LANS-α turbulence modeling for coastal sea: An application to Alderney Race

Anne-Claire Bennis, Feddy Adong, Martial Boutet, Franck Dumas

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

LANS-α turbulence modeling for coastal sea: an application to Alderney Race

A.-C. Bennis [1a], F. Adong[a], M. Boutet[a], F. Dumas[b]

[a]*Normandie Univ., UNICAEN, CNRS, UNIROUEN, Morphodynamique Continentale et Côtière (M2C), Caen, France*
[b]*Shom (STM/REC), Brest, France*

ARTICLE INFO

Article history:

ABSTRACT

The Lagrangian-Averaged Navier-Stokes-α (LANS-α) turbulence model was implemented for the first time in a coastal hydrodynamic model. We present in this paper the details of the implementation, as well as the difficulties encountered. To overcome the difficulties, a convolution filter was used instead of the Helmoltz operator, and incompressibility was imposed in both rough and smooth velocities. The results of the second numerical implementation were tested against the results of simulations without LANS-α for a realistic application showing the tidal dynamic in Alderney Race, Normandy, France, which has the strongest currents in western Europe. The behavior of LANS-α is consistent with the conclusions of former studies, which supports our results. The findings are: i) LANS-α re-energizes the flow recovering higher-resolution turbulence statistics in lower-resolution simulations, leading to 30% savings in computing time, ii) LANS-α produces the two kinds of inertial range for barotropic turbulence, with turbulent energy decays in $k^{-5/3}$ and in k^{-3} (k being the wave number) and iii) LANS-α strongly characterizes the turbulence induced by deformation. In the future, these results need to be compared to measurements and to other turbulence modeling approaches, including dissipative LES or DNS, to evaluate their relevance for the community.

1. Introduction

Turbulence plays a major role in ocean circulation, in particular fostering mixing and kinetic energy dissipation. The swirling motion of eddies transports water, heat, salt, and other chemicals over long distances and helps promote large-scale mixing of the ocean [e.g. 27]. Eddies also promote the rise of nutrients to the surface, which are normally found in colder and deeper waters [e.g. 28].

The full range of oceanic turbulent structures may be captured using direct numerical simulation (DNS), which consists in solving the full Navier-Stokes equations. This method ensures that the main features of the turbulence,

[1]corresponding author: A.-C. Bennis, Normandie Univ., UNICAEN, CNRS, UNIROUEN, Morphodynamique Continentale et Côtière (M2C), 24 rue des Tilleuls, 14000 Caen, France. Email: ac.bennis@unicaen.fr. Phone: +33.2.31.56.57.18. Fax: +33.2.31.56.57.57.

including eddy sizes, vorticity, circulation and nonlinear convection, are known but requires very fine spatial resolution to capture all characteristics of the turbulence. The smallest mesh cell needs to be smaller than the Kolmogorov scale, varying in both time and space. To date, the use of DNS for realistic coastal applications has been impractical due to large computational costs inherent to the very fine resolution. To overcome this, turbulence models must be introduced to capture the unrepresented effects occurring below the available resolution of numerical simulations. This therefore entails mimicking the effects of the smaller scales on the larger scales without explicitly calculating them. This is usually achieved by introducing a statistical averaging procedure, leading to the Reynolds-averaged Navier-Stokes (RANS) equations or by separating the spatial scales, which leads to the large eddy simulation (LES) equations. Both equation sets generate unclosed terms which are generally modeled with a dissipative approach, usually based on a turbulent viscosity. These approaches are well founded, since the role of the small scales, which are being modeled, is to remove the energy generated through nonlinear interactions of the larger scales that are explicitly computed. However, flow variability is also caused by nonlinear interactions of small-scale motions together while being swept along by the larger motions. Thus, the potential danger of enhancing viscous dissipation is that we lose the energy and information from the smaller scales and produce, as a result, unrealistically and inaccurately low variability.

The LANS-α model is a non-dissipative approach that modifies nonlinearity to alter energy transfer among scales, thereby providing an alternative method to reach closure without enhancing viscosity. The main feature of LANS-α regularization is to convect the flow with a smoothed velocity $\vec{u}$ leading to a reduction in the nonlinear effects by a magnitude governed by the smoothing properties. Note that the LANS-α equations can be rewritten in the classic LES framework with an explicit expression of the subgrid scale stress [more details in 11] and can therefore be interpreted as part of the LES methods. LANS-α has been implemented in the POP ocean model and applied to the circulation of the Atlantic Ocean, as well as to the motion of the circumpolar current [12, 25, 13]. The experiments of [12] and [25] showed remarkable energization of the eddy and mean kinetic energy fields equivalent to twice the model resolution. This aspect of LANS-α is especially interesting for coastal modeling. However, POP-α is not used for the moment due to unresolved numerical stability issues.

Many scientists, fishermen and sailors have reported the existence of highly energetic turbulent cells in Alderney Race, France. In the last few years, efforts to develop marine renewable energy have enabled the scientific community to characterize and modelize its turbulent activity. In-situ measurements [9, 26] over one year have revealed that high turbulent activity is present in Alderney Race, inducing significant changes in velocity magnitude and direction within a few seconds. Acoustic Doppler current profiler recordings showed the three-dimensional nature of the turbulent cells. In parallel, numerical modeling at local and regional scales was performed using the lattice Boltzmann method and classical LES [22, 6]. The authors concluded that the rough sea bottom of Alderney Race is the likely reason for ejection of the turbulent cells. The present study is a follow-up to these earlier studies by implementing and assessing the performance of the LANS-α turbulence model applied to Alderney Race.

After an introduction, the methodology is described in Section 2 with information on the coastal numerical model and on implementation of the LANS-α turbulence model. The LANS-α momentum equations, the smoothing procedure and the modified numerical algorithm are detailed. Section 3 presents Alderney Race and its mean particularities, as well as the numerical configuration, settings and results. The latter are discussed in terms of the impact of spatial resolution, turbulence energy decay and bottom-induced turbulence. Lastly, Section 4 presents the conclusions of the study and future prospects.

2. Methodology

2.1. The Coastal Numerical Model

The coastal numerical model used in the study was the Model for Applications at Regional Scales (MARS) [16]. MARS solves the three-dimensional primitive equations using incompressible, Boussinesq and hydrostatic assumptions. The equations are written in the framework of the terrain following σ coordinate:

$$\sigma = \frac{z - \eta}{\eta + H}, \qquad -1 \leq \sigma \leq 0, \tag{1}$$

where η is the free surface elevation, H is the water depth corresponding to the bathymetric absolute value of bottom position and z is the vertical coordinate. The governing equations are in the (x, y, σ, t) spatio-temporal frame-

work:

$$\frac{\partial v_x}{\partial t} + g\frac{\partial \eta}{\partial x} = G_x + \frac{1}{D^2}\frac{\partial}{\partial \sigma}\left(\nu_V \frac{\partial v_x}{\partial \sigma}\right), \tag{2}$$

$$\frac{\partial v_y}{\partial t} + g\frac{\partial \eta}{\partial y} = G_y + \frac{1}{D^2}\frac{\partial}{\partial \sigma}\left(\nu_V \frac{\partial v_y}{\partial \sigma}\right), \tag{3}$$

$$\frac{1}{D}\frac{\partial p}{\partial \sigma} = -\rho g, \tag{4}$$

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (D\mathbf{v}) + \frac{\partial \tilde{v}_\sigma}{\partial \sigma} = 0, \tag{5}$$

where $\mathbf{V} = (v_x, v_y, v_\sigma) = (\mathbf{v}, v_\sigma)$ is the fluid velocity vector, p is the fluid pressure, ∇ is the horizontal gradient, ρ is the density, g is gravitational acceleration, $D = H + \eta$ is the depth and ν_V the vertical turbulent viscosity. $\mathbf{G} = (G_x, G_y)$ are the x and y components of the baroclinic momentum term incorporating advective terms, Coriolis forcing, surface and internal pressure gradient and horizontal eddy viscosity:

$$\mathbf{G} = -\mathcal{L}(\mathbf{v}, \tilde{v}_\sigma) + \mathbf{f} \times \mathbf{v} - \frac{1}{\rho_0}\nabla P_a + \mathbf{\Pi} + \mathcal{D}_H(\mathbf{v}), \tag{6}$$

where ρ_0 is the seawater reference density, $\mathbf{f}$ is the Coriolis parameter, $\mathcal{D}_H(\mathbf{v}) = \nu_H \Delta \mathbf{v}$ is the horizontal diffusion operator, ν_H is the horizontal turbulent viscosity, P_a is the atmospheric pressure at sea surface, $\mathbf{\Pi}$ is the internal pressure gradient [16]. $\tilde{v}_\sigma$ is the vertical velocity in the sigma coordinate framework:

$$\tilde{v}_\sigma = \frac{1}{D}\left(v_\sigma - \frac{\partial z}{\partial t} - v_x\frac{\partial z}{\partial x} - v_y\frac{\partial z}{\partial y}\right), \tag{7}$$

and the non-linear operator $\mathcal{L}$ used to compute the advective term is defined as:

$$\mathcal{L}(\mathbf{v}, \tilde{v}_\sigma) = v_x\frac{\partial \mathbf{v}}{\partial x} + v_y\frac{\partial \mathbf{v}}{\partial y} + \tilde{v}_\sigma\frac{\partial \mathbf{v}}{\partial \sigma}. \tag{8}$$

Horizontal viscosity, acting on the horizontal diffusion operator, is assessed according to Okubo [24]:

$$\nu_H = 0.01 \times \text{fvisc} \times \Delta_H^{1.15}, \tag{9}$$

where Δ_H is the horizontal mesh size and fvisc is a user coefficient. The vertical mixing term ν_V is calculated using the well-known turbulent closure $K - \epsilon$ modified according to [30] and based on the generic length scale approach [29].

The barotropic formulation is obtained by integrating Eq. (2)-(5) from the bottom ($\sigma = -1$) to the surface ($\sigma = 0$) and considering kinematic boundary conditions. Barotropic and barocline modes are coupled and the solving algorithm was modified to the LANS-α set-up (see Section 2.2.5 for additional information). MARS uses finite differences to discretize the primitive equations, and Alternate Direction Implicit (ADI) scheme for the time stepping [16]. This procedure ensures a high numerical stability by removing the numerical constraints imposed by the Courant-Friedrich-Lewy criteria. The ADI scheme splits the time step into two parts: the first half time devoted to the computation of predictor fields, and the second half time where the latter are corrected. This scheme is only implicit with respect to the direction of the computation, leading to computation of the free surface elevation at every half time step, whereas the baroclinic and barotropic velocities are calculated alternatively.

