispensable in guiding the preliminary choice, the actual problems of particular discretizations frequently show up on the stage of realistic setups. We demonstrate here that the triangular A-grids on large scales are sensitive to the details of stepwise bottom representation on z-coordinate meshes. The stepwise bottom, in essence, is the reason why stabilization is needed, but even the stabilized A-grid setups are prone of producing noisy elevation field over the regions with rough topography. The noise is triggered in most cases by the patchy structure of vertical velocity field in this case, which is partly emphasized through too many no-slip boundary conditions imposed on the horizontal velocity over the deep part (so that adjacent unconstrained velocities react in a noisy way). This issue is not a severe one, but annoying in practice because multiple (refined) meshes and topography representations have to be tried before a satisfactory solution is found. One may hypothesize that stabilization and topography-induced noise will be of less relevance on terrain-following meshes, and this remains to be seen.

The quasi-B-grid setup does not share this type of difficulty, but has the other one. Namely, because of its too large velocity space, it tends to create scales that are not maintained by other dynamics (see the analysis by Ringler and Randall (2002b)). Here the solution lies in tuning the dissipation and advection terms in the momentum equation, and we hope that the recipes described above are sufficient in most cases of practical relevance. Computation of momentum advection on scalar control volumes and subsequent averaging to centroids (MC scheme above) is arguably most helpful. It adds filtering which works well in combination with gentle biharmonic and/or modified-Leith viscosity. And yet, as we have seen from baroclinic channel experiments, the levels of turbulent kinetic energy stay lower than on the A-grid, which implies that the net dissipation is higher.

Note that similar difficulty (stemming from the large size of velocity space) was also reported for the horizontal velocity representation with nonconforming linear elements (Danilov et al. , 2008). The basis functions in this case are associated with edges, so that one gets an even larger velocity space. Once again, stable performance of momentum advection was an issue on that discretization and reprojection of advected velocities on linear continuous functions was solving the problem in practice.

Full consequences of momentum advection discretization require further studies as different implementations may lead to differences in the vorticity balance, especially in the vicinity of topography.

The finite-volume setups also benefit from a richer choice of advection schemes. Although we have not found significant effect from the high-order schemes described here (QRU, MUSCL, MUSCL-FCT) on the kinetic energy levels in the baroclinic instability tests, there are other aspects (like spurious diapycnal mixing) which remain to be studied.

There are arguments in favor of both, the A-grid and quasi-B-grid, setups, but the absence of stabilization makes the the latter a more consistent (yet not necessarily easier to use) choice. From the viewpoint of numerical efficiency, the A-grid setup is about 20% faster in simulations reported here, the difference comes largely from the overhead in computing momentum advection and biharmonic viscosity in the quasi-B-grid setup.

Recently, the hexagonal C-grid has been suggested as a promising framework for the large-scale modeling of ocean and atmosphere (Ringler et al. , 2010). Its scalar part is similar to those of A- and quasi-B-grids (it uses the Barth control volumes instead of median-dual ones). An interesting future task is the comparison of hexagonal C-grid to the setups discussed here, especially because the size of its velocity space is intermediate between those of A- and quasi-B-grids.

7. Conclusions

We summarize the main points proposed above. We describe two FV setups, one formulated on a triangular A-grid and using median-dual control volumes, and the other one, using cell–median-dual discretization and called the quasi-B-grid. For the A-grid case we suggest the implementation of stabilization which is needed in a general case on a stepwise z-coordinate bottom. For the quasi-B-grid we propose to compute the horizontal momentum advection on scalar control volumes and use the modified Leith viscosity as measures to maintain stability of its large velocity space. Both setups show robust performance in tests performed by us.

Many other discretizations are in principle possible beyond these simple approaches. While the focus of ongoing research is largely on numerical accuracy offered by various discretizations, the issues of numerical efficiency and stable performance in tasks of large-scale ocean circulation are not less important. The setups considered above give examples that work stable and efficiently, but in each case there is a price to pay.

Acknowledgments

I am indebted to Jens Schröter for numerous discussions of issues mentioned here and to anonymous reviewers for critical comments that helped improve the presentation.

References

Abalakin, I., Dervieux, A., Kozubskaya, T., 2002. A vertex-centered high-order MUSCL scheme applying to linearized Euler acoustics. INRIA, Rapport de recherche 4459.

