MERSEA IP

Marine EnviRonment and Security for
the European Area - Integrated Project

Free surface and variable volume in the NEMO code

Ref: MERSEA-WP09-CNRS-STR-03-1A
[22.5.2007]

AUTHORS: Bruno Levier, Anne-Marie Treguier, Gurvan Madec and Valerie Garnier

AFFILIATION: Laboratoire de Physique des Oceans, Brest

Co-ordinator:

Ifremer

Institut Français de Recherche pour l'Exploitation de la Mer - France

Contents

Introduction

Historically, the NEMO system (Madec *et al.* 1998) was built for climate studies using strong hypothesis on the dynamics, such as the "rigid lid" approximation, which eliminates the fast gravity waves. A first improvement (Roullet and Madec 2000) was made by implementing a free surface while filtering the fast external gravity waves, which allows one to keep a large time step. Roullet and Madec (2000) compared a linear formulation (with fixed volume) with a non-linear one (with a variable first level thickness), and concluded that the linear formulation was the best compromise for climate (the linear formulation does not strictly conserve the ocean salt content unlike the non-linear formulation, but the differences were not large enough to play a large role in the experiments of Roullet and Madec (2000)).

People interested in regional and/or coastal studies must simulate rapidly phenomena with short spatial scales, such as tidal waves. With the filtering scheme of Roullet and Madec (2000), these phenomena are very rapidly damped, or not simulated at all (Talandier *et al.* 2003). That's why a time-splitting scheme has recently been implemented, which allows the representation of the fast dynamics, while keeping a large time step (Bessières 2003). But this is not sufficient: although the free surface is implemented, the vertical grid of the model is fixed in time. This fact does not allow a good representation of tidal waves, and as mentioned above it precludes an exact conservation of the salt content. This is why the decision was made, as part of the MERSEA european project, to implement a variable volume in the NEMO system.

In this report, we first review the representation of the free surface in the NEMO system, with the different schemes available at the present time (chapter1). Then we introduce the equations of the free surface and the implementations induced by the variable volume (chapter 2). We present two kind of experiments in order to investigate the behavior of the model with the variable volume implemented; chapter 3 focusses on the conservation of salt content, and chapter 4 on the representation of a gravity wave.

Chapter 1

Kelvin waves in a channel

Before implementing a variable volume option, it is necessary to understand how the free surface equation is handled in the code. Three schemes are available: explicit, filtered and time-splitting. The explicit scheme is the simplest but the presence of fast barotropic waves requires a very small time step (of order of a few seconds). The filtered free surface scheme (Roullet and Madec 2000) uses an additionnal force directly in the primitive equations to filter External Gravity Waves (EGW). This force cancels the propagation of EGWs which have a frequency higher than a cutoff frequency, allowing the possibility to keep a relative large time step. The time-splitting scheme Bessières (2003) splits the fast barotropic part and the slow baroclinic part of the dynamics.

Talandier *et al.* (2003) studied the properties of the filtered scheme compared to the explicit scheme. Bessières (2003) compared the filtered scheme and the time-splitting scheme. These two studies considered an initial-value problem: a Kelvin wave solution was initialized in a periodic channel and left to propagate without non linearities nor friction, and the analytical solution was compared with the numerical simulation. This case is not relevant to tides in regional models, where the tidal waves are forced at the open boundaries. In this report, we perform the Kelvin wave experiment with open boundary conditions. We also reproduce the case of periodic boundary conditions, to allow a direct comparison of the three schemes.

1.1 The barotropic Kelvin wave

1.1.1 Analytical solution

This section is repeated for completeness from the report of Talandier *et al.* (2003). We detail here the analytical solution for a non-dispersive Kelvin wave in a zonal channel of width equal to b.

Considering the following hypotheses: homogeneous fluid, hydrostatic approximation, $\eta \ll H$, linearization, flat bottom, f plan, no forcing and no dissipation; we have the following shallow water equations:

$$\frac{\partial u}{\partial t} - fv = -g\frac{\partial \eta}{\partial x} \qquad (1.1)$$

$$\frac{\partial v}{\partial t} + fu = -g\frac{\partial \eta}{\partial y} \qquad (1.2)$$

$$\frac{\partial \eta}{\partial t} + H\nabla.\mathbf{u} = 0 \qquad (1.3)$$

where **u** = (u,v) is the barotropic velocity; f the Coriolis parameter; g the gravity; η the free surface against a zero reference level and H the total channel depth.

If we set $\{\eta,u,v\}(x,y,t) = \mathbf{RE}\{(\Phi,U,V)(x,y)e^{-i\omega t}\}$, we can have the V expression according to Φ, replacing the time derivative term of u in 1.1 with its expression obtained in taking the time derivative of 1.2. Doing the same thing for U, we get the following expressions:

$$U(x,y) = \frac{i\omega g}{(f^2-\omega^2)}\left(\partial_x\Phi + \frac{if}{\omega}\partial_y\Phi\right) \tag{1.4}$$

$$V(x,y) = \frac{i\omega g}{(f^2-\omega^2)}\left(\partial_y\Phi - \frac{if}{\omega}\partial_x\Phi\right) \tag{1.5}$$

Moreover, eliminating u and v componants from equations 1.1 and 1.2 with the continuity equation 1.3, we get the EGWs propagation equation:

$$\left(\partial_t^2 + f^2\right)\eta = c_0^2\nabla^2\eta \tag{1.6}$$

with the phase velocity $c_0 = \sqrt{gH}$. As we are in a zonal channel, we can re-write $\Phi(x,y) = E(y).e^{ikx}$ and substitute η in 1.6 with this writing. We obtain the following second order equation for E(y):

$$E''(y) - \gamma^2 E(y) = 0 \quad \text{with} \quad \gamma^2 = k^2 + \frac{f^2-\omega^2}{c_0^2} \tag{1.7}$$

The real solution for this differential equation is:

$$E(y) = A.e^{\gamma y} + B.e^{-\gamma y} \quad \text{such as} \quad \Phi(x,y) = \left(A.e^{\gamma y} + B.e^{-\gamma y}\right)e^{ikx}$$

Applying the boundary condition of zero normal velocity along the North and South boundary (y=0 and y=b) of the channel, whatever the time step; we have the following system from the equation 1.5:

$$A(\gamma + \frac{f}{c_0}) + B(\frac{f}{c_0} - \gamma) = 0 \tag{1.8}$$

$$A.e^{\gamma b}(\gamma + \frac{f}{c_0}) + B.e^{-\gamma b}(\frac{f}{c_0} - \gamma) = 0 \tag{1.9}$$

The non-dispersive wave solution is obtained in setting for instance $\gamma^{-1} = -c_0/f = R_O$ the external Rossby radius of deformation which implies B = 0. So the function Φ is writting as:

$$\Phi(x,y) = A.e^{-\frac{y}{R_0}}e^{ikx} \tag{1.10}$$

Finally one deducts the analytical solution for a barotropic kelvin wave in a zonal channel:

$$\eta(x,y,t) = A.e^{-\frac{y}{R_0}}\cos(kx-\omega t) \tag{1.11}$$

$$u(x,y,t) = \sqrt{\frac{g}{H}}\,\eta(x,y,t) \tag{1.12}$$

$$v(x,y,t) = 0 \tag{1.13}$$

1.1.2 The Kelvin amphidromy

In our periodic zonal channel configuration, we simulate the dominant semi-diurnal wave M2 (period $T_k = 44700$ s i.e 12h25') using 2 barotropic Kelvin waves; one propagates eastward (K1) and one westward (K2) respectivly along the South and North boundary. The channel, centered on 45° N, has a zonal and meridional length of 1000 km and 600 km.

Using equations 1.11 and 1.12 we form the following solution:

$$\begin{aligned} \eta(x,y,t) &= A.[e^{-\frac{y}{R_0}}\cos(k_0x-\omega_0 t)+e^{\frac{y}{R_0}}\cos(k_0x+\omega_0 t)] && (1.14)\\ u(x,y,t) &= A.\sqrt{\frac{g}{H}}\,[e^{-\frac{y}{R_0}}\cos(k_0x-\omega_0 t)-e^{\frac{y}{R_0}}\cos(k_0x+\omega_0 t)] && (1.15)\\ v(x,y,t) &= 0 && (1.16) \end{aligned}$$

where $k_0 = \frac{2\pi}{\lambda_k}$, $\omega_0 = \frac{2\pi}{T_k}$, $\lambda_k = \sqrt{gH}.T_k$ and $R_0 = \frac{\sqrt{gH}}{f}$ are respectively the wavenumber, the pulsation, the wavelength and the external Rossby radius for the Kelvin wave. The signal and domain characteristics are summarized in Table 1.1.

M2 wave	Channel domain	Parameters
$T_k = 44700$ s	$L_z = 1000$ km	$\Omega = 7.2910^{-5}\ s^{-1}$
$\lambda_k = 1000$ km	$L_m = 600$ km	$\phi = 45°$ N
A = 0.5 m	H = 51.03 m	$f = 2.\Omega\sin\phi$
$R_0 \approx 217$ km		g = 9.81 m.s^{-2}

Table 1.1: Characteristics of the M2 wave and the geometry of the channel.

We notice that the depth H of the channel was calculated while fixing the wavelength λ_k and the wave period T_k repectively to 1000 km and 44700 seconds.

For these characteristics, Figures 1.1 and 1.2 show respectively the Kelvin amphidromy and the analytical solution for the SSH at four time steps.

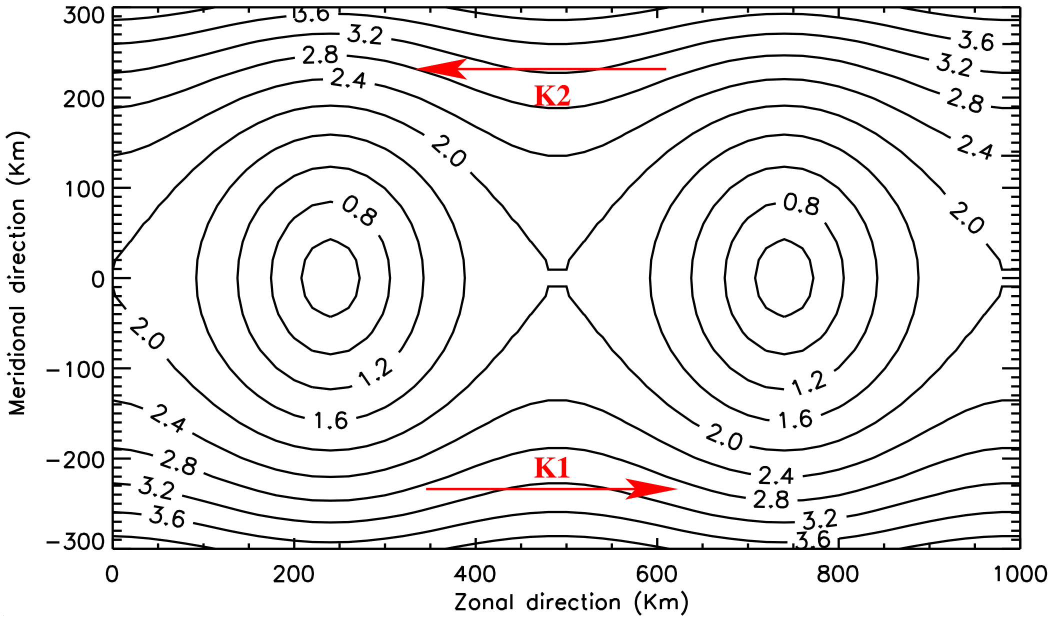


Figure 1.1: Co-ranges lines and amphydromic points for the propagation of 2 Kelvin waves in a zonal channel.