2.2. Implementation of the LANS-α Turbulence Model

2.2.1. The LANS-α Primitive Equations

Our implementation of LANS-α in MARS is based on the following primitive-equation form presented in [12]:

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{v} + u_z\frac{\partial v_z}{\partial z} + \underbrace{\left[\nabla\mathbf{u}^T\cdot(\mathbf{v}-\mathbf{u}) - \alpha^2\nabla(|\nabla\mathbf{u}|^2)/2\right]}_{\text{LANS terms}} - \mathbf{f}\times\mathbf{u} = -\frac{1}{\rho_0}\nabla p + \mathcal{D}_{\mathbf{H}}(\mathbf{v}) + \frac{\partial}{\partial z}\left(\nu\frac{\partial \mathbf{v}}{\partial z}\right), \tag{10}$$

$$\frac{\partial p}{\partial z} = -\rho g, \tag{11}$$

$$\mathbf{u} - \alpha^2\Delta\mathbf{u} = \mathbf{v}, \tag{12}$$

$$\nabla\cdot\mathbf{u} + \frac{\partial u_z}{\partial z} = 0, \tag{13}$$

in which two velocities appear $(v_x, v_y, v_z) = (\mathbf{v}, v_z)$ and $(u_x, u_y, u_z) = (\mathbf{u}, u_z)$, which are related by the Helmholtz operator in Eq. (12). In this formulation, the small-scales structures in the rough velocity (v_x, v_y, v_z) are advected by the smooth velocity (u_x, u_y, u_z). The α parameter in Eq. (12) is the smoothing length scale, also called the filter width, acting on the rough velocity, which specifies the degree of smoothing. The additional non-linear term labeled "LANS terms" is necessary to ensure Kelvin's circulation theorem. The Eq. (10)-(13) uses incompressible, Boussinesq and hydrostatic assumptions, as in the initial MARS model. The Eq. (11) is a simplified form of the z-component of the momentum equation due to the hydrostatic assumption. The Eq. (13) is the mass conservation equation for an incompressible fluid, which is applied to the smooth velocity.

Using this form, it is clear that for $\alpha \longrightarrow 0$, the set of equations (10)-(13) reverts back to the standard primitive equations solved by MARS. Note that to recover the form presented in [12], where the LANS-α contribution is $\nabla\mathbf{u}^T \cdot \mathbf{v}$, it is necessary to develop the extra non-linear term using the identity $\nabla\mathbf{u}^T \cdot \mathbf{u} = \nabla\left(|\mathbf{u}|^2\right)/2$ and include the term $-\nabla\left(|\mathbf{u}|^2\right)/2 - \alpha^2\nabla(|\nabla\mathbf{u}|^2)/2$ into the definition of the pressure.

2.2.2. Smoothing Procedure

As described previously, MARS is a semi-implicit code coupling barotropic and baroclinic velocity fields through a predictor-corrector procedure. However, this approach requires smoothing of the velocity during both the predictor and corrector steps, which increases computational time. To overcome this difficulty, smoothing is computed by a convolution filter that has the same effects as the Helmholtz operator with a smaller computational form [e.g. 25]. In the discrete form, the rough and smooth fluid velocities are related by:

$$\mathbf{u}[i, j] = (\mathbf{v} * K_\alpha)[i, j] = \sum_{l=-\mathrm{fw}}^{\mathrm{fw}} \sum_{m=-\mathrm{fw}}^{\mathrm{fw}} w_\alpha[l, m]\mathbf{v}[i - l, j - m], \tag{14}$$

where K_α is the convolution kernel, w_α represents the filter weights and fw is the half stencil size used during the weighted average procedure. The filter weights of the used 2D 7-point filter are listed in Table 1.

Table 1. Weights of the 2D 7-point filter from [25].

j+3	0.12	0.14	0.16	0.35	0.16	0.14	0.12
j+2	0.14	0.16	0.18	0.40	0.18	0.16	0.14
j+1	0.16	0.18	0.20	0.45	0.20	0.18	0.16
j	0.35	0.40	0.45	1.00	0.45	0.40	0.35
j-1	0.16	0.18	0.20	0.45	0.20	0.18	0.16
j-2	0.14	0.16	0.18	0.40	0.18	0.16	0.14
j-3	0.12	0.14	0.16	0.35	0.16	0.14	0.12
	i-3	i-2	i-1	i	i+1	i+2	i+3

When smoothing is performed with the Helmholtz operator, enforcing the incompressibility on the smooth velocity is sufficient to also ensure incompressibility on the rough velocity. However, this property is clearly not suitable when the weighted averaged procedure is employed. For this reason, we imposed the incompressibility in both the smooth and rough velocities, which requires addition of the following equation to the set of equations (10)-(13):

$$\nabla \cdot \mathbf{v} + \frac{\partial v_z}{\partial z} = 0. \tag{15}$$

2.2.3. Sigma Coordinates Framework

Implementation of LANS-α in MARS (hereafter noted MARS-α) is ensured by rewriting the set of equations (10)-(13) and Eq. (15) in vertical σ coordinate framework, such that:

$$\frac{\partial v_x}{\partial t} + g\frac{\partial \eta}{\partial x} = G_{\alpha x} + \frac{1}{D^2}\frac{\partial}{\partial \sigma}\left(\nu_V \frac{\partial v_x}{\partial \sigma}\right), \tag{16}$$

$$\frac{\partial v_y}{\partial t} + g\frac{\partial \eta}{\partial x} = G_{\alpha y} + \frac{1}{D^2}\frac{\partial}{\partial \sigma}\left(\nu_V \frac{\partial v_y}{\partial \sigma}\right), \tag{17}$$

$$\frac{1}{D}\frac{\partial p}{\partial \sigma} = -\rho g, \tag{18}$$

$$\frac{\partial \eta}{\partial t} + \nabla \cdot \left[D\left(\gamma_R \mathbf{v} + \gamma_S \mathbf{u}\right)\right] + \frac{\partial\left[\gamma_R \tilde{v}_\sigma + \gamma_S \tilde{u}_\sigma\right]}{\partial \sigma} = 0. \tag{19}$$

where the γ_R parameter is used to determine whether both $\mathbf{u}$ and $\mathbf{v}$ are non-divergent ($\gamma_R = 0.5$) or only one of the two is non-divergent ($\gamma_R = 1$ for $\mathbf{v}$ and $\gamma_R = 0$ for $\mathbf{u}$). In the work presented here, $\gamma_R = 0.5$ was chosen because of the smoothing procedure such that $\nabla \cdot \mathbf{v}$ is not always equal to zero if only $\mathbf{u}$ is defined as a non-divergent field.

$\tilde{u}_\sigma$ is the vertical smooth velocity in the sigma coordinate framework given by:

$$\tilde{u}_\sigma = \frac{1}{D}\left(u_\sigma - \frac{\partial z}{\partial t} - u_x\frac{\partial z}{\partial x} - u_y\frac{\partial z}{\partial y}\right), \tag{20}$$

and $\mathbf{G}_\alpha = (G_{\alpha x}, G_{\alpha y})$ are the x and y components of the baroclinic LANS-α momentum forcing terms:

$$\mathbf{G}_\alpha = -\mathcal{L}_\alpha(\mathbf{v}, \tilde{v}_\sigma) - \mathcal{N}_\alpha(\mathbf{u}) - \mathbf{f} \times \mathbf{u} - \frac{1}{\rho_0}\nabla P_a + \mathbf{\Pi} + \mathcal{D}_H(\mathbf{v}). \tag{21}$$

The non-linear operator $\mathcal{L}_\alpha$ is now based on the smooth velocity advection:

$$\mathcal{L}_\alpha(\mathbf{u}, \mathbf{v}) = u_x\frac{\partial \mathbf{v}}{\partial x} + u_y\frac{\partial \mathbf{v}}{\partial y} + \tilde{u}_\sigma\frac{\partial \mathbf{v}}{\partial \sigma}, \tag{22}$$

and the additional non-linear term begins:

$$\mathcal{N}_\alpha(\mathbf{u}, \mathbf{v}) = \left[\nabla \mathbf{u}^T \cdot (\mathbf{v} - \mathbf{u}) - \alpha^2 \nabla(|\nabla \mathbf{u}|^2)/2\right] - \frac{1}{D}\nabla z \cdot (\mathbf{v} - \mathbf{u})\frac{\partial \mathbf{u}}{\partial \sigma}, \tag{23}$$

where we have neglected all contributions from order α^2/D^2. The solving algorithm implemented in MARS requires simultaneously solving barotropic (vertically integrated 2D) and baroclinic (remaining 3D) velocity, leading to the construction of a specific MARS-α barotropic model.

2.2.4. MARS-α Barotropic Model

The MARS-α barotropic formulation is obtained by integrating Eq. (10)-(13) over the vertical from the bottom ($\sigma = -1$) to the surface ($\sigma = 0$) and considering kinematic boundary conditions. It yields the following set of equations:

$$\frac{\partial \bar{v}_x}{\partial t} + g\frac{\partial \eta}{\partial x} = \bar{G}_{\alpha x} + \frac{\tau_{Sx}}{\rho D} - \frac{\tau_{Bx}}{\rho D}, \tag{24}$$

$$\frac{\partial \bar{v}_y}{\partial t} + g\frac{\partial \eta}{\partial y} = \bar{G}_{\alpha y} + \frac{\tau_{Sy}}{\rho D} - \frac{\tau_{By}}{\rho D}, \tag{25}$$

$$\frac{\partial \eta}{\partial t} + \nabla \cdot \left[D\left(\gamma_R \bar{\mathbf{v}} + \gamma_S \bar{\mathbf{u}}\right)\right] = 0, \tag{26}$$

where $\tau_\mathbf{S} = (\tau_{Sx}, \tau_{Sy})$ and $\tau_\mathbf{B} = (\tau_{Bx}, \tau_{By})$ are the surface and bottom stress, respectively. $\bar{\mathbf{u}} = (\bar{u}_x, \bar{u}_y)$ is the smooth barotropic velocity vector defined as:

$$\bar{\mathbf{u}} = \int_0^1 \mathbf{u}\mathrm{d}\sigma, \tag{27}$$

and $\bar{\mathbf{G}}_\alpha$ is the vertically integrated baroclinic momentum LANS-α forcing terms:

$$\bar{\mathbf{G}}_\alpha = \int_0^1 \left(-\frac{1}{\rho_0}\nabla P_a + \mathbf{\Pi} + \mathcal{D}_H(\mathbf{v}) - \mathcal{L}_\alpha(\mathbf{u}, \mathbf{v}) - \mathcal{N}_\alpha(\mathbf{u}, \mathbf{v}) - f \times \mathbf{u}\right)\mathrm{d}\sigma. \tag{28}$$

2.2.5. MARS-α Solving Algorithm

The MARS-alpha solving algorithm is the adaptation of the MARS algorithm to the augmentation with the LANS-alpha turbulence model. When MARS is modified to use the LANS-alpha model, the following changes are required: (i) there are two full velocity fields to consider, the rough velocity $\mathbf{v}$, and the smooth velocity $\mathbf{u}$; (ii) likewise, there are smooth and rough barotropic velocities, $\bar{\mathbf{u}}$ and $\bar{\mathbf{v}}$; (iii) the momentum forcing term must be modified with the use of appropriate velocities in the nonlinear and Coriolis terms and the addition of the nonlinear term $\mathcal{N}_\alpha(\mathbf{u}, \mathbf{v})$; (iv) the continuity equation becomes Eq. (19) with $\mathbf{u}$ and $\mathbf{v}$ that are non-divergent. Because the time derivative in the MARS-α baroclinic momentum equation is applied to the rough velocity $\mathbf{v}$, one must take a time step in $\mathbf{v}$ and then compute $\mathbf{u}$ by smoothing each vertical level. The barotropic-baroclinic coupling implies to simultaneously solve the 2D and 3D systems of equations. A predictor-corrector algorithm is used and the time step is splitted into two parts: a first estimate of the solution is computed during the first half time step (predictor phase) and then this estimate is corrected during the second half time step (corrector phase). Each phase of the predictor-corrector algorithm contains two steps, one for each component (see Fig. 1).