Blazek, J., 2001. Computational fluid dynamics: Principles and applications. Elsevier

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

Danilov. S., Wang, Q., Losch, M., Sidorenko, D., Schröter, J., 2008. Modeling ocean circulation on unstructured meshes: comparison of two horizontal discretizations. Ocean Dynamics 58, 365–374.

Fox-Kemper, B., Menemenlis, D., 2008. Can large eddy simulation techniques improve mesoscale rich ocean models? In: Ocean modeling in an eddying regime, Ed. M. W. Hecht and H. Hasumi, Geophysical Monograph 177, AGU, 319–337.

Kalnay, E., Kanamitsu, M., Kistler, R., Collins, W., Deaven, D., Gandin, L., Iredell, M., Saha, S., White, G., Woollen, J., Zhu, Y., Chelliah, M., Ebisuzaki, W., Higgins, W., Janowiak, J., Mo, K. C., Ropelewski, C., Wang, J., Leetmaa, A., Reynolds, R., Jenne, R., Joseph, D., 1996. The NCEP/NCAR 40-year reanalysis project. Bulletin of the American Meteorological Society 77, 437-471.

Killworth, P. D., Stainforth, D., Webb, D. J., Paterson, S. M., 1991. The development of a free-surface Brian-Cox-Semptner Ocean model. J. Phys. Oceanogr. 21, 1333–1348.

Le Roux, D. Y., Rostand, V., Pouliot, B., 2007. Analysis of numerically induced oscillations in 2D finite-element shallow-water models. Part I: Inertia-gravity waves. SIAM J. Sci. Comput. 29, 331-360.

Le Roux, D. Y., Pouliot, B., 2008. Analysis of numerically induced oscillations in 2D finite-element shallow-water models. Part II: Free planetary waves, SIAM J. Sci. Comput. 30, 1971–1991.

Le Roux, D. Y., Hanert, E., Rostand, V., Pouliot, B., 2009. Impact of mass lumping on gravity and Rossby waves in 2D finite-element shallow-water models. Int. J. Numer. Meth. Fluids 59, 767–790.

Le Sommer, J., Penduff, T., Theetten, S., Madec, G., Barnier, B., 2009. How momentum advection schemes influence current-topography interactions at eddy-permitting resolution. Ocean Modelling 29, 1-14.

Miura, H., 2007. An upwind-biased conservative advection scheme for spherical hexagonal-pentagonal grids. Mon. Wea. Rev. 135, 4038–4044.

Ollivier-Gooh, C., Van Altena, M., 2002. A high-order-accurate unstructured mesh finite-volume scheme for the advection/diffusion equation. J. Comput. Phys. 181, 729–752.

Ouvrard, H., Kozubskaya, T., Abalakin, I., Koobus, B., Dervieux, A., 2009. Advective vertex-centered reconstruction scheme on unstructured meshes. INRIA, Rapport de recherche 7033.

Pacanowsky, R. C., Philander, S. G. H., 1981. Parameterization of vertical mixing in numerical-models of tropical oceans, J. Phys. Oceanogr. 11, 1443–1451.

Piggott, M. D., Pain, C. C., Gorman, G. J., Marshall, D. P., Killworth, P. D., 2008. Unstructured adaptive meshes for ocean modeling. In: Ocean modeling in an eddying regime, Ed. M. W. Hecht and H. Hasumi, Geophysical Monograph 177, AGU, 383–408.

Ringler, T. D., Randall, D. A., 2002. A potential enstrophy and energy conserving numerical scheme for solutions of the shallow-water equations on sphere. Mon. Wea. Rev. 130, 1397–1410.

Ringler, T. D., Randall, D. A., 2002. The ZM grid: an alternative to the Z grid. Mon. Wea. Rev. 130, 1411–1422.

Ringler, T. D., Thuburn, J., Klemp, J. B., Skamarock, W. C., 2010. A unified approach to energy conservation and potential vorticity dynamics for arbitrarily-structured C-grids. J. Comput. Phys. 229, 3065–3090.

Skamarock, W. C., Menchaca, M., 2010. Conservative transport schemes for spherical geodesic grids: high-order reconstructions for forward-in-time schemes. Mon. Wea. Rev. 138, 4497–4508.

Skamarock, W. C., Gassmann, A., 2011. Conservative transport schemes for spherical geodesic grids: high-order flux operators for ODE-based time integration. Mon. Wea. Rev. doi: 101175/MWR-D-10-05056.1.

Sidorenko, D., Wang, Q., Danilov, S. and Schröter, J., 2011. FESOM under Coordinated Ocean-ice Reference Experiment forcing. Ocean Dynamics 61, 881-890.