1.2 Numerical simulations with NEMO

1.2.1 The free surface treatment

For the filtered free surface, an additional force is added in momentum equations of the model. This force only dissipates phenomena with timescale lower than $2\pi T_c$ with T_c the cutoff time parameter set to $2\Delta t$ after a stability analysis (Δt is the time step model) and normally does not influence larger timescales.

The time-splitting scheme exploits the time scale difference between barotropic and baroclinic gravity waves. It is possible to approximate the fast mode by fluctuations of the depth averaged fluid, and the slow modes by deviations from the depth average (Griffies *et al.* 2000). Barotropic equations are integrated with gravity waves resolved using small time steps. The barotropic sub-cycle is time averaged.

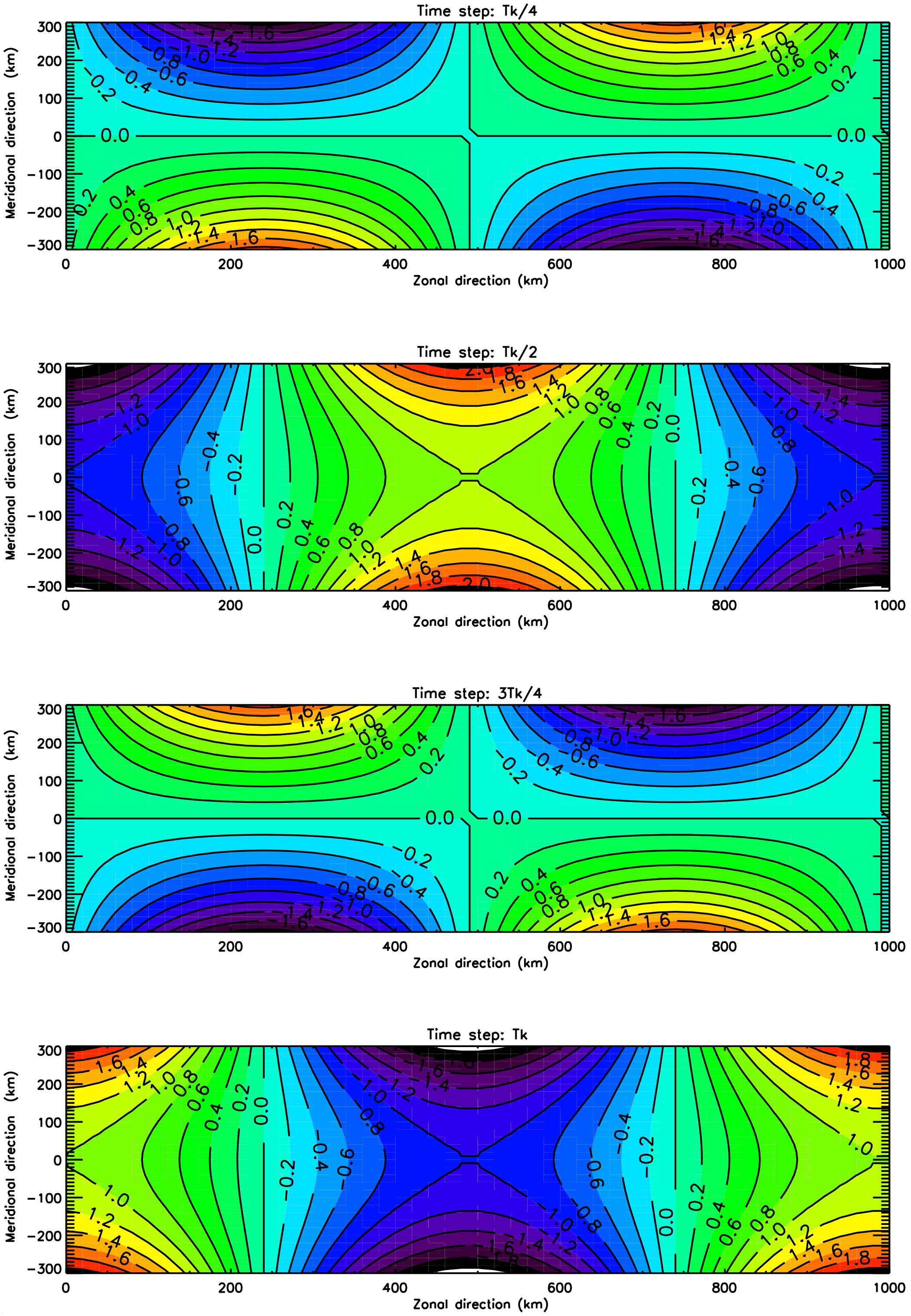
Our analytical solution being quite linear, we have deleted all dissipation and advective terms in OPA momentum equations so they are reduced to (without any forcing):

$$\partial_t \mathbf{u} + f \wedge \mathbf{u} = -\frac{1}{\rho}\nabla P_h - g\nabla\eta - T_c\nabla\partial_t\eta \tag{1.17}$$

where P_h is the hydrostatic pressure and the last term in the right hand side is the additional force. The free surface equation is (no forcing here):

$$\partial_t \eta = -\nabla.(H\mathbf{U}) \tag{1.18}$$

with **U** the vertically integrated velocity. We notice that the free surface is linear in OPA (cf. Roullet and Madec (2000))


Time step: Tk/4

Time step: Tk/2

Time step: 3Tk/4

Time step: Tk

Figure 1.2: Analytical SSH (m) for the barotropic Kelvin waves propagation during one period T_k.

1.2.2 Model configuration

Our configuration is a periodic zonal channel with the following number of points, N_x=52, N_y=33 and N_z=2. The horizontal resolution in both direction is equal to 20 km so that the domain dimensions are 1000 x 600 km. The wavelength λ_k and the meridional characteristic length R_0 are quite well sampled with this spatial step.

The homogeneous fluid is set to a temperature of 10° C and a salinity of 35.5 p.s.u. The initial dynamical state is setting with the analytical equations (14)-(16) at time t=0 and the parameters defined in Table 1.1. We must emphasize that after this time, all simulations are free, i.e with no forcing.

1.2.3 Simulations

We tested the behavior of the filtering force and time-splitting scheme for two values of the time-step.

To compare numerical simulations to their analog analytical solution, we have calculated the root mean square errors (RMS errors) over all the domain with the following method:

$$RMS = \left(\frac{\sum_{dom} \{\eta_s - \eta_a\}^2}{\sum_{dom} \eta_a^2} \right)^{1/2}$$

where η_s and η_a are respectively the numerical and analytical SSH, $\sum_{dom}$ represent the summation over all the domain, i.e over i=1,N_x and j=1,N_y.

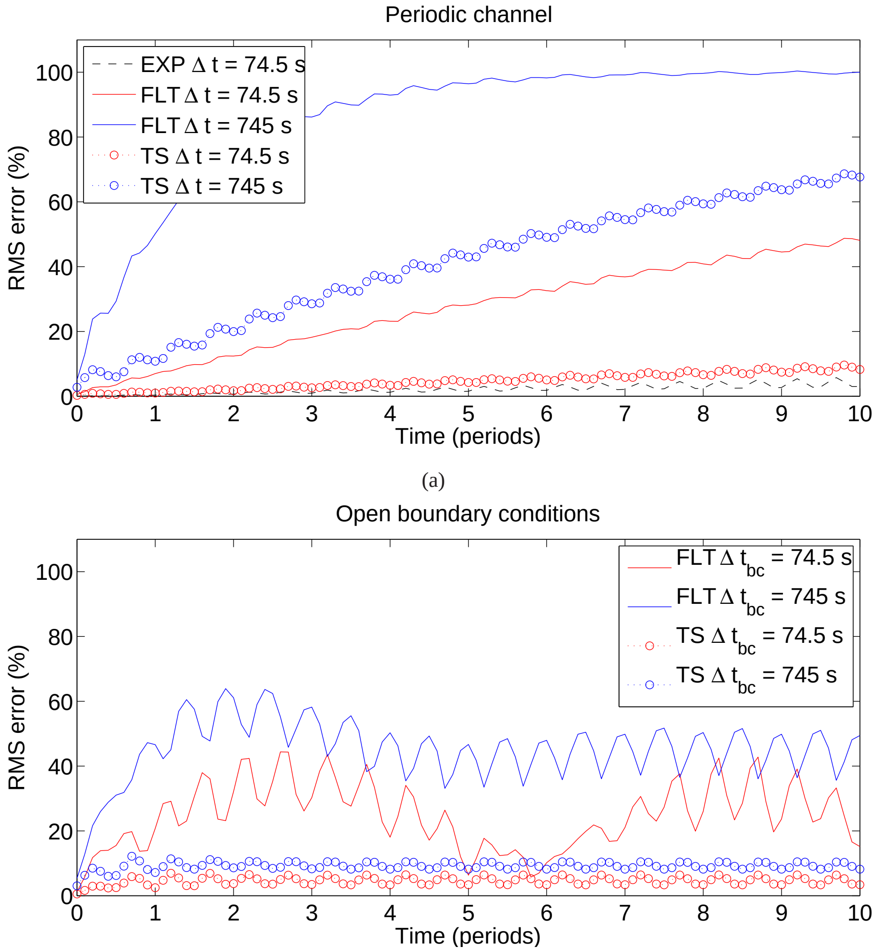
The Figure 1.3(a) shows the rms error calculated in periodic boundary conditions for the explicit scheme with Δt = 74.5 s, for the filtered scheme with Δt = 74.5 s and Δt = 745 s, and for the time-splitting scheme with Δt = 74.5 s and Δt = 745 s (with a barotropic time step of Δt_{bt} = 74.5 s). The Kelvin waves are completly canceled over the 10 periods of the run for the simulation with the filtered scheme and Δt = 745 s, while the rms error reached 70% with the time-splitting scheme and Δt = 745 s. With Δt = 74.5 s, the rms error was 50% for the filtered scheme, 10% for the time-splitting scheme and 6% for the explicit scheme.

According to Talandier *et al.* (2003) the loss of the tidal signal is strongly related to the additional force, because of there is non linear advective terms in the NEMO momentum equations and the setting of the horizontal and vertical diffusion coefficients are set to zero in both dynamical and tracer equations. The simulation with the explicit scheme is indeed the less diffusive (errors are generated by the spatial resolution, a more accurate spatial discretization will decrease them). The time-splitting case averages the values calculated in the barotropic loop; the tidal signal is therefore damped.

With the open boundary conditions, the rms error for the filtered scheme varies between 10% and 40% with Δt = 74.5 s, between 35% and 65% with Δt = 745 s. The rms error for the time-splitting scheme varies between 3% and 6% with Δt = 74.5 s, and between 8% and 11% with Δt = 745 s.

We conclude that the time-splitting scheme is well adapted to simulate forced tidal waves. The additional force is too diffusive, even with a periodic forcing at the open boundaries.

However, even with the time-splitting scheme, a large error will result if the domain is large and the tidal wave has to travel many wavelenght away from its forcing region, as shown in the periodic case.



(a)

(b)

Figure 1.3: RMS error of the sea surface height for two baroclinic time-steps (Δt = 74.5 s and Δt = 745 s) and three time-stepping schemes. The time-splitting scheme is better than the filtered scheme **(a) periodic channel**: explicit scheme with Δt = 74.5 s, filtered scheme with Δt = 74.5 s and Δt = 745 s; time-splitting scheme with Δt = 74.5 s and Δt = 745 s (the barotropic time step Δt_{bt} is 74.5 s) **(b) open boundary conditions**: filtered scheme with Δt = 74.5 s and Δt = 745 s; time-splitting scheme with Δt = 74.5 s and Δt = 745 s (the barotropic time step Δt_{bt} is 74.5 s)

Chapter 2

Method, equations

The representation of nonlinearities in the free surface equation requires the vertical grid to change with time (variable volume). The discretization of the variable volume is determined by volume and energy constraints (Griffies 2004). The variable volume computed from a nonlinear free surface equation ensures perfect conservation of ocean salt content. The fresh water flux modifies the sea surface elevation and the thicknesses of the model layers, and consequently the model volume.