Only steps 1 and 4 related to the x-components are detailed hereafter. Similar treatments being provided for the y-components, steps 2 and 3 are shown in Appendix A.1 and A.2, respectively. Note that the MARS-α algorithm described in this section uses an explicit bottom friction term because the semi-implicit version has introduced additional complications. The boxes indicate additional steps introduced by the smoothing operator.

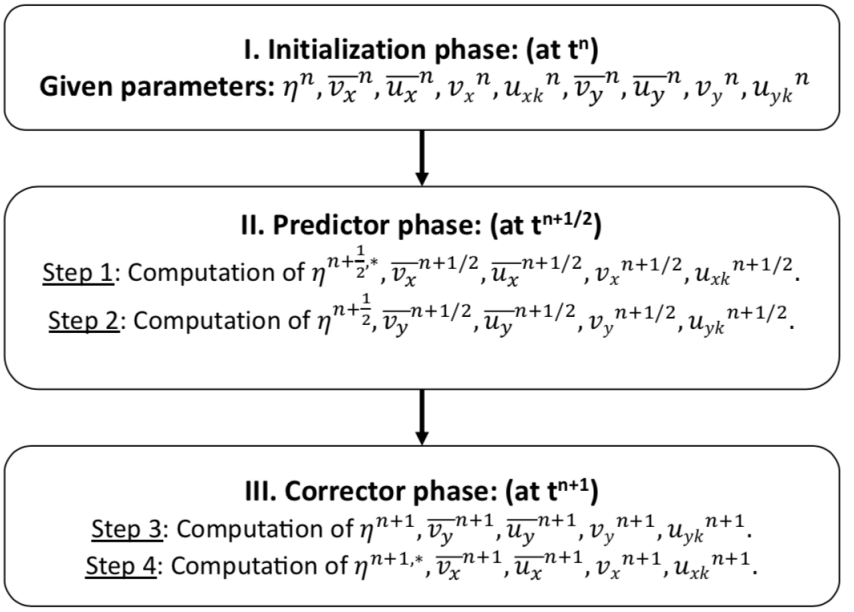

Fig. 1. Description of the MARS-α solving algorithm.

Predictor phase.

Step1: Barotropic part of the x-components: compute $\eta^{n+1/2,*}$, $\bar{v}_x^{n+1/2}$ and $\bar{u}_x^{n+1/2}$ by solving the following coupled system

$$\bar{v}_x^{n+1/2} + \theta_N \frac{g\Delta t}{2} \frac{\partial \eta^{n+1/2,*}}{\partial x} = \bar{v}_x^n + \frac{\Delta t}{2}\left(\bar{G}_{\alpha x}^n + \frac{\tau_{Sx}^n}{\rho D_x^n} - f_t v_{Bx}^n - \theta_P g \frac{\partial \eta^n}{\partial x}\right), \tag{29}$$

$$\eta^{n+1/2,*} + \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[D_x^n\left(\gamma_R \bar{v}_x^{n+1/2} + \gamma_S \bar{u}_x^{n+1/2}\right)\right] = \eta^n - \frac{\Delta t}{2}\frac{\partial}{\partial y}\left[D_y^n\left(\gamma_R \bar{v}_y^{n+1/2,*} + \gamma_S \bar{u}_y^{n+1/2,*}\right)\right], \tag{30}$$

where $\theta_N = 0.55$, $\theta_P = 0.45$, Δt is the time step, v_{Bx} is the x-component of the rough velocity at the bottom and f_t is the bottom friction coefficient which is a function of drag coefficient, bottom velocity, depth and half time step.

$\bar{v}_y^{n+1/2,*}$ is estimated by an implicit discretization of (24) and $\bar{u}_y^{n+1/2,*}$ by smoothing $\bar{v}_y^{n+1/2,*}$:

$$\bar{v}_y^{n+1/2,*} = \bar{v}_y^n + \frac{\Delta t}{2}\left(\bar{G}_{\alpha y}^n + \frac{\tau_{Sy}^n}{\rho D_x^n} - \frac{\tau_{By}^n}{\rho D_x^n} - g\frac{\partial \eta^n}{\partial y}\right). \tag{31}$$

$$\boxed{\bar{u}_y^{n+1/2,*} = K_\alpha * \bar{v}_y^{n+1/2,*}} \tag{32}$$

Baroclinic part of the x-components: compute $v_x^{n+1/2}$ by solving the following equation (where k are the indices of the sigma vertical levels. u_{xk} and v_{xk} represent the x-component of the smooth and the rough velocity for a given k index, respectively.)

$$v_x^{n+1/2} - \frac{\Delta t}{2}\frac{\partial}{\partial \sigma}\left(\frac{\nu_V}{(D_x^n)^2}\frac{\partial v_x^{n+1/2}}{\partial \sigma}\right) = v_x^n + \tfrac{\Delta t}{2}\left(G_{\alpha x}^n - g\frac{\partial \eta^n}{\partial x}\right). \tag{33}$$

$$\boxed{u_{xk}^{n+1/2} = \left(K_\alpha * v_{xk}^{n+1/2}\right), \quad k = 1, \cdots, \text{kmax}} \tag{34}$$

Corrector phase.

Step 4: Barotropic part of the x-components: compute η^{n+1}, $\bar{v}_x^{n+1}$ and $\bar{u}_x^{n+1}$ by solving the following coupled system

$$\begin{aligned} \bar{v}_x^{n+1} + \theta_N \frac{g\Delta t}{2}\frac{\partial \eta^{n+1}}{\partial x} &= \bar{v}_x^{n+1/2} + \frac{\Delta t}{2}\left(\bar{G}_{\alpha x}^{n+1/2} + \frac{\tau_{Sx}^{n+1/2}}{\rho D_x^{n+1,*}}\right) \\ &- \frac{\Delta t}{2}\left(f_t v_{Bx}^{n+1/2} + \theta_P g\frac{\partial \eta^{n+1,*}}{\partial x}\right), \end{aligned} \tag{35}$$

$$\begin{aligned} \eta^{n+1} + \frac{\Delta t}{2}\frac{\partial}{\partial x}\left[D_x^{n+1,*}\left(\gamma_R \bar{v}_x^{n+1/2} + \gamma_S \bar{u}_x^{n+1/2}\right)\right] &= \eta^{n+1,*} - \frac{\Delta t}{2}\frac{\partial}{\partial y}\left(D_y^{n+1,*}\gamma_R \bar{v}_y^{n+1,*}\right) \\ &- \frac{\Delta t}{2}\frac{\partial}{\partial y}\left(D_y^{n+1,*}\gamma_S \bar{u}_y^{n+1,*}\right), \end{aligned} \tag{36}$$

where $\bar{v}_y^{n+1,*}$ and $\bar{u}_y^{n+1,*}$ are estimated as:

$$\bar{v}_y^{n+1,*} = \bar{v}_y^{n+1/2} + \frac{\Delta t}{2}\left(\bar{G}_{\alpha y}^{n+1/2} + \frac{\tau_{Sy}^{n+1/2}}{\rho D_y^{n+1,*}} - \frac{\tau_{By}^{n+1/2}}{\rho D_y^{n+1,*}} - g\frac{\partial \eta^{n+1/2}}{\partial y}\right). \tag{37}$$

$$\boxed{\bar{u}_y^{n+1,*} = K_\alpha * \bar{v}_y^{n+1,*}} \tag{38}$$

Baroclinic part of the x-components: compute v_x^{n+1} by solving the following equation

$$v_x^{n+1} - \frac{\Delta t}{2}\frac{\partial}{\partial \sigma}\left(\frac{\nu_V}{\left(D_x^{n+1}\right)^2}\frac{\partial v_x^{n+1}}{\partial \sigma}\right) = v_x^{n+1/2} + \tfrac{\Delta t}{2}\left(G_{\alpha x}^{n+1/2} - g\frac{\partial \eta^{n+1/2}}{\partial x}\right). \tag{39}$$

$$\boxed{u_{xk}^{n+1} = \left(K_\alpha * v_{xk}^{n+1}\right), \quad k = 1, \cdots, \text{kmax}} \tag{40}$$

2.3. Numerical Implementation and Practical Considerations

2.3.1. Computation of Barotropic Components

Computation of the barotropic components in each step is the most difficult part of implementation of the MARS-α algorithm. In the following, we focus on Step 1 (ie. computation of the sea surface height and of the x-component velocities for the predictor phase), however the other steps are treated in similar fashion. To review this difficulty in more detail, we need to review the computation of the barotropic components. In the MARS model, the algorithm for solving the barotropic components consists in substituting the equation of momentum in the continuity equation to obtain an implicit equation for the time evolution of sea surface height, and then explicitly inferring barotropic velocity from the sea surface height. The resulting equation for the sea surface height is:

$$\eta^{n+1/2,*} - g\theta_N\left(\frac{\Delta t}{2}\right)^2 \frac{\partial}{\partial x}\left(D_x^n \mathrm{bv} \frac{\partial \eta^{n+1/2,*}}{\partial x}\right) = \mathrm{Yxe} - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^n \mathrm{bv Yv}\right) \tag{41}$$

$$\bar{v}_x^{n+1/2} = \mathrm{bv}\left(\mathrm{Yv} - \theta_N \frac{g\Delta t}{2}\frac{\partial \eta^{n+1/2,*}}{\partial x}\right) \tag{42}$$

with

$$\begin{aligned}
\mathrm{Yv} &= \bar{v}_x^n + \frac{\Delta t}{2}\left(\bar{G}_x^n + \frac{\tau_{Sx}^n}{\rho D_x^n} - \mathrm{ft}\left(v_{Bx}^n - \beta \bar{v}_x^n\right) - \theta_P g \frac{\partial \eta^n}{\partial x}\right),\\
\mathrm{Yxe} &= \eta^n - \frac{\Delta t}{2}\frac{\partial}{\partial y}\left(D_x^n \bar{v}_y^{n+1/2,*}\right),\\
\mathrm{bv} &= \frac{1}{(1+\beta \mathrm{ft})}.
\end{aligned} \tag{43}$$

At that point, the right-hand side of Eq. (41) is known, which leads to a tridiagonal matrix system solved using LU factorization. The velocity $\bar{v}_x^{n+1/2}$ is then deduced explicitly from Eq. (42) after resolution of Eq. (41).