Szmelter, J., Smolarkiewicz, P., 2010. An edge-based unstructured mesh discretization in geospherical framework, J. Comput. Phys. 229, 4980-4995.

Timmermann, R., Danilov, S., Schröter, J., Böning, C., Sidorenko, D., Rollenhagen, K., 2009. Ocean circulation and sea ice distribution in a finite-element global ice–ocean model. Ocean Modelling 27, 114–129

Wang, Q., Danilov, S., Schröter, J., 2008. Finite Element Ocean circulation Model based on triangular prismatic elements, with application in studying the effect of topography representation. J. Geophys. Res. 113, C05015. doi:10.1029/2007JC004482

Webb, D.J., de Cuevas, B. A., Richmond, C., 1998. Improved advection schemes for ocean models. J. Atm. Ocean. Tech. 15, 1171–1187.

White, L., Deleersnijder, E., Legat, V., 2008. A three-dimensional unstructured mesh shallow-water model, with application to the flows around an island and in a wind driven, elongated basin. Ocean Modell. 22, 26–47.

Captions to figures

Fig. 1. Schematics of mesh geometry. Left panel: In the horizontal plane, the scalar quantities and vertical velocities are located at mesh nodes (circles). The horizontal velocities are at nodes on A-grid and on centroids (squares) on quasi-B-grid. An edge is characterized by its two nodes i_1 and i_2, two neighboring triangles t_1 and t_2, the edge vector $\mathbf{L}$ directed to i_2 (t_1 on the left) and two cross-vectors $\mathbf{S}(1:2)$ directed to centroids. The median-dual control cells in the horizontal plane are formed by connecting mid-edges with centroids (thin lines). Control cells for the horizontal velocities on quasi-B-grid coincide with triangles. Three-dimensional control volumes are prisms based on respective control cells with top and bottom faces on the level surfaces $\overline{z}_n$. Right panel: In the vertical plane, the temperature, salinity, pressure and horizontal velocities are at mid-levels Z_n. The vertical velocity is at full levels $\overline{z}_n$.

Fig. 2. Schematics explaining boundary conditions on the horizontal velocity on A-grid. The horizontal velocities at vertical wall edges are set to zero (four-stars). The 'partial' control volumes hosting these locations are skipped in horizontal velocity computations, so that one always deals with full control volumes in layers from $n = 1$ to $n = N_i^{min} - 1$. Arrows show locations where the bottom drag is applied. The vertical velocity is zero only at bottom locations, but is allowed at vertical walls to accommodate volume fluxes through faces of control volumes.

Fig. 3. Snapshots of elevation (left) and temperature (middle and right) (at approximately 100 m depth) in zonally reentrant channel. Left and middle panels: A-grid, the Miura advection; right: quasi-B-grid, MUSCL advection and MC momentum advection.

Fig. 4. Doubled kinetic energy per unit mass (m^2/s^2) as a function of time (days) in channel experiments. The two gray curves correspond to MA momentum advection on quasi-B-grid with weak dissipation (thick) with $A_{bh} = 10^{10}$ m^2/s and $C_{mL} = 0.5$ and strong dissipation (thin) with $A_{bh} = 3.0 \times 10^{10}$ m^2/s and $C_{mL} = 1.0$. They show similar energy levels, pointing at the dominance of dissipation due to upwinding. Simulations with MC momentum advection (black thick curve) reach higher energy levels but even they are below the result for A-grid (thin black curve). Initial evolution phase is very similar in all cases and is retained only for A-grid. The reference value of 0.11 m^2/s is not achieved, but A-grid simulations are the closest to

it.

Fig. 5. Snapshots of vertical velocity (m/s) at approximately 100 m depth in quasi-B-grid runs with different advection of momentum. Left column: MC scheme (top); MA scheme, $C_{mL} = 0.5$ (bottom). Right column: MC (top); MD, $C_{mL} = 0.5$ (bottom). Temperature advection is with the Miura (left column) or MUSCL (right column) schemes.

Fig. 6. Snapshots of simulated elevation in the North Atlantic on completing 1 year of integration in quasi-B-grid (left) and A-grid (right) setups. While the pattern is very similar in both cases, the A-grid develops noise in the shallow regions (the periphery of the Labrador Sea and the vicinity of Iceland; there are many other places along the western coast yet they cannot be discerned in the figure). Bottom panels zoom into the area around Iceland to visualize the noise on A-grid. In most cases it can be eliminated by refining the mesh.