The improvement involves two major modifications by allowing layer thicknesses to be time-varying and by computing a nonlinear free surface equation.

2.1 Equations

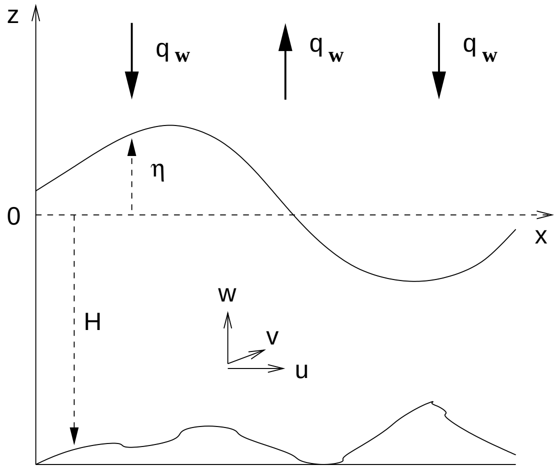


Figure 2.1: The free surface. η is the deviation of the sea surface from rest. *H* is the depth of the bottom. q_w is the freshwater forcing (positive if evaporation is greater than precipitation, otherwise negative).

Notations:

- $u(x,y,z)\,,v(x,y,z)$: horizontal components of the velocity, $= u_h$,
- $w(x,y,z)$: vertical com,ponent of the velocity,

- $\eta(x,y)$: deviation of the sea surface from rest (sea surface height),
- $H(x,y)$: depth of the bottom,
- q_w: fresh water forcing, in units of velocity (volume per unit time per unit area), crossing the ocean surface.

The kinematic boundary conditions define the ocean domain and describe its volume budget. The sea floor is described by specifying a surface with no normal flow.

The bottom of the ocean is a material surface and the kinematic boundary condition is the no-normal flow condition, wich implies

$$w = -\mathbf{u}_h.\nabla_h H = -u\frac{\partial H}{\partial x} - v\frac{\partial H}{\partial y} \quad \text{at} \quad z = -H(x,y)$$

The sea surface is defined by means of an equation of motion of the sea surface height, and the ocean surface is generally permeable to fresh water fluxes.

$$\frac{d\eta}{dt} = w - q_w \quad \Rightarrow \quad w = \frac{\partial\eta}{\partial t} + \mathbf{u}_h.\nabla_h\eta - q_w \quad \text{at} \quad z = \eta(x,y)$$

The surface height $z = \eta$ has a time tendency determined by an advective flux of height, the Eulerian vertical velocity and the fresh water velocity (Pacanowski and Griffies 2000). The presence of horizontal advection of the free surface height makes this equation nonlinear.

Knowledge of the surface currents and fresh water flux q_w allow one to time step the free surface height through use of the surface kinematic boundary condition. However, because the motion of the free surface height is associated with fast barotropic motions, it is more useful algorithmically to determine η within the barotropic system. Additionally, a direct discretization of the surface kinematic boundary condition would require a discretization of the advective term.

Instead of directly discretizing the kinematic boundary condition, perform a vertical integral of the continuity equation over the full depth of the ocean to find

$$\int_{-H}^{\eta} div\,\vec{u}\;dz = \int_{-H}^{\eta}\frac{\partial w}{\partial z} + \int_{-H}^{\eta}\frac{\partial u}{\partial x} + \int_{-H}^{\eta}\frac{\partial v}{\partial y} = 0$$

$$\Rightarrow \quad w(z=\eta) - w(z=-H) = \int_{-H}^{\eta}\frac{\partial u}{\partial x} + \int_{-H}^{\eta}\frac{\partial v}{\partial y}$$

Using the surface and bottom boundaries conditions, one obtain

$$\boxed{\frac{\partial\eta}{\partial t} = -\nabla_h U + q_w} \quad \text{where} \quad \nabla_h U = \frac{\partial}{\partial x}\int_{-H}^{\eta} u\,dz + \frac{\partial}{\partial y}\int_{-H}^{\eta} v\,dz$$

The time tendency of the free surface height is determined by the convergence of the vertically integrated transport plus the fresh water flux through the sea surface.

In the **fixed volume case**, the sea surface height η is supposed to be small relative to the total depth H. This hypothesis leads to the definition of the approximated integrated transport U_0. The upper boundary condition is applied at $z = -H$ and $z = 0$. The time tendency of the free surface height is written

$$\frac{\partial\eta_0}{\partial t} = -\nabla_h U_0 + q_w \quad \text{where} \quad U_0 = \int_{-H}^{0} u_h\,dz$$

In the **variable volume** case, the integrated transport is calculated on the full water column:

$$U = \int_{-H}^{\eta} u_h\,dz$$

2.2 Implementation

In Roullet and Madec (2000) only the first level thickness varied. This limits the applications, because with strong tidal amplitude the tidal elevation can be larger that the reference thickness of the first level equations. In our case, all the levels vary. The sea surface elevation amplitude is distributed at each vertical level, at each time step (see A.1 for the algorithm and the new module added to the code).

Because of the variations of the grid in time, the expression of the vertical velocity is changed (see A.8).

With the variable volume, the cell grid of a given level are no longer at the same depth, even in z-coordinates. So the calculation of the hydrostatic pressure gradient must always include a correction, as in s-coordinates (see A.4). Moreover, this calculation is not the same as in fixed volume case, due to simplifications that do not occur in the variable volume case. The hydrostatic pressure gradient is now calculated from the bottom to the surface (and no longer to the surface of the ocean at rest), so that the surface pressure gradient is also included in this calculation. Therefore the surface pressure gradient no longer needs to be calculated in a separate manner. With the time-splitting scheme, this implies to remove the surface pressure gradient (implicitly calculated in the baroclinic part) before estimating the barotropic part; the surface pressure gradient is calculated in the barotropic part, as in the fixed volume case.

In order to implement variable volume, we time-step the heat and salt content instead of temperature and salinity (see A.5 and A.6). We also modify (A.7) the time stepping of momentum (like the ROMS model (Schetpet.....) for consistency). This was not done by Roullet and Madec (2000). It implies to multiply the components of the tracers and momentum equations by the vertical scale factors (thickness of the level) just before the time-stepping. Once the time-stepping is performed, the new variables must be divided by the scale factors.

Tracer equation:

$$\partial_t(hc) \quad = \quad h * [\text{advective and diffusive terms}]$$

where c is a tracer concentration (temperature, salinity) and h is the thickness of a layer.

Momentum equation:

$$\partial_t(hu_h) \quad = \quad h * [\text{advective, diffusive, ... terms}]$$

where u_h is the horizontal component of the velocity (u and v), and h is the thickness of a layer.

In the fixed volume case, the free surface condition is applied at $z = 0$ (linearization) and there is a non-zero vertical velocity at $z = 0$, which balances freshwater forcing and the variations of the free surface elevation. In the variable volume case, there is zero advective flux at the surface, and the concentration/dilution effect on salinity is zero. The corresponding subroutines of the code must be modified.

The list of the modified modules of NEMO is in A.3.

Chapter 3

Salt conservation with freshwater forcing

The first question that arises when you implement new capacities to a code is, how to validate the modifications? In our case a simple test is to check the conservation of salt content. In a nonlinear free surface code with varying volume, the salt must be perfectly conserved, and the total volume variations must follow exactly the E-P forcing. The conservation of salt is not exact with the linearized free surface (Roullet and Madec (2000), their Fig. 5).

Our configuration is a periodic zonal channel with flat bottom. It is designed after the baroclinic case of Ezer *et al.* (2002). The number of points is 66 by 66 and 31 vertical levels (ORCA2 grid). The horizontal resolution in both direction is equal to 8 km so that the physical domain dimensions are 512 x 512 km by 5000 m depth. It is an f-plane ($f_0 = 1.00274\,.10^{-4}s^{-1}$). The speed of linear gravity waves, which are the relevant dynamics in the model, is $c = \sqrt{gH} \simeq 220\,m.s^{-1}$, so that it takes 38 mn to travel the basin.

We want to test the salt conservation in the presence of large vertical diffusion, so we set the background K_z vertical diffusion coefficient to $10^{-2}m^2.s^{-1}$ for tracer and momentum (variables *avt0* and *avm0* in the namelist). We also activate the TKE scheme, because it is used in most applications of NEMO; this means that the time-stepping of the vertical diffusion is made with the implicit scheme. We also use the *ln_zdfevd* option to increase the mixing in the case of convection ($avevd = 1m^2.s^{-1}$ in the namelist). We verify in the output files that the vertical mixing coefficient is equal to the background over most of the domain, excepted in the three or four surface layers where it is 1 due to convection of saline water.

The horizontal diffusion/viscosity is zero and the advection scheme is the centered one.

The baroclinic time-step is the same for all the simulations presented in this chapter, and the barotropic time-step is equal to the baroclinic time-step.

The fluid is stratified in temperature (analytic profile) with a uniform salinity of 35.5 psu. The initial dynamical state is set by a barotropic zonal velocity of 0.1 $m.s^{-1}$ and the corresponding sea surface slope at time $t = 0$ ($\eta_y = -fu/g$).

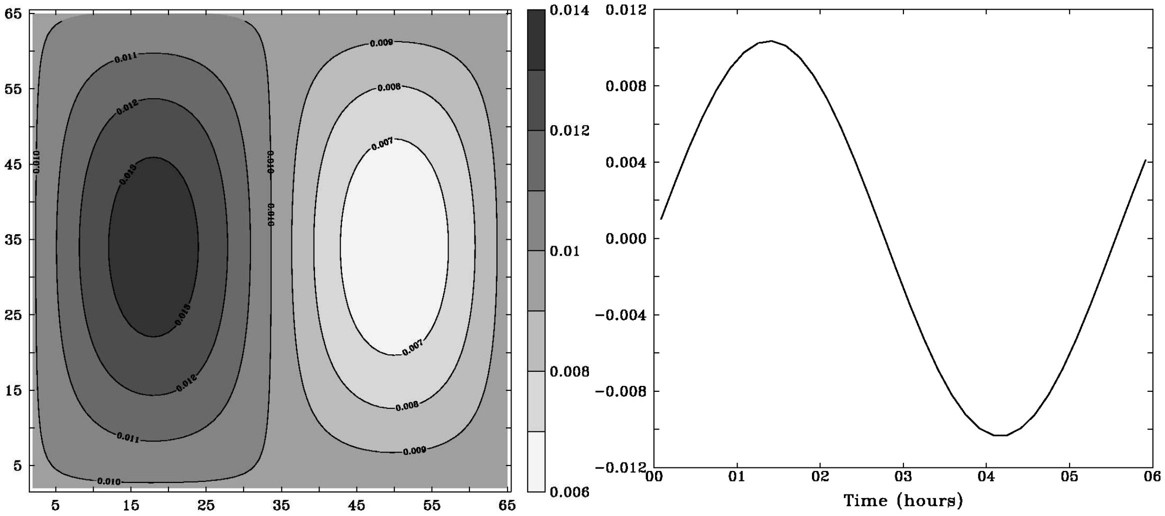
N_x	N_y	N_z	Δ_x (km)	Δ_y (km)	L_x (km)	L_y (km)
66	66	31	8	8	512	512

Table 3.1: Domain characteristics

The total volume flux *emp* and concentration/dilution effect *emps* are prescribed as periodic forcing with a period of 5 hours 33 mn and 20 s, with alternance of evaporation and precipitation.

$$\begin{aligned} emp &= \frac{10^9}{L_x L_Y} sin\left(\frac{2\pi}{N_x}x\right) sin\left(\frac{\pi}{N_y}y\right) sin\left(\frac{2\pi}{2000}t\right) + 10^{-2} sin\left(\frac{2\pi}{2000}t\right) \\ emps &= emp, \text{ contribution to salinity} \end{aligned}$$

Fig. 3.1(a) shows the forcing at one time step and fig. 3.1(b) shows the forcing at one point of the domain over one period. One can see that there is an alternance of precipitation and evaporation. The time to travel along the domain is 38 mn, so we take a short forcing period (less than 6 hours) and run the model for a duration of 6 hours in most of the cases (60 hours in some cases in order to have 10 periods). The amplitude of the forcing is very large (maximum water flux of 0.3 Sv) to test the model in extreme conditions.



