1 Ocean Dynamics (DYN)

1.1 Wetting and drying

There are two main options for wetting and drying code (wd): (a) an iterative limiter (il) and (b) a directional limiter (dl). The directional limiter is based on the scheme developed by Warner et al. [2013] for ROMS which was in turn based on ideas developed for POM by Oey [2006]. The iterative limiter is a new scheme. The iterative limiter is activated by setting $\mathrm{ln\_wd\_il} = .\mathrm{true.}$ and $\mathrm{ln\_wd\_dl} = .\mathrm{false.}$. The directional limiter is activated by setting $\mathrm{ln\_wd\_dl} = .\mathrm{true.}$ and $\mathrm{ln\_wd\_il} = .\mathrm{false.}$.

```
!----------------------------------------------------------------------
&namwad  !   Wetting and drying
!----------------------------------------------------------------------
   ln_wd_il          = .false   ! T/F activation of iterative limiter for  wetting and drying scheme
   ln_wd_dl          = .true.   ! T/F activation of directional llimiter for wetting drying scheme
   ln_wd_dl_bc       = .true.   ! T/F Directional limiteer Baroclinic option
   ln_wd_dl_rmp      = .true.   ! T/F Turn on directional limiter ramp
   rn_wdmin0         =  0.30    ! dpoth at which wetting/drying starts
   rn_wdmin1         =  0.2     ! Minimum wet depth on dried cells
   rn_wdmin2         =  0.0001  ! Tolerance of min wet depth on dried cells
   rn_wdld           =  2.5     ! Land elevation below which wetting/drying is allowed
   nn_wdit           =   20     ! Max iterations for W/D limiter
   rn_wd_sbcdep      =  5.0     ! Depth at which to taper sbc fluxes
   rn_wd_sbcfra      =  0.999   ! Fraction of SBC fluxes at taper depth (Must be <1)
/
```

The following terminology is used. The depth of the topography (positive downwards) at each (i, j) point is the quantity stored in array $\mathrm{ht\_wd}$ in the NEMO code. The height of the free surface (positive upwards) is denoted by ssh. Given the sign conventions used, the water depth, h, is the height of the free surface plus the depth of the topography (i.e. $\mathrm{ssh} + \mathrm{ht\_wd}$).

Both wd schemes take all points in the domain below a land elevation of $\mathrm{rn\_wdld}$ to be covered by water. They require the topography specified with a model configuration to have negative depths at points where the land is higher than the topography's reference sea-level. The vertical grid in NEMO is normally computed relative to an initial state with zero sea surface height elevation. The user can choose to compute the vertical grid and heights in the model relative to a non-zero reference height for the free surface. This choice affects the calculation of the metrics and depths (i.e. the $\mathrm{e3t\_0}, \mathrm{ht\_0}$ etc. arrays).

Points where the water depth is less than $\mathrm{rn\_wdmin1}$ are interpreted as "dry". $\mathrm{rn\_wdmin1}$ is usually chosen to be of order $0.05\mathrm{m}$ but extreme topographies with very steep slopes require larger values for normal choices of time-step. Surface fluxes are also switched off for dry cells to prevent freezing, boiling etc. of very thin water layers. The fluxes are tappered down using a $\tanh$ weighting function to no flux as the dry limit $\mathrm{rn\_wdmin1}$ is approached. Even wet cells can be very shallow. The depth at which to start tapering is controlled by the user by setting $\mathrm{rn\_wd\_sbcdep}$. The fraction (< 1) of sufrace fluxes to use at this depth is set by $\mathrm{rn\_wd\_sbcfra}$.

Both versions of the code have been tested in six test cases provided in the WAD_TEST_CASES configuration and in "realistic" configurations covering parts of the north-west European shelf. All these configurations have used pure sigma coordinates. It is expected that the wetting and drying code will work in domains with more general s-coordinates provided the coordinates are pure sigma in the region where wetting and drying actually occurs.

The next sub-section descrbies the directional limiter and the following sub-section the iterative limiter. The final sub-section covers some additional considerations that are relevant to both schemes.

1.1.1 Directional limiter (*wet_dry.F90*)

The principal idea of the directional limiter is that water should not be allowed to flow out of a dry tracer cell (i.e. one whose water depth is less than rn_wdmin1).