Similarly, to solve Eq. (29) for $\bar{v}_x^{n+1/2}$, $\bar{u}_x^{n+1/2}$, and $\eta^{n+1/2,*}$, we need to replace $\bar{v}_x^{n+1/2}$ and $\bar{u}_x^{n+1/2}$ into Eq. (30), where $\bar{u}_x^{n+1/2}$ is obtained by application of the smoothing operator into Eq. (29), such that:

$$\bar{v}_x^{n+1/2} = \mathrm{Yv}_\alpha - \theta_N \frac{g\Delta t}{2}\frac{\partial \eta^{n+1/2,*}}{\partial x}, \tag{44}$$

$$\bar{u}_x^{n+1/2} = \mathrm{Yvs}_\alpha - \theta_N \frac{g\Delta t}{2}\left(\frac{\partial \eta^{n+1/2,*}}{\partial x}\right) * K_\alpha, \tag{45}$$

where:

$$\mathrm{Yv}_\alpha = \bar{v}_x^n + \frac{\Delta t}{2}\left(\bar{G}_{\alpha x}^n + \frac{\tau_{Sx}^n}{\rho D_x^n} - \mathrm{ft} v_{Bx}^n - \theta_P g \frac{\partial \eta^n}{\partial x}\right) \quad \text{and} \quad \mathrm{Yvs}_\alpha = \mathrm{Yv}_\alpha * K_\alpha. \tag{46}$$

Here we now obtain a semi-implicit formulation by rewriting Eq. (44) as:

$$(1+\mathrm{ft})\,\bar{v}_x^{n+1/2} = \mathrm{Yv}_\alpha + \mathrm{ft}\bar{v}_x^n - \theta_N \frac{g\Delta t}{2}\frac{\partial \eta^{n+1/2,*}}{\partial x}, \tag{47}$$

$$(1+\mathrm{ft})\,\bar{u}_x^{n+1/2} = \mathrm{Yvs}_\alpha + \mathrm{ft}\bar{u}_x^n - \theta_N \frac{g\Delta t}{2}\left(\frac{\partial \eta^{n+1/2,*}}{\partial x}\right) * K_\alpha. \tag{48}$$

The sea surface height equation in the Step1 of the MARS-α algorithm is built by substituting Eq. (47) into Eq. (30)

$$\begin{aligned}
\eta^{n+1/2,*} &- g\theta_N\left(\frac{\Delta t}{2}\right)^2\left[\gamma_R \frac{\partial}{\partial x}\left(D_x^n \mathrm{bv}\frac{\partial \eta^{n+1/2,*}}{\partial x}\right) + \gamma_S \frac{\partial}{\partial x}\left(D_x^n \mathrm{bv}\frac{\partial \eta^{n+1/2,*}}{\partial x}\right) * K_\alpha\right]\\
&= \mathrm{Yxe}_\alpha - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^n \mathrm{bv}\overline{\mathrm{Yv}}_\gamma\right),
\end{aligned} \tag{49}$$

where:

$$\mathrm{Yxe}_\alpha = \eta^n - \frac{\Delta t}{2}\frac{\partial}{\partial y}\left[D_y^n\left(\gamma_R \bar{v}_y^{n+1/2,*} + \gamma_S \bar{u}_y^{n+1/2,*}\right)\right], \tag{50}$$

$$\overline{\mathrm{Yv}}_\gamma = \gamma_R\left(\mathrm{Yv}_\alpha + \mathrm{ft}\bar{v}_x^n\right) + \gamma_S\left(\mathrm{Yvs}_\alpha + \mathrm{ft}\bar{u}_x^n\right). \tag{51}$$

The matrix system formed by Eq. (49) can be solved using an iterative conjugate gradient routine, with smoothing within each iteration [12]. In such case, the computing cost is high, because a similar system needs to be solved at each time step. In addition, in this exact derivation, the interest of using the ADI scheme is lost since the tridiagonal linear system is solved along the direction of computation. Following the work of [12], we have designed a reduced algorithm by neglecting the smoothing step on the left-hand side of Eq. (49), leading to the following algorithm:

$$\eta^{n+1/2,*} - g\theta_N\left(\frac{\Delta t}{2}\right)^2\frac{\partial}{\partial x}\left(D_x^n \mathrm{bv}\frac{\partial \eta^{n+1/2,*}}{\partial x}\right) = \mathrm{Yxe}_\alpha - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^n \mathrm{bv}\overline{\mathrm{Yv}}_\gamma\right), \tag{52}$$

$$\bar{v}_x^{n+1/2} = \mathrm{bv}\left(\mathrm{Yv}_\alpha + \mathrm{ft}\bar{v}_x^n - \theta_N\frac{g\Delta t}{2}\frac{\partial \eta^{n+1/2,*}}{\partial x}\right), \tag{53}$$

$$\bar{u}_x^{n+1/2} = \mathrm{bv}\left(\mathrm{Yvs}_\alpha + \mathrm{ft}\bar{u}_x^n - \theta_N\frac{g\Delta t}{2}\frac{\partial \eta^{n+1/2,*}}{\partial x}\right). \tag{54}$$

This reduced algorithm uses only a single smoothing step, which is in the calculation of $\overline{\mathrm{Yv}}_\gamma$ in Eq. (51). Two additional equations were added to the original MARS algorithm (41): the computation of Yvs_α in Eq. (46) and the final smooth velocity in Eq. (54).

2.3.2. Diffusion Operator

Although LANS-α is based on modification of the nonlinear terms without introducing viscous dissipation, in practice, the use of an artificial viscosity is still necessary to remove energy near the grid scale in order to avoid numerical instabilities. In addition, LANS-α, as presented in this paper, mainly concerns the modeling of the effects of horizontal eddies, does not replace vertical mixing parameterization and can be activated at the same time [12, 13].

2.3.3. Numerical stability

The implementation of LANS-alpha is not straightforward and the classical solving algorithms have been significantly modified. This can lead to numerical instability issues as reported by the LANL group. Our code is stable for 5-, 7-, 9-point filters and for all spatial resolutions (600m, 120m and 30m). However, instabilities were occurred as expected. The necessary changes to remove them are detailed below. First, instabilities due to the non-divergence of the rough velocity, which was not respected with the discrete filter, have been made our code unstable. We had to impose a non-divergence condition to the rough velocity to stabilize the code. Second, instabilities have been generated near boundaries. A modification of the width of the filter, according to the number of neighbouring wet cells, has eliminated this problem. Third, numerical instabilities were produced when the starting simulation time is the time of the maximum velocity because of the extreme values of the tidal current (up to 5 m/s) in Alderney Race. By postponing one hour the beginning of the simulation, instabilities were suppressed. Finally, the semi-implicit computation of the bottom friction coefficient (ft) performed by default in MARS has induced instabilities and we had to disable it to address this issue.

3. Case Study: Study Site and Numerical Configuration

Our implementation of LANS-α has already tested against the results of [12] for the idealized test case of a wind-driven flow over a seamount [1]. We therefore present here the realistic case of Alderney Race.

3.1. Alderney Race

The turbulence model assessment was performed for Alderney Race, which is in the middle of the English Channel, between La Hague Cape and Alderney Island (see Fig. 2). Alderney Race is a coastal area that ranges in depth from 20 to 60 meters and is a mega-tidal environment where the tidal range can reach 10 meters. The hydrodynamics are mainly driven by semi-diurnal tides, generating an oscillating current, the direction of which varies between north-east and south-west during the flood and the ebb, respectively. Ocean waves (swell- and wind-waves) propagate through Alderney Race and interact with tidal flow [4, 5]. Narrowing caused by the presence of Alderney Island accelerates the tidal current, which peaks 5 m/s during spring tides, making it the most energetic tidal site in Western Europe. This is a strategic issue for tidal energy companies, and it is therefore essential for them to properly understand the complex hydrodynamics of Alderney Race. Tidal hydrodynamics have long been studied and are reasonably well simulated by numerical models [e.g. 2]. The impacts of wave-current interactions on the tidal stream energy are the subject of in-depth studies [e.g. 19, 4, 5]. Turbulence is also a key parameter for the tidal industry and is the focus of our paper. Three-dimensional turbulent structures with characteristic lengths of a few tens of meters have been observed and are highly energetic and intermittent. The origin of the turbulence is not fully understood at present due to the highly complex environment of Alderney Race. Recent works [22, 6] suggest that the turbulent structures are ejected from the sea bottom to the sea surface. Indeed, the sea bottom is very rough and uneven with diverse features including rocks, faults, folds, sand patches, pebbles and dune fields [more details in 10]. Marked turbulent activity is produced when the strong tidal current interacts with such features [22]. In the following sections, we focus on a small portion of Alderney Race where severe bathymetric gradients are present.

3.2. Numerical Configuration and Settings

The computational domain was composed of three nested grids, the horizontal resolution of which is respectively 600 m, 120 m and 30 m, which are hereafter named Rank0, Rank1 and Rank2 (see Fig. 2). For all three grids, 20 sigma levels, evenly distributed over the vertical, were used. The simulations were performed using the fine bathymetry described in [10] for two days, i.e. 29-30 September 2017. These two days of neap tide were chosen because of the availability of in-situ data, that will be used in the future for comparison. Tides were generated using the tidal components atlas of the Service Hydrographique et Océanographique de la Marine (SHOM) [18] imposed at the open boundaries of Rank0. The Rank0 numerical outputs, i.e. flow velocity and sea surface height, were used to force the hydrodynamics at the boundaries of Rank1, and the Rank1 outputs to force Rank2. The ocean waves and wind effects were not included in these simulations in order to separate tidal effects from the coupled effects on turbulence. Simulations were carried out with and without LANS-α, and the same numerical parameters were employed in both cases, except for the horizontal turbulence modelling. For the vertical mixing parametrization, two cases were investigated with the use of the $K-\epsilon$ turbulent closure [30] and a constant vertical viscosity of $0.2\ \mathrm{m}^2.\mathrm{s}^{-1}$. The latter parameter was chosen as the minimum viscosity in order to avoid numerical instability in the experiments. The parameters for the smoothing procedure in LANS-α were those described in [25]. Different filter sizes were tested, and filters with 5, 7 and 9 points ensured numerical stability. Few differences were found between results with the two filter sizes, therefore only the results with the 7-point filter are presented here. In addition, the use of the 7-point filter reduced computational costs by more than 10% compared to the 9-point filter. Details of the 12 numerical configurations (MARS and MARS-α) are summarized in Table 2 and their computational cost is presented in Table 3.