(a)

(b)

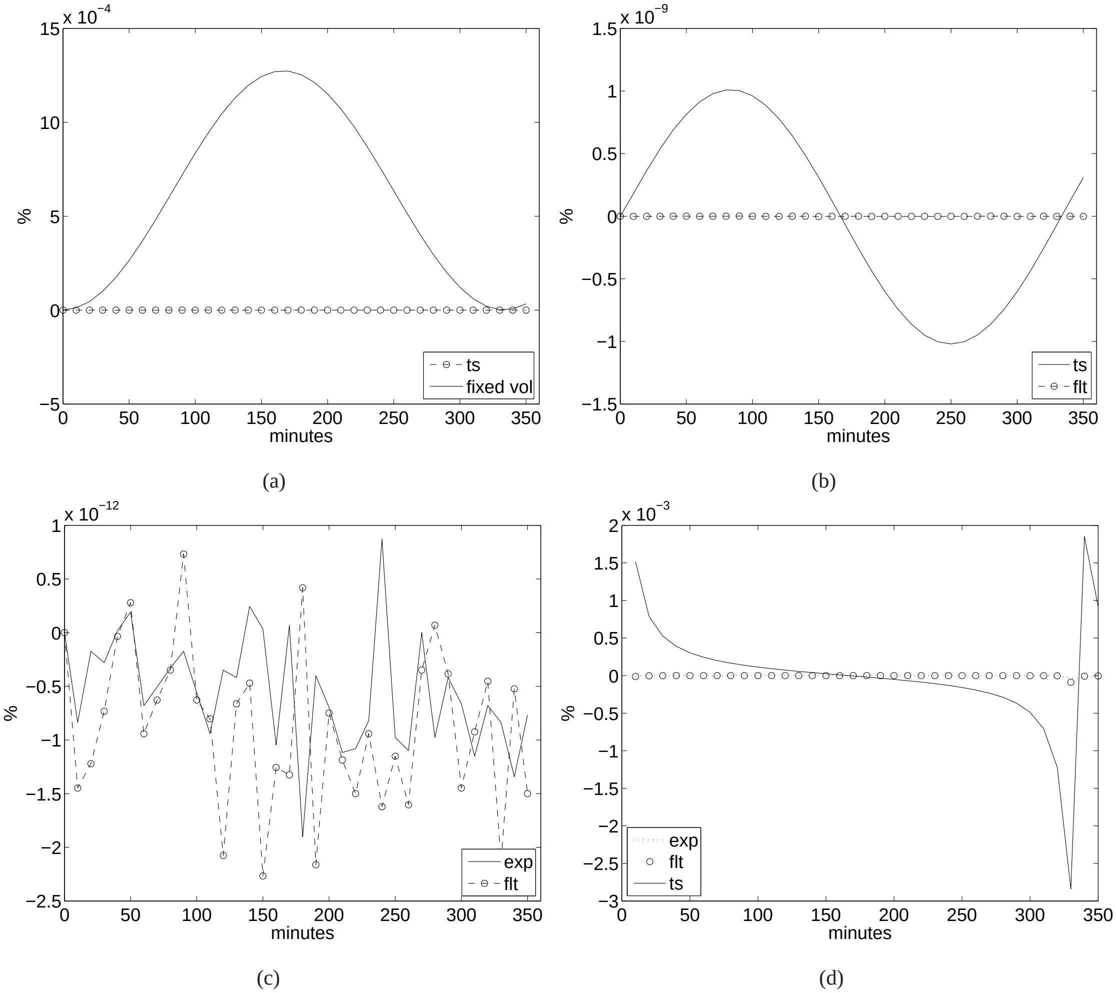
Figure 3.1: E-P forcing. **(a)** Over the whole domain at a specific time, **(b)** At one point (i=j=33) for one period.

In the following figures, we present the “salt quantity anomaly” ΔQ_s, which is the salt quantity minus its initial value. If S is the salinity and δV the volume of an ocean grid cell,

$$\Delta Q_s = Q_s(t=0) - Q_s \quad \text{with} \quad Q_s = \sum_{ocean\,cells} S\,\delta V$$

This quantity is calculated explicitly in the code in double precision.

In the fixed-volume version of NEMO, the maximum salt content variation is $1.3\,10^{-3}$ percent of the initial value (Fig. 3.2a). This is far from an exact conservation. With the variable volume there is a large improvement, which depends on the temporal scheme. The maximum variation of salt content is around 10^{-9} percent when we use the time-splitting scheme (Fig.3.2a, b and d) . With the explicit and filtered schemes, the conservation of salt content is again improved with a maximum of $2\,10^{-12}$% of the initial value (Fig.3.2c and d).



(a)

(b)

(c)

(d)

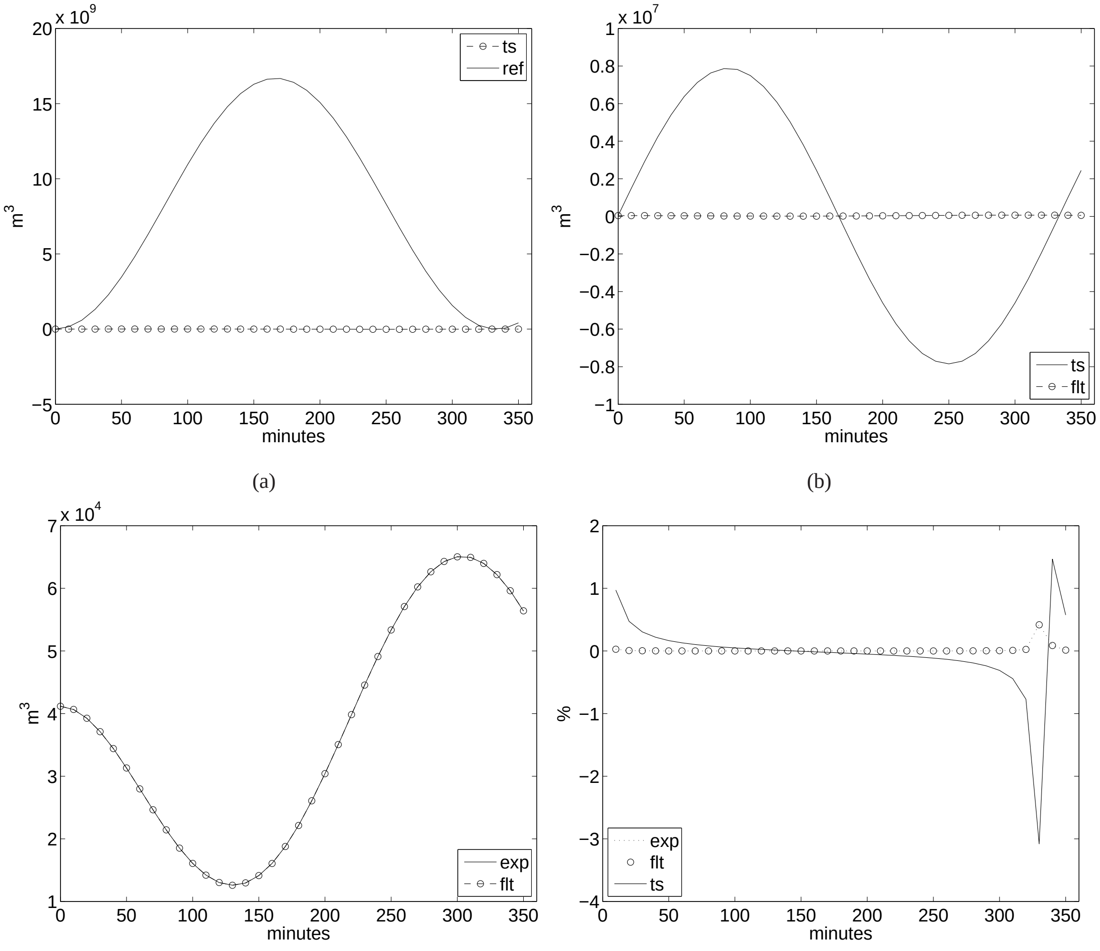
Figure 3.2: ΔQ_s (Variation of the salt content in percentage of the initial salt content) **(a)** time-splitting and fixed volume, **(b)** time-splitting and filtered, **(c)** explicit and filtered, **(d)** the 3 schemes in % of ΔQ_s for the fixed volume simulation

Another diagnostic to consider is the volume conservation. The volume variation ΔV must be equal to the amount of the freshwater forcing V_{E-P}.

$$\Delta V = V(t=0) - V \quad \text{with} \quad V = \sum_{ocean\,cells} \delta V$$

$$V_{E-P}(t) = \sum_t \left(\sum_x \sum_y (P + R - E) \right)$$

The same comments than with the salt conservation can be made for the volume conservation. The volume is well conserved in the case of the explicit and filtered schemes (Fig. 3.3c and d). With the time-splitting scheme (Fig. 3.3b and d) the conservation is worse but still good compared to the reference simulation with fixed volume (Fig. 3.3a).



(a)

(b)

(c)

(d)

Figure 3.3: ΔV minus V_{E-P} **(a)** time-splitting and fixed volume, **(b)** filtered and time-splitting, **(c)** explicit and filtered, **(d)** the 3 schemes in % of V_{E-P}

A model with variable volume must conserve the salt content, even with freshwater loss or supply, because those forcings do not carry salt. The volume must also be conserved: the variation of the mesh grid must balance exactly the freshwater forcing. Verifying these two basic points were essential in order to validate the variable volume version of NEMO. The results are very good (depending of the scheme used), so that we can say that the new version of the code with variable volume works fine.

Chapter 4

Gravity wave

We want to show the influence of variable volume on the dynamics. What is the behavior of a gravity wave in shallow coastal area ?