All the changes associated with this option are made to the barotropic solver for the non-linear free surface code within dynspg_ts. On each barotropic sub-step the scheme determines the direction of the flow across each face of all the tracer cells and sets the flux across the face to zero when the flux is from a dry tracer cell. This prevents cells whose depth is rn_wdmin1 or less from drying out further. The scheme does not force h (the water depth) at tracer cells to be at least the minimum depth and hence is able to conserve mass / volume.

The flux across each u-face of a tracer cell is multiplied by a factor zuwdmask (an array which depends on ji and jj). If the user sets ln_wd_dl_ramp = .False. then zuwdmask is 1 when the flux is from a cell with water depth greater than rn_wdmin1 and 0 otherwise. If the user sets ln_wd_dl_ramp = .True. the flux across the face is ramped down as the water depth decreases from 2 * rn_wdmin1 to rn_wdmin1. The use of this ramp reduced grid-scale noise in idealised test cases.

At the point where the flux across a u-face is multiplied by zuwdmask , we have chosen also to multiply the corresponding velocity on the "now" step at that face by zuwdmask. We could have chosen not to do that and to allow fairly large velocities to occur in these "dry" cells. The rationale for setting the velocity to zero is that it is the momentum equations that are being solved and the total momentum of the upstream cell (treating it as a finite volume) should be considered to be its depth times its velocity. This depth is considered to be zero at "dry" u-points consistent with its treatment in the calculation of the flux of mass across the cell face.

Warner et al. [2013] state that in their scheme the velocity masks at the cell faces for the baroclinic timesteps are set to 0 or 1 depending on whether the average of the masks over the barotropic sub-steps is respectively less than or greater than 0.5. That scheme does not conserve tracers in integrations started from constant tracer fields (tracers independent of x, y and z). Our scheme conserves constant tracers because the velocities used at the tracer cell faces on the baroclinic timesteps are carefully calculated by dynspg_ts to equal their mean value during the barotropic steps. If the user sets ln_wd_dl_bc = .True., the baroclinic velocities are also multiplied by a suitably weighted average of zuwdmask.

1.1.2 Iterative limiter (*wet_dry.F90*)

Iterative flux limiter (*wet_dry.F90*)

The iterative limiter modifies the fluxes across the faces of cells that are either already "dry" or may become dry within the next time-step using an iterative method.

The flux limiter for the barotropic flow (devised by Hedong Liu) can be understood as follows:

The continuity equation for the total water depth in a column

$$\frac{\partial h}{\partial t} + \nabla.(h\mathbf{u}) = 0. \tag{1.1}$$

can be written in discrete form as

$$\begin{align}
\frac{e_1 e_2}{\Delta t}(h_{i,j}(t_{n+1}) - h_{i,j}(t_e)) &= -(\mathrm{flxu}_{i+1,j} - \mathrm{flxu}_{i,j} + \mathrm{flxv}_{i,j+1} - \mathrm{flxv}_{i,j}) \tag{1.2}\\
&= \mathrm{zzflx}_{i,j}. \tag{1.3}
\end{align}$$

In the above h is the depth of the water in the column at point (i, j), $\mathrm{flxu}_{i+1,j}$ is the flux out of the "eastern" face of the cell and $\mathrm{flxv}_{i,j+1}$ the flux out of the "northern" face of the cell; t_{n+1} is the new timestep, t_e is the old timestep (either t_b or t_n) and $\Delta t = t_{n+1} - t_e$; $e_1 e_2$ is the area of the tracer cells centred at (i, j) and zzflx is the sum of the fluxes through all the faces.

The flux limiter splits the flux zzflx into fluxes that are out of the cell (zzflxp) and fluxes that are into the cell (zzflxn). Clearly

$$\mathrm{zzflx}_{i,j} = \mathrm{zzflxp}_{i,j} + \mathrm{zzflxn}_{i,j}. \tag{1.4}$$

The flux limiter iteratively adjusts the fluxes flxu and flxv until none of the cells will "dry out". To be precise the fluxes are limited until none of the cells has water depth less than $\mathrm{rn\_wdmin1}$ on step $n+1$.