Table 2. List of numerical parameters. Simulation names, type of turbulence modeling (LANS-α and vertical mixing), description of the LANS-α filter, details on the spatial resolution and time step are summarized. For cases Run 1 and Run 3, the constant turbulent viscosity is set to $0.2\ \mathrm{m}^2/\mathrm{s}$**.**

Name	LANS-α	Vertical mixing	Convolution filter	Spatial resolution (m)	Time step (s)
Run 1	yes	constant	7	600, 120, 30	30, 6, 1.5
Run 2	yes	$k-\epsilon$	7	600, 120, 30	30, 6, 1.5
Run 3	no	constant	-	600, 120, 30	30, 6, 1.5
Run 4	no	$k-\epsilon$	-	600, 120, 30	30, 6, 1.5

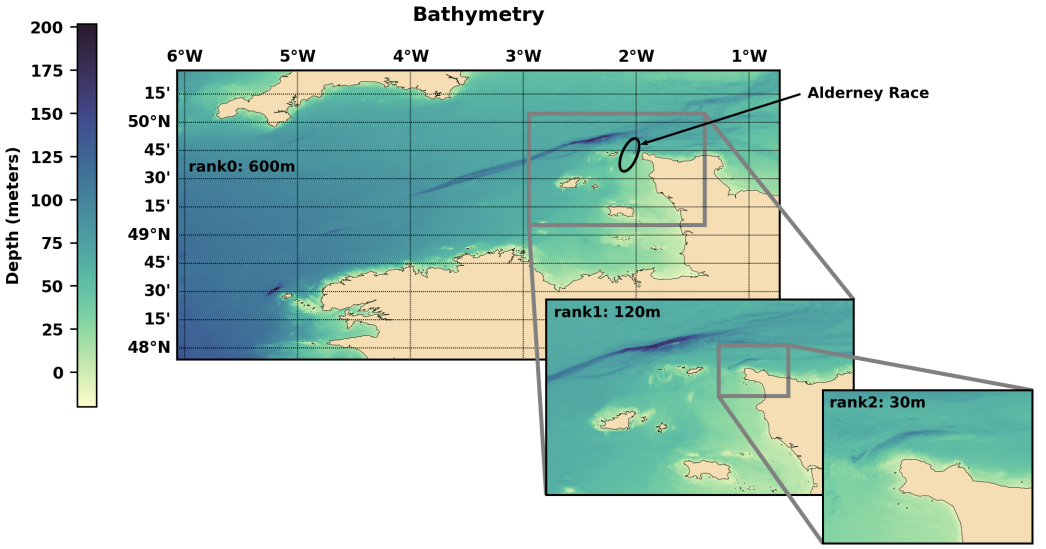

Fig. 2. Bathymetric depth from [10] used in simulations for the three nested grids (Rank0, Rank1 and Rank2).

3.3. Results and Discussion

Turbulent kinetic energy (TKE) for Runs 1 and 3 and for different spatial resolutions of 30 m and 120 m are shown in Fig. 3 and 4 for ebb and flood, respectively. TKE was computed from barotropic velocities and over 4 hours around the peak of velocity with a time step of 20 minutes such that:

$$\mathrm{TKE} = \frac{1}{2}\left(\left\langle(\bar{v}_x - \langle\bar{v}_x\rangle)^2\right\rangle + \left\langle(\bar{v}_y - \langle\bar{v}_y\rangle)^2\right\rangle\right), \tag{55}$$

where $\langle\cdot\rangle$ is the time-average operator.

In the cases presented, TKE levels were always higher for MARS-α than for MARS. This is consistent with the conclusions of [12, 13] and [25], indicating that LANS-α alters the nonlinear terms and increases the TKE. In the same manner as for [12, 13], numerical stability problems were encountered due to this injection of energy. Our simulations using a 5-point filter were stable according to the starting time of the simulation while those performed with 7- and 9-point filters were always stable as long as a constant artificial viscosity of $0.2\,\mathrm{m}^2/\mathrm{s}$ was added. When LANS-α modeling was activated, the 120 m-spatial resolution simulations, i.e. Rank1, produced more TKE than the simulations at 30 m, i.e. Rank2, without LANS-α. This increase in energy was induced by interactions between the tidal flow and the bottom features, which are very well represented here by the high-resolution bathymetry of [10], particularly along the shallowest isobaths of the domain studied (Fig.s 3 and 4, white and light gray contours).

The LANS-α modeling, which is more expensive than MARS for the same spatial resolution (see Table 3), appears to be interesting for the coastal numerical modeler when we compare the TKE computed at different resolutions. Between Run 1 and 3, the computing time for Rank1 is enhanced 2.8-fold with LANS-α. In contrast, MARS-α for Rank1 (120 m) is 30 percent faster than MARS for Rank2 (30 m). Indeed to obtain Rank2 simulations, the hydrodynamics of Rank0 and Rank1 needed to be computed in order to generate boundary conditions for the velocity and the sea surface elevation, which increased the computing time.

TKE decays were examined through wave-number energy spectra computed from barotropic (2D) velocity fluctuations for ebb and flood tides and for a dynamically consistent area on which they were integrated (see Fig. 5). For the largest scale (wave number smaller than $2\,\mathrm{km}^{-1}$), the level and the decay of TKE were similar for MARS and MARS-α. As suggested by [3], [17] and [14] for the two-dimensional turbulence, we observed two types of

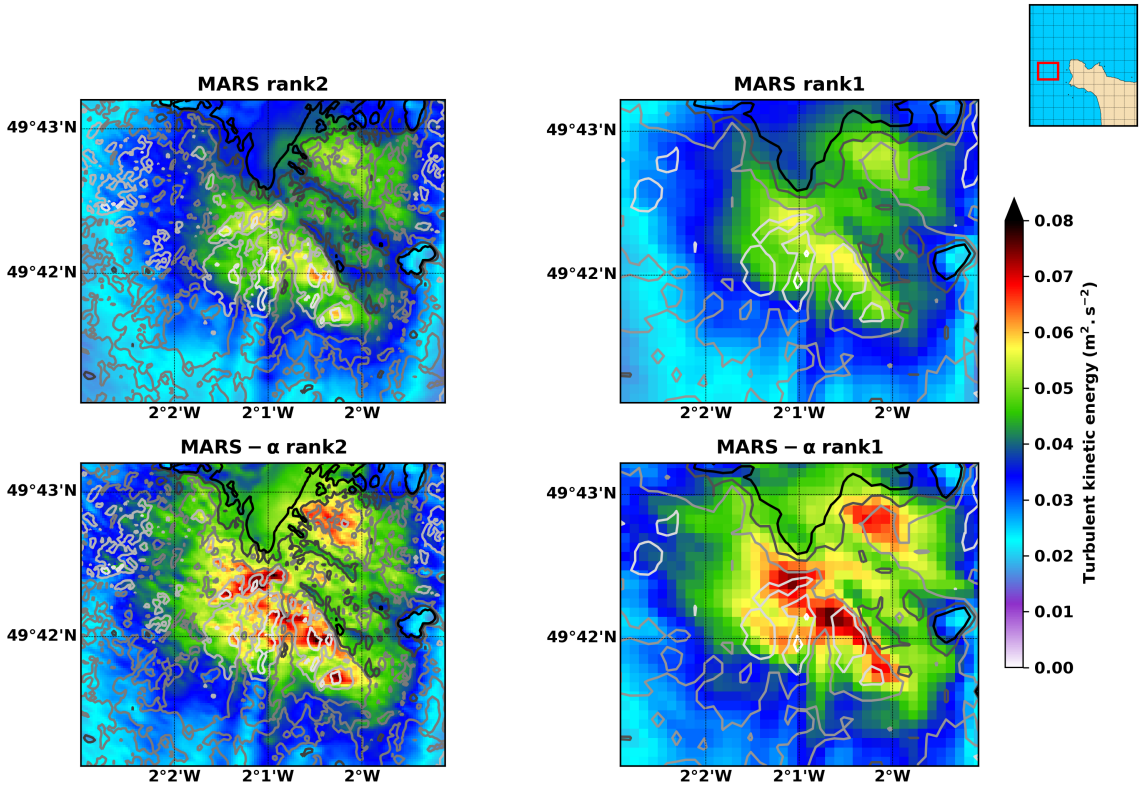

Fig. 3. Contours of TKE for the ebb tide for MARS (top row) and MARS-α (bottom row) simulations and for different spatial resolutions. Rank2 with a horizontal resolution of 30 m is on the left while Rank1 with a horizontal resolution of 120 m is on the right. Velocity fluctuations were computed over 4 hours around maximum velocity with a time step of 20 minutes. Bathymetric contours are shown with a grey scale, ranging from 20 meters (white) to 40 meters (black).

inertial range: (1) an energy transfer range where the turbulent energy spectrum is a function of $k^{-5/3}$ (k being the wave number), and (2) an enstrophy transfer range with a decay of the turbulent energy spectrum following k^{-3}. For case (1), energy was transferred from small (high wave numbers) to large (low wave numbers) scales while for case (2), enstrophy went from large to small scales. We observed similar TKE decays for MARS and MARS-α within inertial ranges. The cut-off wave number, which marks the beginning of the dissipative range in which a k^{-5}-decay is observed, was about 8 km^{-1}.

As explained in [12, 13], LANS-α requires the use of turbulent closure schemes for vertical mixing. The results of Run 1 and 3 previously, discussed above, were produced with constant vertical mixing to facilitate the understanding. However, coastal numerical modelers rightly prefer to employ RANS 2-equations turbulence scheme, such as $K-\epsilon$ [15] or the $K-Kl$ [21]. The most widely used in MARS is the $K-\epsilon$ with the modified coefficients of [30]. By solving one equation for the turbulent kinetic energy (K) and another for the turbulent dissipation (ϵ), the turbulent mixing length is obtained, and the vertical turbulent viscosity is then computed. As expected, a more intense turbulence is simulated by MARS and MARS-α with $K-\epsilon$ than with constant vertical mixing for ebb (see Fig. 6) and flood (see Fig. 7) tides. However, interestingly, MARS-α with $K-\epsilon$ produced a lower level of TKE than MARS with $K-\epsilon$. This contrasts with results obtained with constant vertical mixing and with earlier results of [12]. A plausible explanation may be the horizontal advection terms of the time-evolution equations for K and ϵ, which are not implemented in MARS and MARS-α [8]. These terms become significant when a horizontal turbulent transport occurs as near a strong bathymetric gradient, as shown in Fig. 6. Without this transport, the vertical turbulent mixing is overestimated by the model using $K-\epsilon$, as shown by comparison with the high-frequency radar measurements [20]. This loss of energy is more intense for ebb tide because of the angle between the direction of the tidal velocity (towards the