The model is initialized with a barotropic wave propagating eastward (the Coriolis factor f is set to zero):

$$\begin{cases} \eta(x,y) = \eta_0 \, cos\left(\frac{2\pi}{\lambda}x\right) \\ u(x,y) = \eta_0 \, \frac{g}{c_\phi} \, cos\left(\frac{2\pi}{\lambda}x\right) \\ v(x,y) = 0 \end{cases}$$

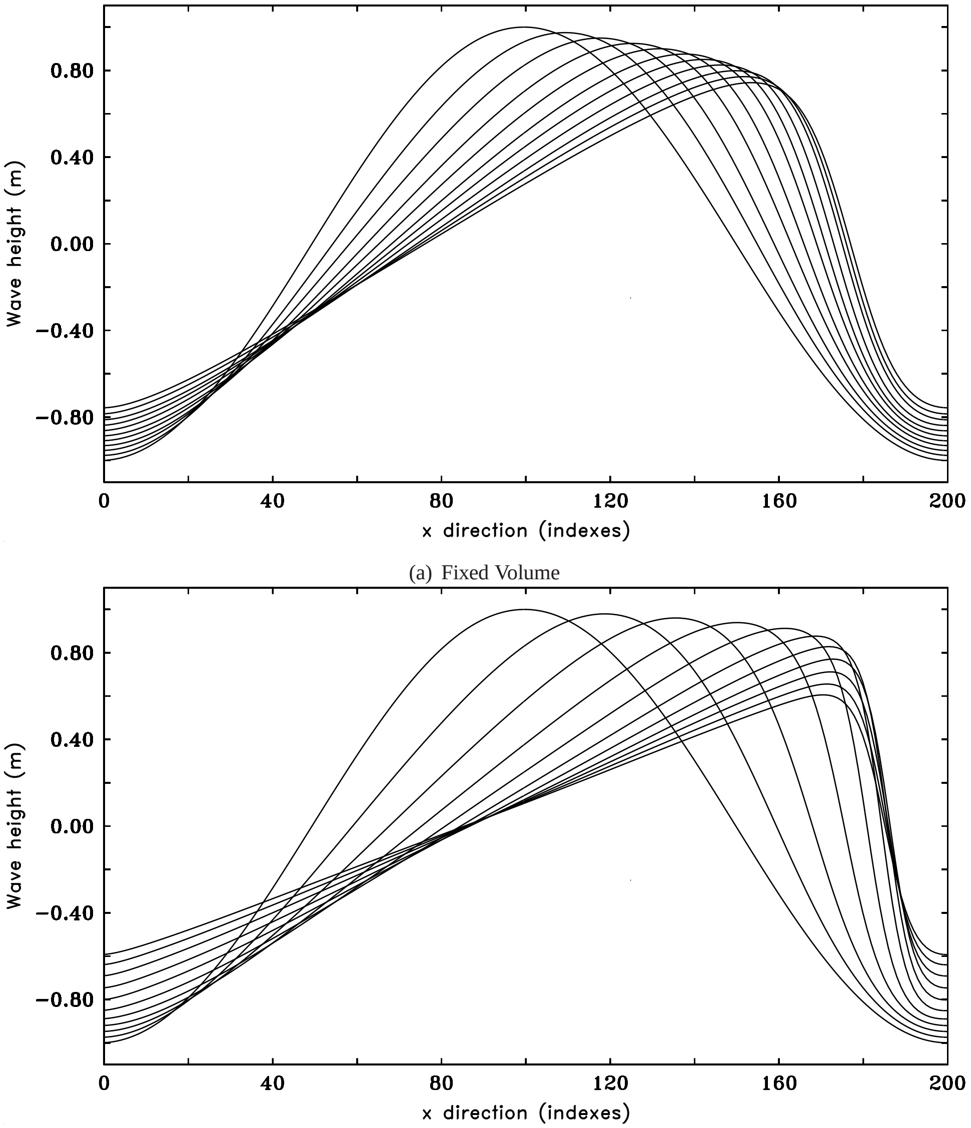
$c_\phi = \sqrt{gH}$ is set to 10 $m.s^{-1}$ so that $H \simeq 10.2\,m$ (depth of the fluid in rest). λ the wavelength is set to 2000 m.

Our configuration is a periodic zonal channel with the following number of points, N_x=202, N_y=12 and N_z=4. The horizontal resolution in both direction is equal to 10 m so that the domain dimensions are 2000 x 100 m.

The homogeneous fluid is set to a temperature of 12° C and a salinity of 35.5 p.s.u. The simulations are free, i.e. with no forcing. The horizontal eddy diffusivity for tracers and momentum is set to 50 $m.s^{-1}$. We run the model for 1000 s.

Fig. 4a shows the behavior of the free surface in the fixed volume case, and Fig. 4b in the variable volume case. With the variable volume we observe as expected the effect of the nonlinearity. The effect of the nonlinearity is to steepen the leading edge of a wave profile and flatten the trailing edge (the increase of wave speed with amplitude causes the leading part of the profile to steepen with time and the trailing part to flatten). This effect can be balanced by the effect of dispersion (the wave does not break because steepening is controlled by viscosity). We found no analytical solution to compare to this case, and it would interesting to compare with other models like ROMS (Shchepetkin and McWilliams 2005) or POLCOMS (Holt and James 2001).

Note: the fluid is homogeneous in salinity and temperature at its initial state. This state must not change. There is no forcing, so salinity and temperature must stay constant.



(a) Fixed Volume

(b) Variable Volume

Figure 4.1: Evolution in time of the sea surface height (1000 s between each curves) with the temporal explicit scheme. For visual effect, the curves are shifted so that x = 0 corresponds to a minimum of the amplitude of the wave. The gravity wave is propagating eastward. The slope forward the crest is steepening, while the amplitude of the wave is decreasing. The slope forward the crest is steeper in the variable volume case (b) than for the fixed volume case (a).

Bibliography

Bessières, L., 2003: Modélisation des ondes de gravité, dans le code de circulation générale opa-mercator. Master’s thesis, Université Paul Sabatier - INSA.

Blumberg, A. F. and G. L. Mellor, 1987: A description of a three-dimensional coastal ocean circulation model. In N. S. Heaps (Ed.), *Three Dimensional Coastal Ocean Models*, pp. 1–16. American Geophysical Union.

Ezer, T., H. Arango, and A. F. Shchepetkin, 2002: Developments in terrain-following ocean models: intercomparisons of numerical aspects. *Ocean Modelling*, 249–267.

Griffies, S. M., 2004: *Fundamentals of ocean climate models*. Princeton University Press.

Griffies, S. M., C. Böning, F. O. Bryan, E. Chassignet, R. Gerdes, H. Hasumi, A. Hirst, A-M. Tréguier, and D. Webb, 2000: Developments in ocean climate modelling. *Ocean Modelling*, 123–192.

Holt, J. T. and I. D. James, 2001: An s coordinate density evolving model of the northwest european continental shelf 1, model description and density structure. *J. Geophys. Res. 106*(C7), 14,015–14,034.

Madec, G., P. Delecluse, M. Imbard, and C. Lévy, 1998: Opa 8.1 ocean general circulation model reference manual. Rapport techn., Université P. et M. Curie. Notes de l’IPSL.

Pacanowski, R. C. and S. M. Griffies, 2000: *MOM 3.0 Manual*.

Roullet, G. and G. Madec, 2000, October: Salt conservation, free surface, and varying levels: a new formulation for ocean general circulation models. *J. Geophys. Res. 105*(C10), 23927–23942.

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

Talandier, C., C. Le Provost, and G. Roullet, 2003: Tidal waves in the opa gcm. Rapport techn., Laboratoire de Physique des Océans.

ANNEXES

Appendix A

Code

A.1 Algorithm of the variable grid

We want to distribute the variations of the thickness of the water column (due to variations of the sea surface elevation η) over all the layers of the model. The variations must affect the surface layers more than the bottom layers (and the last layer must be unchanged).

$$e_{3T}^{i,j,\mathbf{k},t} = e_{3T}^{i,j,\mathbf{k},0} + e_{3T}^{i,j,\mathbf{k},0} * \frac{\eta^{i,j,t}}{\nu^{i,j}} * \sum_{\mathbf{n}=k}^{jpk-1} e_{3T}^{i,j,\mathbf{n},0} \qquad \text{with} \qquad \nu^{i,j} = \sum_{\mathbf{k}=1}^{jpk-1} \left[e_{3T}^{i,j,\mathbf{k},0} \sum_{\mathbf{n}=k}^{jpk-1} e_{3T}^{i,j,\mathbf{n},0} \right]$$

We can verify that the sum of the new vertical scale factor $fse_{3t}^{i,j,\mathbf{k},t}$ is equal to the thickness of the whole water column at rest $H^{i,j}$ plus the sea surface elevation $\eta^{i,j,t}$:

$$\begin{aligned} \sum_{\mathbf{k}=1}^{jpk-1} e_{3T}^{i,j,\mathbf{k},t} &= \sum_{\mathbf{k}=1}^{jpk-1} e_{3T}^{i,j,\mathbf{k},0} + \frac{\eta^{i,j,t}}{\nu^{i,j}} * \sum_{\mathbf{k}=1}^{jpk-1} \left[e_{3T}^{i,j,\mathbf{k},0} \sum_{\mathbf{n}=k}^{jpk-1} e_{3T}^{i,j,\mathbf{n},0} \right] \\ &= H^{i,j} + \eta^{i,j,t} * \frac{\nu^{i,j}}{\nu^{i,j}} = H^{i,j} + \eta^{i,j,t} \end{aligned}$$

Once the scale factors for the T points are known, we can deduce the scale factor for the W points and the depth of the T and W points.

Scale factors:

$$e_{3t}(k) = e_{3t}^{0}(k) + e_{3t}^{0}(k) * \eta * cor(k)$$

$$\begin{aligned} e_{3w}(1) &= e_{3t}(1) \\ e_{3w}(k>1) &= \frac{1}{2}\left[e_{3t}(k-1) + e_{3t}(k)\right] \end{aligned}$$

Depth at T and W points

$$\begin{aligned}
gdep_t(1) &= \frac{e_{3w}(1)}{2} = \frac{e_{3t}(1)}{2} \\
gdep_t(k>1) &= gdep_t(k-1) + e_{3w}(k) \\
&= \sum_{n=1}^{k-1} e_{3t}(n) + \frac{e_{3t}(n)}{2} \\
\\
gdep_w(1) &= 0 \\
gdep_w(k>1) &= \sum_{n=2}^{k} e_{3t}(n-1)
\end{aligned}$$

This is done in the new module `domvvl`.

A.2 Module `domvvl.F90`

This module contains three subroutines. The first one, `dom_vvl_ini` is only called at the initialisation. The second one, `dom_vvl`, is called at each time step, and updates the new grid. The last one, `dom_vvl_ssh`, is called at each time step, and computes the sea surface elevation at the time after.

A.2.1 subroutine `dom_vvl_ini`

We calculate the coefficient applied to each layer and which do not depend of time.

$$\mu_T^{i,j,\mathbf{k}} = \frac{\sum_{\mathbf{n}=k}^{jpk-1} e_{3T}^{i,j,\mathbf{n},0}}{\sum_{\mathbf{k}=1}^{jpk-1}\left[e_{3T}^{i,j,\mathbf{k},0}\sum_{\mathbf{n}=k}^{jpk-1} e_{3T}^{i,j,\mathbf{n},0}\right]}$$

$$\mu_U^{i,j,\mathbf{k}} = \frac{\sum_{\mathbf{n}=k}^{jpk-1} e_{3U}^{i,j,\mathbf{n},0}}{\sum_{\mathbf{k}=1}^{jpk-1}\left[e_{3U}^{i,j,\mathbf{k},0}\sum_{\mathbf{n}=k}^{jpk-1} e_{3U}^{i,j,\mathbf{n},0}\right]}$$

$$\mu_V^{i,j,\mathbf{k}} = \frac{\sum_{\mathbf{n}=k}^{jpk-1} e_{3V}^{i,j,\mathbf{n},0}}{\sum_{\mathbf{k}=1}^{jpk-1}\left[e_{3V}^{i,j,\mathbf{k},0}\sum_{\mathbf{n}=k}^{jpk-1} e_{3V}^{i,j,\mathbf{n},0}\right]}$$

$$\mu_F^{i,j,\mathbf{k}} = \frac{\sum_{\mathbf{n}=k}^{jpk-1} e_{3F}^{i,j,\mathbf{n},0}}{\sum_{\mathbf{k}=1}^{jpk-1}\left[e_{3F}^{i,j,\mathbf{k},0}\sum_{\mathbf{n}=k}^{jpk-1} e_{3F}^{i,j,\mathbf{n},0}\right]}$$