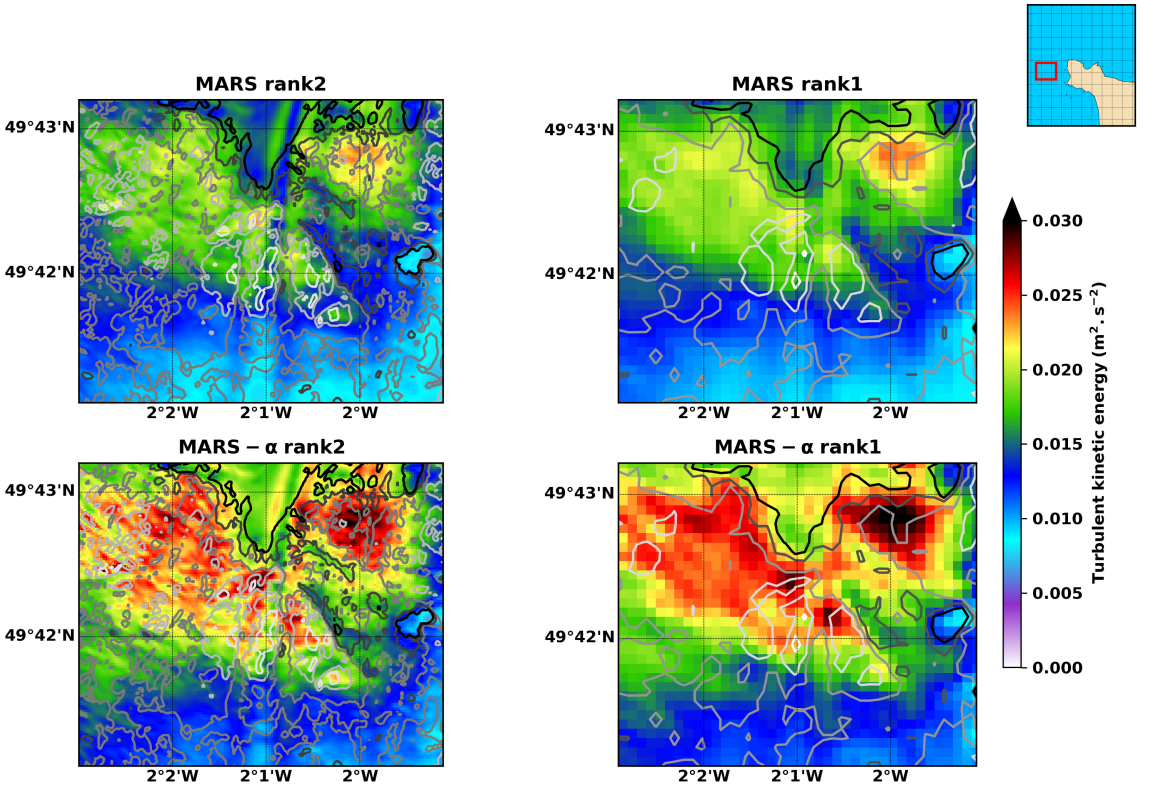

Fig. 4. Same as Fig. 3 but for the flood tide.

south-west) and the orientation of the bottom features, which enhances the horizontal transport. Thus, the reduction by LANS-α modeling seems realistic but needs to be confirmed by field measurements or large-eddy simulations using a non-hydrostatic solver.

The spatial structure of the eddy and idenfication of regions with different mixing properties are studied by means of the Okubo-Weiss (hereinafter OW) parameter [23, 31]. This parameter allows us to distinguish the relative importance of deformation and rotation on the production of turbulent eddies. Horizontal maps of OW for ebb (Fig. 8) and flood (Fig. 9) tide show that more turbulent structures are simulated with LANS-α for the same spatial resolution of 30 m. For Rank2 overall, we observed numerous turbulence structures due to deformation, represented by positive OW values. Differences between ebb and flood tide remain from the turbulence induced by rotation, represented by negative OW values. At ebb, these values are located along a circular arc following the northern coast of Cotentin (Fig. 8). At flood, most of the negative values are located next to the 40-m isobath because the current velocity direction is oriented towards the north-east (see Fig. 9). The small region of Rank2, discussed previously, contains

Table 3. Total number of grid points, simulated time (Sim. Time) in days and computing time (Comp. Time) in hours of each simulations.

	Rank0	**Rank1**	**Rank2**
Grid points	*649×475×20*	*938×916×20*	*850×957×20*
Sim. Time (d)	3	2.5	2.25
Comp. Time (h): Run 1	22.82	308.55	994.13
Comp. Time (h): Run 2	25.87	358.28	1154.83
Comp. Time (h): Run 3	9.35	110.57	362.93
Comp. Time (h): Run 4	12.53	158.45	484.80

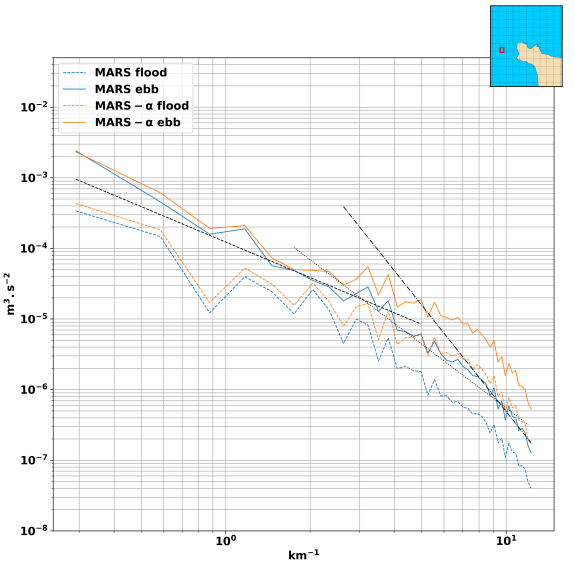

Fig. 5. TKE decay over the wave number k in logarithmic scale, for MARS and MARS-α during ebb and flood tide. Dashed, dotted and dashed-dotted lines are respectively for the curves TKE $= k^{-5/3}$, TKE $= k^{-3}$ and TKE $= k^{-5}$, respectively. Spectra were computed with barotropic velocity fluctuations relative to a time-mean 4 four hours and were averaged over time steps. They were computed on a dynamically consistent area, on which they were integrated, the dimensions of which were 1.71 km and 1.26 km in latitudinal and longitudinal directions, respectively.

mainly deformation-induced turbulent structures caused by the interactions with the bottom.

4. Conclusions and Perspectives

The LANS-α turbulence model was implemented for the first time within a hydrostatic coastal hydrodynamic model. We presented the details of implementation of the model and, in particular, the smoothing procedure, which is based on a convolution filter, the changes in the equations for the barotropic/baroclinic modes and the solving algorithm. Unfortunately, the efficiency of the ADI scheme implemented by [16] was was lost because of implementation of LANS-α, which involves solving equations along the direction of computation. MARS-α results were shown to be consistent with those of [12, 13] and [25]. MARS-α re-energized the flow and produced higher levels of TKE at a spatial resolution of 120 m than MARS at 30 m, leading to a 30% reduction in computing time. TKE decay was similar with or without LANS-α, with a decay following $k^{-5/3}$ and k^{-3} within the inertial subrange. When K-epsilon was used for vertical mixing, LANS-alpha was able to correct the misevaluation of TKE due to the particular implementation of the $K - \epsilon$ in MARS, which does not include the horizontal advection terms. The level of energy was lower with MARS-α in cases where the horizontal advection of TKE was significant. This is consistent with the conclusions of [20]. On the basis of the foregoing, it can be concluded that LANS-α produces satisfactory results and could be useful for the coastal modeler community. Our work needs to be pursued, however, in particular through comparisons with high-frequency oceanographic data, including current velocity magnitude and direction, which are necessary to validate the numerical model and to determine whether or not the LANS-α level of turbulence is consistent with the observations. Moreover, it is important to test these results against other numerical results computed with traditional sub-grid scale LES turbulence models [e.g 6, 7] to highlight the added value of LANS-α modeling compared to the other computational models of turbulence already in use.

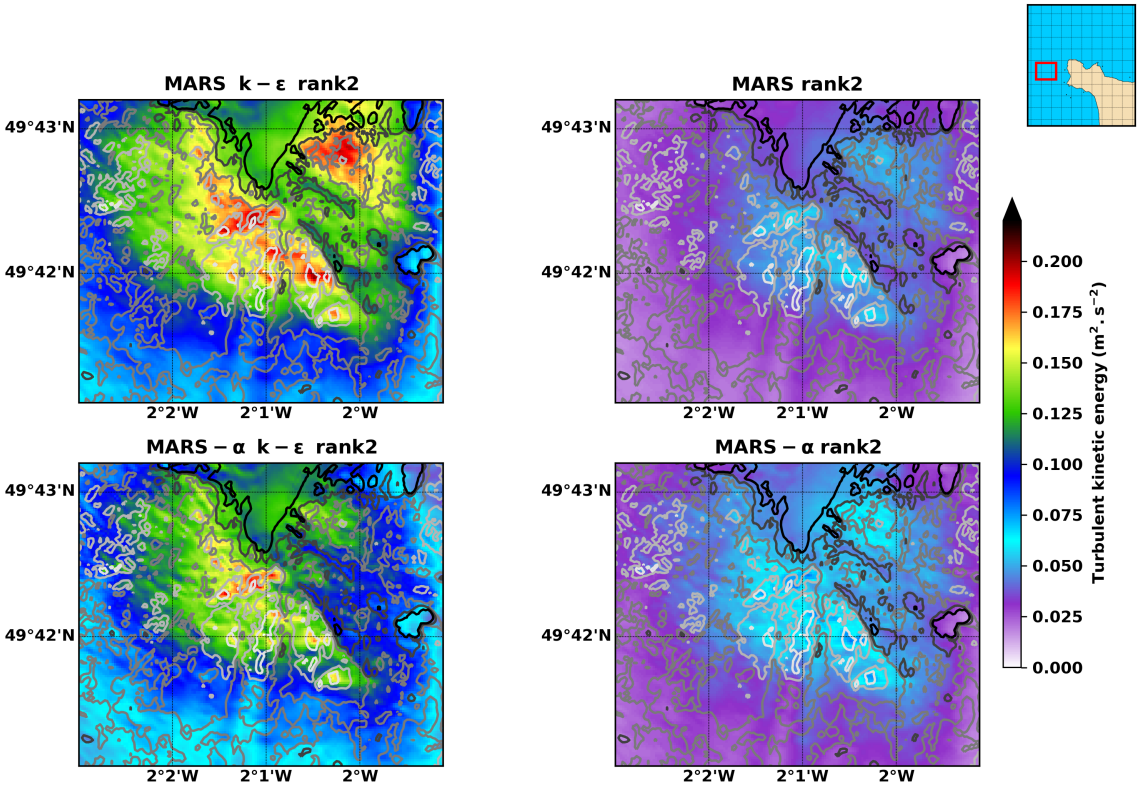

Fig. 6. Contours of TKE for the ebb tide for MARS (top row) and MARS-α (bottom row) simulations and for Rank2 at a horizontal resolution of 30 m. The simulations on the left used $K-\epsilon$ closure to model vertical mixing instead of constant vertical mixing as on the right.

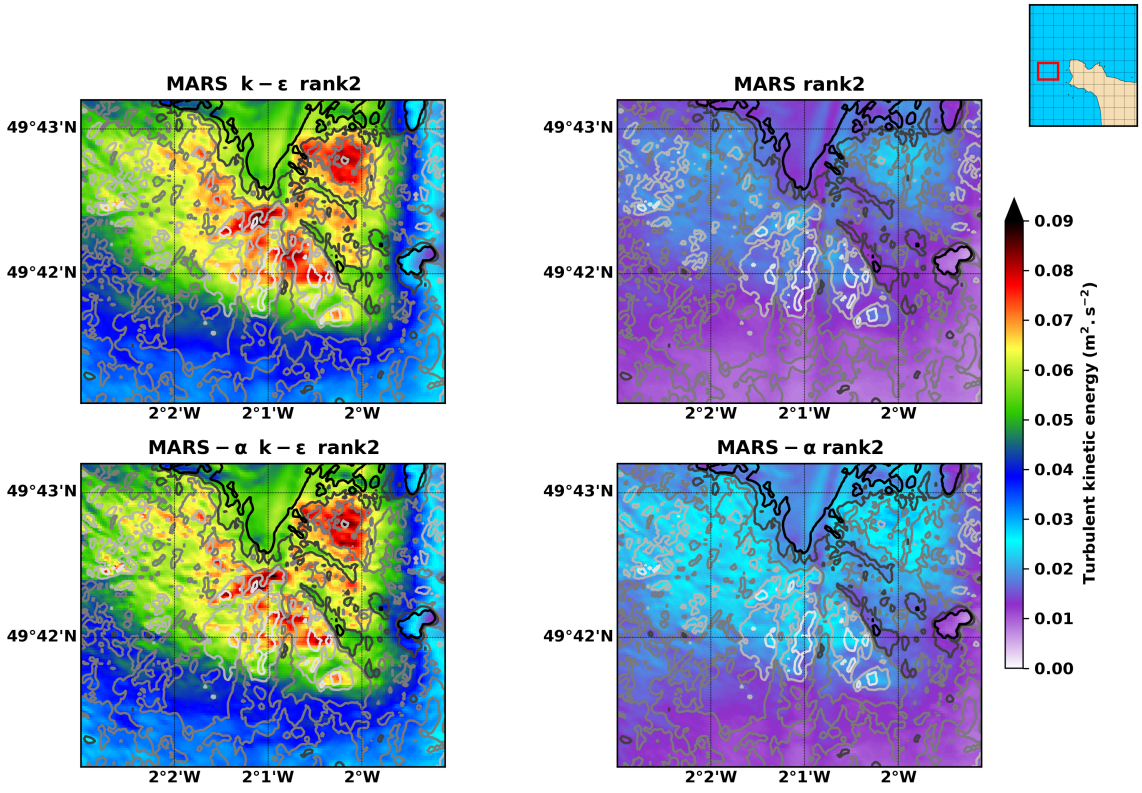

Fig. 7. Same as Fig. 6 but for the flood tide.

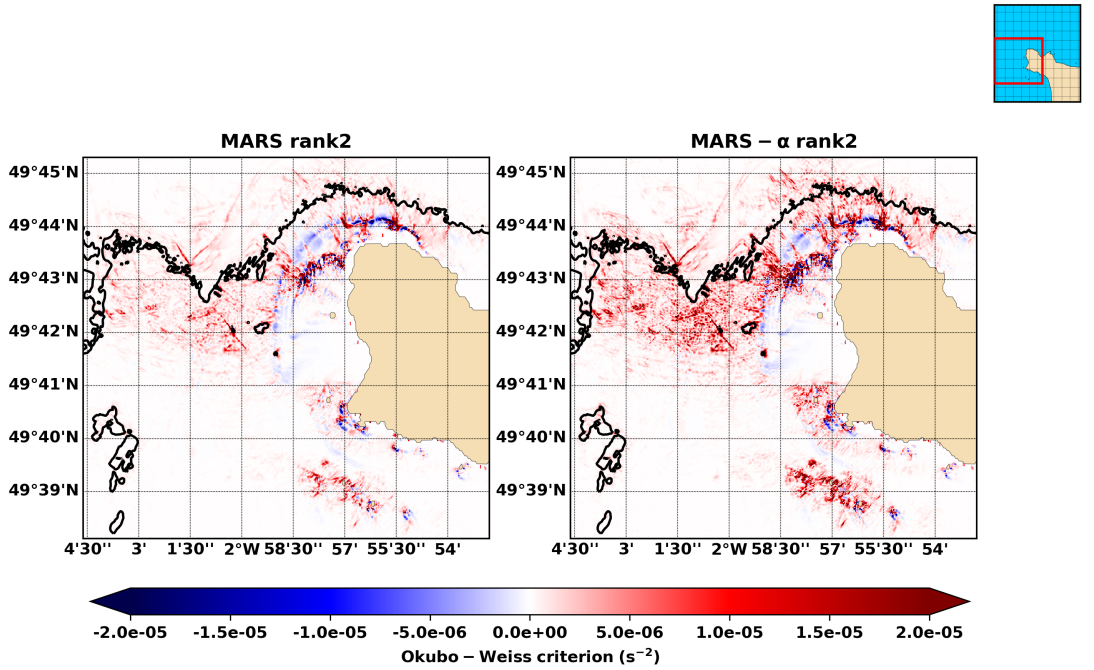

Fig. 8. Okubo-Weiss criterion computed with barotropic currents for the ebb maximum currents period: MARS Rank2 (left panel) and MARS-α Rank2 (right panel) with a constant vertical viscosity. Bathymetric contours are shown with a grey scale, ranging from 20 meters (white) to 40 meters (black).

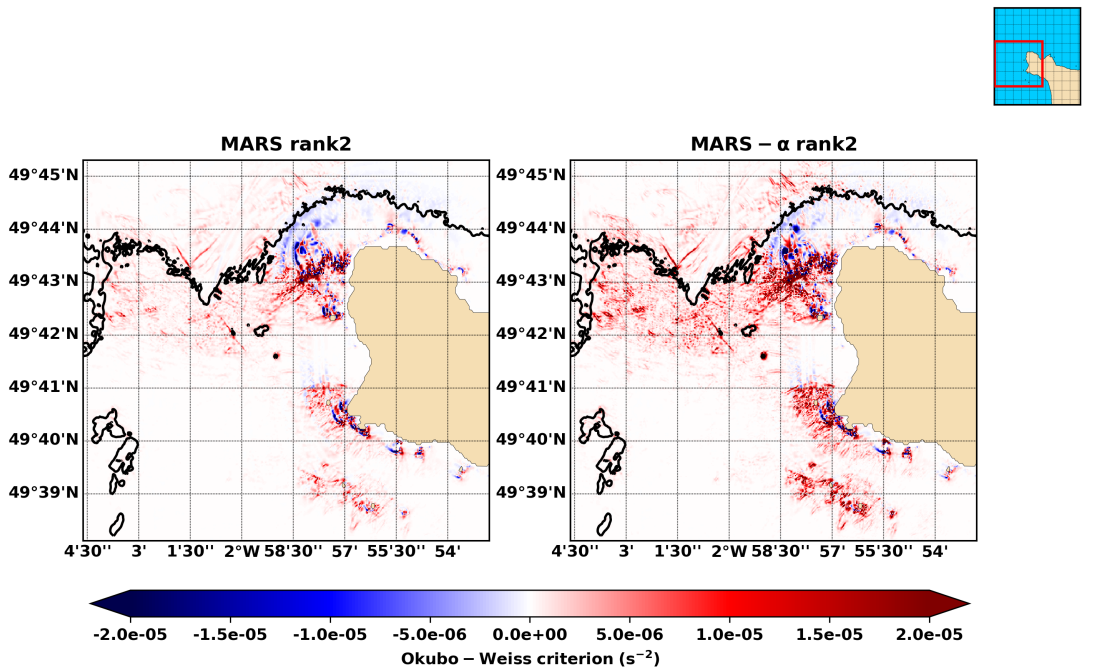

Fig. 9. Same as Fig. 8 but for flood tide.

Authors contribution

ACB: Conceptualization, Writing - original draft/review & editing, Methodology, Investigation, Supervision, Funding acquisition. **FA**: Writing - original draft & editing, Investigation, Development & Simulation, Software. **MB**: Writing - original draft/review & editing, Investigation, Simulation, Software. **FD**: Writing - review & editing, Investigation.

Competing interest

The authors declare no conflict of interest.

Acknowledgements

F. Adong acknowledges the post-doctoral grant of CNRS and France Energies Marines in the framework of HYD2M project (ANR-10-IEED-0006-07). A.-C. Bennis is supported by UNICAEN in the framework of HYD2M projet. M. Boutet is supported by CNRS. The authors thank P. Bailly Du Bois for providing the high-resolution bathymetry. Results acquired with Ifremer MARS software. The authors acknowledge the Centre Régional Informatique et d'Applications Numériques de Normandie (CRIANN) for providing the computational resources and support. The authors are grateful to two anonymous reviewers for their comments and suggestions.

Supplementary Material

The source code as well as the result files are available from the SEANOE platform (https://doi.org/10.17882/76523) for reviewers and editors only, MARS 3D being not an open-source code.

Appendix A. Additional informations about the MARS-α solving algorithm

Appendix A.1. ***Predictor phase***

Step 2: Barotropic part of the y-components: compute $\eta^{n+1/2}$, $\bar{v}_y^{n+1/2}$ and $\bar{u}_y^{n+1/2}$ by solving the following coupled system

$$\begin{aligned} \bar{v}_y^{n+1/2} + \theta_N \frac{g\Delta t}{2}\frac{\partial \eta^{n+1/2}}{\partial y} &= \bar{v}_y^{n} + \frac{\Delta t}{2}\left(\bar{G}_{\alpha y}^n + \frac{\tau_{Sy}^n}{\rho D_y^{n+1/2,*}}\right) \\ &- \frac{\Delta t}{2}\left(f_t v_{By}^n + \theta_P g \frac{\partial \eta^n}{\partial y}\right), \end{aligned} \tag{A.1}$$

$$\begin{aligned} \eta^{n+1/2} + \frac{\Delta t}{2}\frac{\partial}{\partial y}\left[D_y^{n+1/2,*}\left(\gamma_R \bar{v}_y^{n+1/2} + \gamma_S \bar{u}_y^{n+1/2}\right)\right] &= \eta^n - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^{n+1/2,*}\gamma_R \bar{v}_x^{n+1/2,*}\right) \\ &- \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^{n+1/2,*}\gamma_S \bar{u}_x^{n+1/2,*}\right), \end{aligned} \tag{A.2}$$

where v_{By} is the y-component of the rough velocity at the bottom.