A.2.2 subroutine `dom_vvl`

At each time step, we calculate the sea surface elevation at each grid point, the depth and total depth.

$$\eta_U^{i,j} = \frac{1}{2}\frac{mask_U^{i,j,1}}{e_{1U}^{i,j}\,e_{2U}^{i,j}}\left(e_{1T}^{i,j}\,e_{2T}^{i,j}\,\eta^{i,j} + e_{1T}^{i+1,j}\,e_{2T}^{i+1,j}\,\eta^{i+1,j}\right)$$

$$\eta_V^{i,j} = \frac{1}{2}\frac{mask_V^{i,j,1}}{e_{1V}^{i,j}\,e_{2V}^{i,j}}\left(e_{1T}^{i,j}\,e_{2T}^{i,j}\,\eta^{i,j} + e_{1T}^{i,j+1}\,e_{2T}^{i,j+1}\,\eta^{i,j+1}\right)$$

$$\eta_F^{i,j} = \frac{1}{4} mask_F^{i,j,1}\left(\eta^{i,j} + \eta^{i+1,j} + \eta^{i,j+1} + \eta^{i+1,j+1}\right)$$

Scale factors at T levels

$$\begin{aligned}
fse_{3T}^{i,j,k} &= e_{3T}^{i,j,k,0}(1+\eta_T^{i,j}\,\mu_T^{i,j,k}) \\
fse_{3U}^{i,j,k} &= e_{3U}^{i,j,k,0}(1+\eta_U^{i,j}\,\mu_U^{i,j,k}) \\
fse_{3V}^{i,j,k} &= e_{3V}^{i,j,k,0}(1+\eta_V^{i,j}\,\mu_V^{i,j,k}) \\
fse_{3F}^{i,j,k} &= e_{3F}^{i,j,k,0}(1+\eta_F^{i,j}\,\mu_F^{i,j,k})
\end{aligned}$$

Scale factors at W levels

$$\begin{array}{rl}
\text{surface} & \text{for} \quad k \geq 2 \\
fse_{3W}^{i,j,1} = fse3t^{i,j,1} & fse_{3W}^{i,j,k} = 0.5\,(fse_{3T}^{i,j,k-1} + fse_{3T}^{i,j,k}) \\
fse_{3UW}^{i,j,1} = fse3u^{i,j,1} & fse_{3UW}^{i,j,k} = 0.5\,(fse_{3U}^{i,j,k-1} + fse_{3U}^{i,j,k}) \\
fse_{3VW}^{i,j,1} = fse3v^{i,j,1} & fse_{3VW}^{i,j,k} = 0.5\,(fse_{3V}^{i,j,k-1} + fse_{3V}^{i,j,k})
\end{array}$$

T and W points depth

$$\begin{array}{rl}
\text{surface} & \text{for} \quad k \geq 2 \\
fsdept^{i,j,1} = 0.5 * fse_{3w}^{i,j,1} & fsdep_T^{i,j,k} = fsdep_T^{i,j,k-1} + fse_{3W}^{i,j,k} \\
fsdepw^{i,j,1} = 0 & fsdep_W^{i,j,k} = fsdep_W^{i,j,k-1} + fse_{3T}^{i,j,k-1} \\
fsde3w^{i,j,1} = fsdep_T^{i,j,1} - \eta_T^{i,j} & fsde_{3W}^{i,j,k} = fsdep_T^{i,j,k} - \eta_T^{i,j}
\end{array}$$

Ocean depth at U- and V-points

$$\begin{aligned}
hu^{i,j} &= \sum_k fse_{3U}^{i,j,k}\,mask_U^{i,j,k} \\
hv^{i,j} &= \sum_k fse_{3V}^{i,j,k}\,mask_V^{i,j,k}
\end{aligned}$$

Inverse of the local depth

$$\begin{aligned}
hur^{i,j} &= \frac{mask_U^{i,j,1}}{fse_{3U}^{i,j,1} + \sum_{k\geq 2} fse_{3U}^{i,j,k}\,mask_U^{i,j,k}} \\
hvr^{i,j} &= \frac{mask_V^{i,j,1}}{fse_{3V}^{i,j,1} + \sum_{k\geq 2} fse_{3V}^{i,j,k}\,mask_V^{i,j,k}}
\end{aligned}$$

A.2.3 subroutine `dom_vvl_ssh`

The sea surface elevation at the time after *ssha* is needed at several moments:

- in `tra_nxt` for the flux form caculation,
- in `dyn_nxt` for the flux form caculation,
- in `dynspg_exp`, `dynspg_flt` and `dynspg_ts` for the swap of the ssh arrays,
- in `wzv` for the calculation of the vertical velocity.

The dynamics equation of the sea surface evolution can be written:

$$\frac{\partial \eta}{\partial t} = -\nabla_h U + q_w$$

For the explicit and filtered schemes, the expression of *ssha* is discretized like that:

$$\begin{aligned}
& ssha && = sshb - 2\,rdt \left(zhdiv + \frac{emp}{rauw} \right) \\
\text{where} \quad & zhdiv && = \frac{e_{2U}^{i,j}\, zun^{i,j} - e_{2U}^{i-1,j}\, zun^{i-1,j} + e_{2V}^{i,j}\, zvn^{i,j} - e_{2V}^{i,j-1}\, zvn^{i,j-1}}{e_{1T}^{i,j}\, e_{2T}^{i,j}} \\
\text{and} \quad & zun^{i,j} && = \sum_k fse_{3U}^{i,j,k}\, un^{i,j,k}, \quad zvn^{i,j} = \sum_k fse_{3V}^{i,j,k}\, vn^{i,j,k}
\end{aligned}$$

For the time-splitting scheme, the expression of *ssha* is discretized like that[1]:

$$\begin{aligned}
& ssha && = sshb\_b - 2\,rdt \left(zhdiv + \frac{emp}{rauw} \right) \\
\text{where} \quad & zhdiv && = \frac{e_{2U}^{i,j}\, zun^{i,j} - e_{2U}^{i-1,j}\, zun^{i-1,j} + e_{2V}^{i,j}\, zuv^{i,j} - e_{2V}^{i,j-1}\, zuv^{i,j-1}}{e_{1T}^{i,j}\, e_{2T}^{i,j}} \\
\text{and} \quad & zun^{i,j} && = un\_b^{i,j,k}, \quad zvn^{i,j} = vn\_b^{i,j,k}
\end{aligned}$$

The variables *sshb_b*, *un_b* and *uv_b* are calculated in the `dynspg_ts` subroutine.
In the `wzv`, *ssha* must be calculated again.

[1] this is necessary to satisfy the conservation of tracers quantity, but the conservation of tracers is not satisfied. Using the same calculation of *ssha* as in the explicit and filtered schemes for the flux form calculation allows the conservation of tracers, but not the conservation of the tracers quantity. The time-splitting algorithm must be modified to garantee both *exact* conservation and constancy preservation properties for tracers (Shchepetkin and McWilliams 2005).

A.3 Modified routines

A.3.1 CPP and namelist variables

A **new key** is introduced: *key_vvl* ; when this key is active the layer thicknesses depend on the free surface (and thus depend on **time**).

Free surface: we need a free surface, *key_dynspg_flt* (filtered free surface, default) or *key_dynspg_exp* (explicit) or *key_dynspg_ts* (time splitting).

Vertical coordinates: *ln_zco*, *ln_sco* or *ln_zps*

- *key_vvl* is compatible with *ln_zco* (z-coordinates, but provided *key_zco* is not active).
- *key_vvl* is compatible with *ln_sco* (s coordinates)
- *key_vvl* is incompatible with *ln_zps* (partial steps) because it is complicated and not coded yet
- *key_vvl* is incompatible with *key_zco* (because in that case scale factors are 1 dimensional).
- *key_vvl* is incompatible with *key_dynspg_rl* (rigid lid) because we need a free surface

Hydrostatic pressure gradient option: at this time, *key_vvl* is compatible only with the standard jacobian formulation of the hydrostatic pressure gradient option (*ln_hpg_sco* = .true.).

A.3.2 Modules

New module: `domvvl.F90` in directory DOM, contains :

- `dom_vvl_init` stores previously calculated *gdept*, *e3t*... etc arrays into *gdepti*, *e3ti*, ... etc calculates *mut*, *muu*, *muv*, *muf* (correction for each grid-points)
- `dom_vvl`
  - calculates new *sshn* at other grid points (*sshnu*, *sshnv* ...)
  - calculates *e3t*, *e3u*, *e3v*, using new ssh (*sshn*)
  - calculates *e3w*
  - uses the e3 to calculate *gdept* (depth of t points relative to free surface), *gdepw* (depth of w points relative to free surface), *gdep3w* = *gdept* - *eta*. (depth of T points relative to a fixed (geopotential) reference level for pressure gradient correction).
  - updates *hu*, *hv*, *hur*, *hvr* used in time splitting and in `solmat` routine
- `dom_vvl_ssh`
  - calculates *ssha*

List of modified routines:
`domain.F90`, `dom_oce.F90`, `domstp.F90`, `domvvl.F90`, `domwri.F90`, `domzgr.F90`, `domzgr_substitute.h90`, `dynhpg.F90`, `dynnxt.F90`, `dynspg_exp.F90`, `dynspg_flt.F90`, `dynspg_ts.F90`, `dynspg_oce.F90`, `istate.F90`, `oce.F90`, `ocesbc.F90`, `par_EEL_R5.h90`, `step.F90`, `traadv_cen2.F90`, `traadv_muscl2.F90`, `traadv_muscl.F90`, `traadv_tvd.F90`, `tranxt.F90`, `trasbc.F90`, `trazdf.F90`, `trazdf_imp.F90`, `wzvmod.F90`

OPA_SRC :

- `istate.F90`
  - subroutine `istate_init` initialization of *sshn*, *sshb*, *sshb_b*, *sshbb*, *un_b*, *vn_b*and call of *dom_vvl* at the end
  - subroutine `istate_eel` modifications to match Ezer et al
- `par_EEL_R5.h90` modifications to match Ezer et al
- `oce.F90` declaration of *sshu*, *sshv*, *sshbb*, *ssha*
- `step.F90` subroutine `stp` call of *dom_vvl* at the end of the dynamics part (after the call of *dyn_nxt*

DOM :

- `domain.F90` subroutine `dom_ini` call of `dom_vvl_ini` after call of `dom_msk`
- `dom_oce.F90` declaration of logical *key lk_vvl* grid variables *gdepti*, *gdepwi*, *gdep3w*, *e3ti*, *e3ui*, *e3vi*, *e3fi*, *e3uwi*, *e3vwi*, *e3w*
- `domstp.F90` subroutine `dom_stp` stop execution in the case where accelerating the convergence is activated
- `domzgr.F90` subroutine `zgr_bat` modifications to match Ezer et al (not zero at the equator)