$\bar{v}_x^{n+1/2,*}$ and $\bar{u}_x^{n+1/2,*}$ are estimated as:

$$\bar{v}_x^{n+1/2,*} = \bar{v}_x^n + \frac{\Delta t}{2}\left(\bar{G}_{\alpha x}^n + \frac{\tau_{Sx}^n}{\rho D_x^n} - \frac{\tau_{Bx}^n}{\rho D_x^n} - g\frac{\partial \eta^n}{\partial x}\right). \tag{A.3}$$

$$\boxed{\bar{u}_x^{n+1/2,*} = K_\alpha * \bar{v}_x^{n+1/2,*}} \tag{A.4}$$

Baroclinic part of the y-components: compute $v_y^{n+1/2}$ by solving the following equation (where u_{yk} and v_{yk} are the y-component of the smooth and the rough velocity for a given k index, respectively.)

$$v_y^{n+1/2} - \frac{\Delta t}{2}\frac{\partial}{\partial\sigma}\left(\frac{\nu_V}{\left(D_y^{n+1/2,*}\right)^2}\frac{\partial v_y^{n+1/2}}{\partial\sigma}\right) = v_{y\cdot} + \tfrac{\Delta t}{2}\left(G_y^n - g\frac{\partial\eta^n}{\partial x}\right). \tag{A.5}$$

$$\boxed{u_{yk}^{n+1/2} = \left(K_\alpha * v_{yk}^{n+1/2}\right), \quad k = 1, \cdots, \text{kmax}} \tag{A.6}$$

Appendix A.2. ***Corrector phase***

Step 3: Barotropic part of the y-components: compute $\eta^{n+1,*}$, $\bar{v}_y^{n+1}$ and $\bar{u}_y^{n+1}$ by solving the following coupled system

$$\begin{aligned}\bar{v}_y^{n+1} + \theta_N\frac{g\Delta t}{2}\frac{\partial\eta^{n+1,*}}{\partial y} &= \bar{v}_y^{n+1/2} + \frac{\Delta t}{2}\left(\bar{G}_{\alpha y}^{n+1/2} + \frac{\tau_{Sy}^{n+1/2}}{\rho D_y^{n+1}}\right)\\ &- \frac{\Delta t}{2}\left(f_t v_{By}^{n+1/2} + \theta_P g\frac{\partial\eta^{n+1/2}}{\partial y}\right),\end{aligned} \tag{A.7}$$

$$\begin{aligned}\eta^{n+1,*} + \frac{\Delta t}{2}\frac{\partial}{\partial y}\left[D_y^{n+1/2}\left(\gamma_R\bar{v}_y^{n+1} + \gamma_S\bar{u}_y^{n+1}\right)\right] &= \eta^{n+1/2} - \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^{n+1/2}\gamma_R\bar{v}_x^{n+1,*}\right)\\ &- \frac{\Delta t}{2}\frac{\partial}{\partial x}\left(D_x^{n+1/2}\gamma_S\bar{u}_x^{n+1,*}\right),\end{aligned} \tag{A.8}$$

where $\bar{v}_x^{n+1,*}$ are $\bar{u}_x^{n+1,*}$ are estimated as:

$$\bar{v}_x^{n+1,*} = \bar{v}_x^{n+1/2} + \frac{\Delta t}{2}\left(\bar{G}_{\alpha x}^{n+1/2} + \frac{\tau_{Sx}^{n+1/2}}{\rho D_y^{n+1/2}} - \frac{\tau_{Bx}^{n}}{\rho D_y^{n+1/2}} - g\frac{\partial\eta^{n+1/2}}{\partial x}\right). \tag{A.9}$$

$$\boxed{\bar{u}_x^{n+1,*} = K_\alpha * \bar{v}_x^{n+1,*}} \tag{A.10}$$

Baroclinic part of the y-components: compute v_y^{n+1} by solving the following equation

$$v_y^{n+1} - \frac{\Delta t}{2}\frac{\partial}{\partial\sigma}\left(\frac{\nu_V}{\left(D_y^{n+1/2}\right)^2}\frac{\partial v_y{}^{n+1}}{\partial\sigma}\right) = v_y^{n+1/2} + \tfrac{\Delta t}{2}\left(G_{\alpha y}^{n+1/2} - g\frac{\partial\eta^{n+1/2}}{\partial x}\right). \tag{A.11}$$

$$\boxed{u_{yk}^{n+1} = \left(K_\alpha * v_{yk}^{n+1}\right), \quad k = 1, \cdots, \text{kmax}} \tag{A.12}$$

References

[1] Adong, F. and A.-C. Bennis, 2019: Lans-α and leray turbulence models for coastal simulations: application to alderney race. *Proceedings of the 13th European Wave and Tidal Energy Conference (EWTEC 2019)*, Sustainable energy research Group, 1–6.

[2] Bailly Du Bois, P., F. Dumas, L. Solier, and C. Voiseux, 2012: In-situ database toolbox for short-term dispersion model validation in macrotidal seas, application for 2d-model. *Continental Shelf Research*, **36**, 63–82.

[3] Batchelor, G. K., 1969: Computation of the energy spectrum in homogeneous two-dimensional turbulence. *The Physics of Fluids*, **12 (12)**, II–233–II–239, doi:10.1063/1.1692443.

[4] Bennis, A.-C., P. Bailly Du Bois, F. Dumas, C. Lathuillère, F. Adong, and J.-F. Filipot, 2018: Towards a realistic numerical modelling of wave-current-turbulence interactions in alderney race. *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*, IEEE, 1–7.

[5] Bennis, A.-C., L. Furgerot, P. B. D. Bois, F. Dumas, T. Odaka, C. Lathuilière, and J.-F. cois Filipot, 2020: Numerical modelling of three-dimensional wave-current interactions in complex environment: Application to Alderney Race. *Applied Ocean Research*, **95**, 102 021, doi: https://doi.org/10.1016/j.apor.2019.102021.

[6] Bourgoin, A., 2019: Bathymetry induced turbulence modelling the alderney race site : regional approach with Telemac-LES. Ph.D. thesis, Normandie Université.

[7] Bourgoin, A., S. S. Guillou, J. Thiébot, and R. Ata, 2020: Turbulence characterization at a tidal energy site using large-eddy simulations: case of the alderneyrace. *Philosophical Transactions of the Royal Society A: mathematical, physical and engineering sciences*, 1–25.

[8] Dumas, F. and G. Langlois, 2009: MARS sientific model description. *Technical report, Ifremer*, 1–107, URL `http://www.ifremer.fr/docmars/html/_static/2009_11_22_DocMARS_GB.pdf`.

[9] Furgerot, L., P. Bailly Du Bois, Y. Méar, E. Poizot, and A.-C. Bennis, 2018: Velocity profile variability at a tidal-stream energy site (aldemey race, france): From short (second) to yearly time scales. *2018 OCEANS - MTS/IEEE Kobe Techno-Oceans (OTO)*, IEEE, 1–8.

[10] Furgerot, L., Y. Poprawski, M. Violet, E. Poizot, P. B. du Bois, M. Morillon, and Y. Mear, 2019: High-resolution bathymetry of the Alderney Race and its geological and sedimentological description (Raz Blanchard, northwest france). *Journal of Maps*, **15 (2)**, 708–718, doi:10.1080/17445647.2019.1657510.

[11] Geurts, B. J., 2014: Theory and application of regularization modeling of turbulence. *Turbulence and Interactions*, M. O. Deville, J.-L. Estivalezes, V. Gleize, T.-H. Lê, M. Terracol, and S. Vincent, Eds., Springer Berlin Heidelberg, Berlin, Heidelberg, 1–12.

[12] Hecht, M. W., D. D. Holm, M. R. Petersen, and B. A. Wingate, 2008: Implementation of the lans-α turbulence model in a primitive equation ocean model. *Journal of Computational Physics*, **227 (11)**, 5691 – 5716.

[13] Hecht, M. W., D. D. Holm, M. R. Petersen, and B. A. Wingate, 2008: The lans-α and leray turbulence parameterizations in primitive equation ocean modeling. *Journal of Physics A: Mathematical and Theoretical*, **41 (34)**, 344 009.

[14] Kraichnan, R. H., 1967: Inertial ranges in two dimensional turbulence. *Physics of Fluids*, **10 (7)**, 1417–1423.

[15] Launder, B. and D. Spalding, 1974: The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, **3 (2)**, 269 – 289.

[16] Lazure, P. and F. Dumas, 2008: An external-internal mode coupling for a 3d hydrodynamical model for applications at regional scale (MARS). *Adv. Water Resources*, **31**, 233–250.

[17] Leith, C. E., 1968: Diffusion approximation for two-dimensional turbulence. *The Physics of Fluids*, **11 (3)**, 671–672, doi:10.1063/1.1691968.

[18] Leroy, R. and B. Simon, 2003: Réalisation et validation d'un modèle de marée en Manche et dans le golfe de Gascogne - Application à la réalisation d'un nouveau programme de réduction de sondages bathymétriques. Tech. rep., Shom.

[19] Lewis, M., S. Neill, P. Robins, M. Hashemi, and S. Ward, 2017: Characteristics of the velocity profile at tidal-stream energy sites. *Renewable Energy*, 1–15.

[20] Lopez, G., A.-C. Bennis, Y. Barbin, A. Sentchev, L. Benoit, and L. Marié, 2020: Surface currents in the alderney race from high-frequency radar measurements and three-dimensional modelling. *Philosophical Transactions of the Royal Society A: mathematical, physical and engineering sciences*, 1–25.

[21] Mellor, G. L. and T. Yamada, 1982: Development of a turbulence closure model for geophysical fluid problems. **20 (C2)**, 851–875.

[22] Mercier, P., M. Grondeau, S. Guillou, J. Thiébot, and E. Poizot, 2020: Numerical study of the turbulent eddies generated by the seabed roughness. Case study at a tidal power site. *Applied Ocean Research*, **97**, 102 082.

[23] Okubo, A., 1970: Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences. *Deep Sea Research and Oceanographic Abstracts*, **17 (3)**, 445 – 454.

[24] Okubo, A., 1971: Oceanic diffusion diagrams. *Deep Sea Research and Oceanographic Abstracts*, **18 (8)**, 789 – 802.

[25] Petersen, M. R., M. W. Hecht, and B. A. Wingate, 2008: Efficient form of the lans-α turbulence model in a primitive-equation ocean model. *Journal of Computational Physics*, **227 (11)**, 5717 – 5735.

[26] Thiébaut, M., J.-F. Filipot, C. Maisondieu, G. Damblans, R. Duarte, E. Droniou, N. Chaplain, and S. Guillou, 2020: A comprehensive assessment of turbulence at a tidal-stream energy site influenced by wind-generated ocean waves. *Energy*, **191**, 116 550.

[27] Thorpe, S. A., 2005: *The Turbulent Ocean*. Cambridge University Press, doi:10.1017/CBO9780511819933.

[28] Tsutsumi, E., et al., 2020: Vertical fluxes of nutrients enhanced by strong turbulence and phytoplankton bloom around the ocean ridge in the luzon strait. *Scientific Reports*, **10**, 17 879.

[29] Umlauf, L. and H. Burchard, 2003: A generic length-scale equation for geophysical turbulence models. *Journal of Marine Research*, **61 (2)**, 235–265.

[30] Warner, J. C., C. R. Sherwood, H. G. Arango, and R. P. Signell, 2005: Performance of four turbulence closure models implemented using a generic length scale method. *Ocean Modelling*, **8 (1-2)**, 81–113.

[31] Weiss, J., 1991: The dynamics of enstrophy transfer in two-dimensional hydrodynamics. *Physica D: Nonlinear Phenomena*, **48 (2)**, 273 – 294.