`domzgr_substitute.h90` definition of scale factors

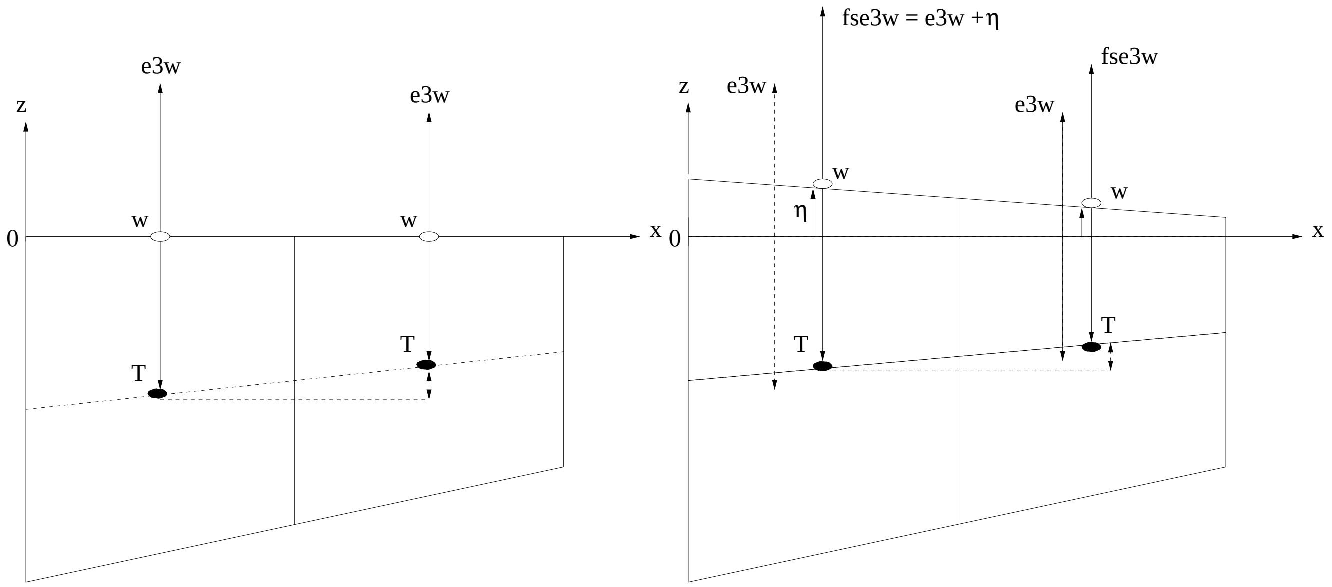
DYN :

- `dynhpg.F90` subroutine `hpg_sco` problem working with density anomaly
- `dynnxt.F90` subroutine `dyn_nxt` time stepping in flux form: multiply u and v by scale factors before time stepping, and divide after. If *key_dynspg_flt* is true, the time stepping is done in `dyn_spg_flt` subroutine.
- `dynspg_exp.F90` subroutine `dyn_spg_exp` does not calculate surface pressure gradient (already done in dynhpg) ; the ssh after used for the ssh swap has already been calculated in `dom_vvl_ssh`
- `dynspg_flt.F90` subroutine `dyn_spg_flt` does not calculate surface pressure gradient (already done in dynhpg). The time stepping is done in flux form: multiply u and v by scale factors before time stepping, and divide after. Add a call to subroutine `sol_mat` because it depends on scale factors. The ssh after used for the ssh swap has already been calculated in `dom_vvl_ssh`
- `dynspg_ts.F90` subroutine `dyn_spg_ts` subtract the surface pressure gradient (calculated in dynhpg) before the barotropic part. The calculation of the surface pressure gradient in the barotropic part is calculated using the product _rhd*sshn_ (*rho* total = *rhd*+1). We take *rhd* at the top level. *hu* and *hv* are calculated at each barotropic time steps (they are renamed *hu_e* and *hv_e*). The ssh after used for the ssh swap has already been calculated in `dom_vvl_ssh`.
- `dynspg_flt.F90` the declaration of *sshn_b*, *sshb_b*, *un_b* and *vn_b* is needed even if *key_dynspg_ts* is not activated, in order to compile the code.
- `wzvmod.F90` subroutine `wzv` new definition of vertical velocity

TRA :

- traadv_cen2.F90, traadv_muscl2.F90, traadv_muscl.F90, traadv_tvd.F90 zwx (for T) and zwy (for S) set to zero (zero advective flux at surface). The heat flux linked with the temperature of evaporation/precipitation is taken care of in trasbc.F90.
- tranxt.F90 subroutine tra_nxt time stepping in flux form (for explicit vertical diffusion): multiply T and S by scale factors before time stepping, and divide after. The ssh after used has been calculated in dom_vvl_ssh.
- trasbc.F90 subroutine tra_sbc add emp for T, zsa set to zero for S (the concentration/dilution effect on salinity is zero)
- trazdf.F90 subroutine tra_zdf call of dom_vvl_ssh
- trazdf_imp.F90 subroutine tra_zdf_imp, time stepping in flux form for implicit vertical diffusion. *fse3tb*, *fse3ta* needed in matrix calculation

A.4 Hydrostatic pressure gradient: example with a first variable level



(a) Fixed volume : η = 0

(b) Variable volume

Vertical section of 2 grid cells in *s* coordinates. (a) Fixed volume. (b) Variable volume.

The hydrostatic pressure gradient between 2 cells is equal to the difference of the water column weight between T points of 2 cells, divided by the distance between the T points ($e_1^{i,j,k}$ or $e_2^{i,j,k}$).

In order to calculate the water column height at T points, vertical scale factors joining each T points are summed. In *s* coordinates, that height can vary from one cell to another and one must give a correction to obtain the gradient at the right depth.

In OPA with **fixed volume**, the vertical scale factor for the first level ($e_{3w}^{i,j,1}$) is equal to twice the real depth of the first T point, and the hydrostatic pressure gradient along *s* surfaces is written:

$$zhpi^{i,j,1} = -g * \frac{1}{e_{1u}^{i,j}} * \left(\frac{e_{3w}^{i+1,j,1}}{2} * \rho^{i+1,j,1} - \frac{e_{3w}^{i,j,1}}{2} * \rho^{i,j,1} \right)$$

The correction added to the pressure gradient is written:

$$zuap = g * \frac{1}{e_{1u}^{i,j}} * \left(\rho^{i+1,j,1} + \rho^{i,j,1} \right) * \frac{1}{2} * \left(\frac{e_{3w}^{i+1,j,1}}{2} - \frac{e_{3w}^{i,j,1}}{2} \right)$$

In OPA with **variable volume**, the SSH is added to the first vertical scale ($e_{3w}^{i,j,1} + \eta^{i,j}$), and the position of the first T point on the vertical (taken from the sea surface) is re-defined as the half of $e_{3w}^{i,j,1} + \eta^{i,j}$. This expression is substituted to the fixed volume case in the hydrostatic pressure gradient computation, which is then written:

$$zhpi^{i,j,1} = -g * \frac{1}{e_{1u}^{i,j}} * \left(\frac{e_{3w}^{i+1,j,1} + \eta^{i+1,j}}{2} * \rho^{i+1,j,1} - \frac{e_{3w}^{i,j,1} + \eta^{i,j}}{2} * \rho^{i,j,1} \right)$$

The calculation of the expression of the correction of the pressure gradient is done in report of the surface of the fluid at rest ($z = 0$). In this case, the T point depth is equal to the true height of the water above the T point minus SSH: $\frac{e_{3w}^{i,j,1}+\eta^{i,j}}{2} - \eta^{i,j} = \frac{e_{3w}^{i,j,1}-\eta^{i,j}}{2}$. Therefore the expression of the correction of the pressure gradient:

$$zuap = g * \frac{1}{e_{1u}^{i,j}} * \left(\rho^{i+1,j,1} + \rho^{i,j,1}\right) * \frac{1}{2} * \left(\frac{e_{3w}^{i+1,j,1} - \eta^{i+1,j}}{2} - \frac{e_{3w}^{i,j,1} - \eta^{i,j}}{2}\right)$$

A.5 Flux form time-stepping of tracer

For a tracer T at a given time-step t, the equation of evolution is written:

$$\frac{fse_{3T}^{t+1}(\mathrm{i,j,k})\,T^{t+1}(\mathrm{i,j,k}) - fse_{3T}^{t-1}(\mathrm{i,j,k})\,T^{t-1}(\mathrm{i,j,k})}{2\,\Delta t} \quad = \quad fse_{3T}^{t}(\mathrm{i,j,k})\,RHS(\mathrm{i,j,k})$$

where *RHS* contains the advection and diffusion terms.

The 3 following steps are performed in the `tra_nxt` subroutine (steps 1 and 2 are done only for the explicit diffusion case; see A.6 for the implicit diffusion case):

1) Thickness weighting

$$\begin{aligned} RHS(\mathrm{i,j,k}) &= RHS(\mathrm{i,j,k}) * fse_{3T}^{t}(\mathrm{i,j,k}) \\ T^{t}(\mathrm{i,j,k}) &= T^{t}(\mathrm{i,j,k}) * fse_{3T}^{t}(\mathrm{i,j,k}) \\ T^{t-1}(\mathrm{i,j,k}) &= T^{t-1}(\mathrm{i,j,k}) * fse_{3T}^{t-1}(\mathrm{i,j,k}) \end{aligned}$$

2) Time stepping (Leap-frog scheme)

$$T^{t+1}(\mathrm{i,j,k}) \quad = \quad T^{t-1}(\mathrm{i,j,k}) + 2\,rdt\,RHS(\mathrm{i,j,k})$$

3) Time filter and swap of arrays

$$\begin{aligned} T^{t-1}(\mathrm{i,j,k}) &= \frac{\alpha * [T^{t-1}(\mathrm{i,j,k}) + T^{t+1}(\mathrm{i,j,k})] + \alpha_1 * T^{t}(\mathrm{i,j,k})}{\alpha * [fse_{3T}^{t-1}(\mathrm{i,j,k}) + fse_{3T}^{t+1}(\mathrm{i,j,k})] + \alpha_1 * fse_{3T}^{t}(\mathrm{i,j,k})} \\ T^{t}(\mathrm{i,j,k}) &= \frac{T^{t+1}(\mathrm{i,j,k})}{fse_{3T}^{t+1}(\mathrm{i,j,k})} \end{aligned}$$

A.6 Flux form time-stepping of tracer: implicit case

The implicit treatment of the vertical diffusion terms is used to accomodate the small vertical spacing required to resolve the important top and bottom boundary layers without drastically reducing the time step as would be the case with the more usual explicit schemes (Blumberg and Mellor 1987). The use of an implicit scheme results in a tri-diagonal matrix which is solved by a Gaussian elimination method.

The vertical diffusive operator for the tracer takes the following semi-discrete space form:

$$D_T^{\nu T} \equiv \frac{1}{e_{3T}} \delta_k \left[\frac{A_w^{\nu T}}{e_{3w}} \delta_{k+1/2}[T] \right]$$

where $A_w^{\nu T}$ is the diffusivity coefficient.

First case: fixed volume

The tracer equation is written:

$$\begin{aligned}
\frac{T^{t+1}(k) - T^{t-1}(k)}{2\Delta t} &= RHS(k) + D_T^{\nu T}(k) \\
T^{t+1}(k) - T^{t-1}(k) &= 2\Delta t\, RHS(k) + 2\Delta t\, D_T^{\nu T}(k) \\
&= 2\Delta t\, RHS(k) + \\
&\quad \frac{2\Delta t}{e_{3T}(\mathrm{k})} \left[\frac{A_w^{\nu T}(\mathrm{k+1})}{e_{3w}(\mathrm{k+1})} \left[T^{t+1}(\mathrm{k+1}) - T^{t+1}(\mathrm{k}) \right] - \frac{A_w^{\nu T}(\mathrm{k})}{e_{3w}(\mathrm{k})} \left[T^{t+1}(\mathrm{k}) - T^{t+1}(\mathrm{k\text{-}1}) \right] \right] \\
&= 2\Delta t\, RHS(k) + \frac{2\Delta t\, A_w^{\nu T}(\mathrm{k+1})}{e_{3T}(\mathrm{k})\, e_{3w}(\mathrm{k+1})} T^{t+1}(\mathrm{k+1}) - \\
&\quad \frac{2\Delta t}{e_{3T}(\mathrm{k})} \left(\frac{A_w^{\nu T}(\mathrm{k+1})}{e_{3w}(\mathrm{k+1})} + \frac{A_w^{\nu T}(\mathrm{k})}{e_{3w}(\mathrm{k})} \right) T^{t+1}(\mathrm{k}) + \frac{2\Delta t\, A_w^{\nu T}(\mathrm{k})}{e_{3T}(\mathrm{k})\, e_{3w}(\mathrm{k})} T^{t+1}(\mathrm{k\text{-}1}) \\
T^{t-1}(k) + 2\Delta t\, RHS(k) &= T^{t+1}(\mathrm{k+1}) \left[\frac{-2\Delta t\, A_w^{\nu T}(\mathrm{k+1})}{e_{3T}(\mathrm{k})\, e_{3w}(\mathrm{k+1})} \right] + \\
&\quad T^{t+1}(\mathrm{k}) \left[1 + \frac{2\Delta t}{e_{3T}(\mathrm{k})} \left(\frac{A_w^{\nu T}(\mathrm{k+1})}{e_{3w}(\mathrm{k+1})} + \frac{A_w^{\nu T}(\mathrm{k})}{e_{3w}(\mathrm{k})} \right) \right] + T^{t+1}(\mathrm{k\text{-}1}) \left[-\frac{2\Delta t\, A_w^{\nu T}(\mathrm{k})}{e_{3T}(\mathrm{k})\, e_{3w}(\mathrm{k})} \right]
\end{aligned}$$

Matrix construction:

$$\begin{aligned}
zwi(k) &= \begin{cases} \dfrac{-2\Delta t\, A_w^{\nu T}(\mathrm{k})}{e_{3T}(\mathrm{k})\, e_{3w}(\mathrm{k})} & \text{for } k \geq 2 \\[2ex] 0 & \text{for } k = 1 \text{ (surface boundary condition)} \end{cases} \\
zws(k) &= -\frac{2\Delta t\, A_w^{\nu T}(\mathrm{k+1})}{e_{3T}(\mathrm{k})\, e_{3w}(\mathrm{k+1})} \\
zwd(k) &= 1 - zwi(k) - zws(k)
\end{aligned}$$

Second member construction:

$$zwy(k) \quad = \quad T^{t-1}(k) + 2\,\Delta t\, RHS(k)$$

System to be solved:

$$zwi(\mathrm{k})\, T^{t+1}(\text{k-1}) + zwd(\mathrm{k})\, T^{t+1}(\mathrm{k}) + zws(\mathrm{k})\, T^{t+1}(\text{k+1}) \quad = \quad zwy(\mathrm{k})$$

$$\begin{pmatrix} zwd(1) & zws(1) & 0 & 0 & 0 & \cdots & 0 \\ zwi(2) & zwd(2) & zws(2) & 0 & 0 & \cdots & 0 \\ 0 & zwi(3) & zwd(3) & zws(3) & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & zwi(k) & zwd(k) \end{pmatrix} \begin{pmatrix} T^{t+1}(1) \\ T^{t+1}(2) \\ T^{t+1}(3) \\ \vdots \\ T^{t+1}(k) \end{pmatrix} = \begin{pmatrix} zwy(1) \\ zwy(2) \\ zwy(3) \\ \vdots \\ zwy(k) \end{pmatrix}$$

$$m.x \,=\, y$$

where m is a tri-diagonal matrix (jpk * jpk). m is decomposed in the product of an upper and lower triangular matrix. The 3 diagonal terms are in arrays *zwd*, *zws*, *zwi*. The second member is in array *zwy*.

Second case: variable volume

The tracer equation is now written:

$$
\begin{aligned}
\frac{fse_{3T}^{t+1}(\text{k})\,T^{t+1}(\text{k}) - fse_{3T}^{t-1}(\text{k})\,T^{t-1}(\text{k})}{2\,\Delta t} &= fse_{3T}^{t}(\text{k})\,RHS(\text{k}) + fse_{3T}^{t}(k)\,D_T^{\nu T}(k) \\
fse_{3T}^{t+1}(\text{k})\,T^{t+1}(\text{k}) - fse_{3T}^{t-1}(\text{k})\,T^{t-1}(\text{k}) &= 2\,\Delta t\; fse_{3T}^{t}(\text{k})\,RHS(\text{k}) + 2\,\Delta t\; fse_{3T}^{t}(k)\,D_T^{\nu T}(k) \\
&= 2\,\Delta t\; fse_{3T}^{t}(\text{k})\,RHS(\text{k}) + \\
&\quad 2\,\Delta t \left[\frac{A_w^{\nu T}(\text{k+1})}{e_{3w}(\text{k+1})} \left[T^{t+1}(\text{k+1}) - T^{t+1}(\text{k}) \right] \right. \\
&\qquad \left. - \frac{A_w^{\nu T}(\text{k})}{e_{3w}(\text{k})} \left[T^{t+1}(\text{k}) - T^{t+1}(\text{k-1}) \right] \right] \\
&= 2\,\Delta t\; fse_{3T}^{t}(\text{k})\,RHS(\text{k}) + \frac{2\,\Delta t\, A_w^{\nu T}(\text{k+1})}{e_{3w}(\text{k+1})}\, T^{t+1}(\text{k+1}) - \\
&\quad 2\,\Delta t \left(\frac{A_w^{\nu T}(\text{k+1})}{e_{3w}(\text{k+1})} + \frac{A_w^{\nu T}(\text{k})}{e_{3w}(\text{k})} \right) T^{t+1}(\text{k}) + \frac{2\,\Delta t\, A_w^{\nu T}(\text{k})}{e_{3w}(\text{k})}\, T^{t+1}(\text{k-1}) \\
fse_{3T}^{t-1}(\text{k})\,T^{t-1}(\text{k}) + 2\,\Delta t\; fse_{3T}^{t}(\text{k})\,RHS(\text{k}) &= T^{t+1}(\text{k+1}) \left[\frac{-2\,\Delta t\, A_w^{\nu T}(\text{k+1})}{e_{3w}(\text{k+1})} \right] + \\
&\quad T^{t+1}(\text{k}) \left[fse_{3T}^{t+1}(\text{k}) + 2\,\Delta t \left(\frac{A_w^{\nu T}(\text{k+1})}{e_{3w}(\text{k+1})} + \frac{A_w^{\nu T}(\text{k})}{e_{3w}(\text{k})} \right) \right] + \\
&\quad T^{t+1}(\text{k-1}) \left[- \frac{2\,\Delta t\, A_w^{\nu T}(\text{k})}{e_{3w}(\text{k})} \right]
\end{aligned}
$$

Matrix construction:

$$
\begin{aligned}
zwi(k) &= \begin{cases} \dfrac{-\,2\,\Delta t\, A_w^{\nu T}(\text{k})}{e_{3w}(\text{k})} & \text{for } k \geq 2 \\[2ex] 0 & \text{for } k = 1 \text{ (surface boundary condition)} \end{cases} \\
zws(k) &= -\,\frac{2\,\Delta t\, A_w^{\nu T}(\text{k+1})}{e_{3w}(\text{k+1})} \\
zwd(k) &= fse_{3T}^{t-1}(\text{k}) - zwi(k) - zws(k)
\end{aligned}
$$

Second member construction:

$$
zwy(k) = fse_{3T}^{t-1}(\text{k})\,T^{t-1}(k) + 2\,\Delta t\; fse_{3T}^{t}(\text{k})\,RHS(k)
$$

The vertical diffusion is performed in the `tra_zdf_imp` or `tra_zdf_exp` routines, which are the last routines called before the `tra_nxt` routine. In the implicit treatment, the time stepping is already done, and call to `tra_nxt` is not necessary.

Dynamics case

The method is the same as for the tracers, but a normalization is made to obtain the general momentum trend. Whatever the treatment (explicit or implicit), the time stepping is performed further (in `dyn_nxt` routine, or in `dyn_spg_flt` routine if `key_dyn_spg_flt` is defined).

A.7 Flux form time-stepping of momentum

For the horizontal component of velocity (*U* or *V*) at a given time-step *t*, the equation of evolution is written:

$$\frac{fse_{3U}^{t+1}(\text{i,j,k})\,U^{t+1}(\text{i,j,k}) - fse_{3U}^{t-1}(\text{i,j,k})\,U^{t-1}(\text{i,j,k})}{2\,\Delta t} = fse_{3U}^{t}(\text{i,j,k})\,RHS(\text{i,j,k})$$

where *RHS* containts the advection and diffusion terms.

The 3 following steps are performed in the `dyn_nxt` subroutine for the explicit and time-splitting schemes, and in the `dynspg_flt` subroutine for the filtered scheme (for the filtered scheme, a part of step 3 - division by scale factors - is done in `dynspg_flt`, and a second part - swap of arrays - in `dyn_nxt`):

1) Thickness weighting

$$\begin{aligned}
RHS(\text{i,j,k}) &= RHS(\text{i,j,k}) * fse_{3U}^{t}(\text{i,j,k}) \\
U^{t}(\text{i,j,k}) &= U^{t}(\text{i,j,k}) * fse_{3U}^{t}(\text{i,j,k}) \\
U^{t-1}(\text{i,j,k}) &= U^{t-1}(\text{i,j,k}) * fse_{3U}^{t-1}(\text{i,j,k})
\end{aligned}$$

2) Time stepping (Leap-frog scheme)

$$U^{t+1}(\text{i,j,k}) = U^{t-1}(\text{i,j,k}) + 2\,rdt\,RHS(\text{i,j,k})$$

3) Time filter and swap of arrays

$$\begin{aligned}
U^{t-1}(\text{i,j,k}) &= \frac{\alpha * [U^{t-1}(\text{i,j,k}) + U^{t+1}(\text{i,j,k})] + \alpha_1 * U^{t}(\text{i,j,k})}{\alpha * [fse_{3U}^{t-1}(\text{i,j,k}) + fse_{3U}^{t+1}(\text{i,j,k})] + \alpha_1 * fse_{3U}^{t}(\text{i,j,k})} \\
U^{t}(\text{i,j,k}) &= \frac{U^{t+1}(\text{i,j,k})}{fse_{3U}^{t+1}(\text{i,j,k})}
\end{aligned}$$

A.8 Vertical velocity

Using the incompressibility hypothesis, the vertical velocity is computed by integrated the horizontal divergence from the bottom to the surface. The boundary conditions are $w = 0$ at the bottom (no flux) and, in the variable volume case, $w = 0$ at the surface. In the variable volume case, we must substract to the vertical velocity of the “fixed volume vertical velocity” the vertical velocity of moving grid-box interfaces due to the motion of free surface

Fixed volume case:

$$\begin{aligned} \frac{\partial w}{\partial k} &= -\chi e_3 \\ \text{where} \quad \chi &= \frac{1}{e_{1T} e_{2T} e_{3T}} \left(\frac{\partial}{\partial i}[e_{2U} e_{3U} u] + \frac{\partial}{\partial j}[e_{1V} e_{3V} v] \right) \\ \text{so that} \quad wn^{i,j,k} &= \sum_{jpk}^{k} \left(fse_{3T}^{i,j,k}\, hdivn^{k} \right) \end{aligned}$$

Variable volume case:

$$\begin{aligned} \frac{\partial w}{\partial k} &= -\chi e_3 - \frac{\partial e_{3T}}{\partial t} \\ wn^{i,j,k} &= \sum_{jpk}^{k} \left(fse_{3T}^{i,j,k}\, hdivn^{k} - \frac{ssha^{i,j} - sshb^{i,j}}{2\, rdt} e_{3T}^{i,j,k}\, mut^{i,j,k} \right) \end{aligned}$$