Let the fluxes on the mth iteration step be denoted by $\mathrm{flxu}^{(m)}$ and $\mathrm{flxv}^{(m)}$. Then the adjustment is achieved by seeking a set of coefficients, $\mathrm{zcoef}^{(m)}_{i,j}$ such that:

$$\begin{aligned} \mathrm{zzflxp}^{(m)}_{i,j} &= \mathrm{zcoef}^{(m)}_{i,j}\,\mathrm{zzflxp}^{(0)}_{i,j} \\ \mathrm{zzflxn}^{(m)}_{i,j} &= \mathrm{zcoef}^{(m)}_{i,j}\,\mathrm{zzflxn}^{(0)}_{i,j} \end{aligned} \tag{1.5}$$

where the coefficients are 1.0 generally but can vary between 0.0 and 1.0 around cells that would otherwise dry.

The iteration is initialised by setting

$$\mathrm{zzflxp}^{(0)}{}_{i,j} = \mathrm{zzflxp}_{i,j}, \quad \mathrm{zzflxn}^{(0)}{}_{i,j} = \mathrm{zzflxn}_{i,j}. \tag{1.6}$$

The fluxes out of cell (i,j) are updated at the $m+1$th iteration if the depth of the cell on timestep t_e, namely $h_{i,j}(t_e)$, is less than the total flux out of the cell times the timestep divided by the cell area. Using (1.2) this condition is

$$h_{i,j}(t_e) - \mathrm{rn\_wdmin1} < \frac{\Delta t}{e_1 e_2}(\mathrm{zzflxp}^{(m)}_{i,j} + \mathrm{zzflxn}^{(m)}_{i,j}). \tag{1.7}$$

Rearranging (1.7) we can obtain an expression for the maximum outward flux that can be allowed and still maintain the minimum wet depth:

$$\begin{aligned} \mathrm{zzflxp}^{(m+1)}_{i,j} = \Big[& (h_{i,j}(t_e) - \mathrm{rn\_wdmin1} - \mathrm{rn\_wdmin2})\frac{e_1 e_2}{\Delta t} \\ & - \mathrm{zzflxn}^{(m)}_{i,j} \Big] \end{aligned} \tag{1.8}$$

Note a small tolerance ($\mathrm{rn\_wdmin2}$) has been introduced here *[Q: Why is this necessary/desirable?]*. Substituting from (1.5) gives an expression for the coefficient needed to multiply the outward flux at this cell in order to avoid drying.

$$\begin{aligned} \mathrm{zcoef}^{(m+1)}_{i,j} = \Big[& (h_{i,j}(t_e) - \mathrm{rn\_wdmin1} - \mathrm{rn\_wdmin2})\frac{e_1 e_2}{\Delta t} \\ & - \mathrm{zzflxn}^{(m)}_{i,j} \Big] \frac{1}{\mathrm{zzflxp}^{(0)}_{i,j}} \end{aligned} \tag{1.9}$$

Only the outward flux components are altered but, of course, outward fluxes from one cell are inward fluxes to adjacent cells and the balance in these cells may need subsequent adjustment; hence the iterative nature of this scheme. Note, for example, that the flux across the "eastern" face of the (i,j)th cell is only updated at the $m+1$th iteration if that flux at the mth iteration is out of the (i,j)th cell. If that is the case then the flux across that face is into the $(i+1,j)$ cell and that flux will not be updated by the calculation for the $(i+1,j)$th cell. In this sense the updates to the fluxes across the faces of the cells do not "compete" (they do not over-write each other) and one would expect the scheme to converge relatively quickly. The scheme is flux based so conserves mass. It also conserves constant tracers for the same reason that the directional limiter does.

Modification of surface pressure gradients (*dynhpg.F90*)

At "dry" points the water depth is usually close to $\mathrm{rn\_wdmin1}$. If the topography is sloping at these points the sea-surface will have a similar slope and there will hence be very large horizontal pressure gradients at these points. The WAD modifies the magnitude but not the sign of the surface pressure gradients (zhpi and zhpj) at such points by mulitplying them by positive factors (zcpx and zcpy respectively) that lie between 0 and 1.

We describe how the scheme works for the "eastward" pressure gradient, zhpi, calculated at the (i,j)th u-point. The scheme uses the ht_wd depths and surface heights at the neighbouring $(i+1,j)$ and (i,j) tracer points. zcpx is calculated using two logicals variables, $\mathrm{ll\_tmp1}$ and $\mathrm{ll\_tmp2}$ which are evaluated for each grid column. The three possible combinations are illustrated in figure 1.1.

The first logical, $\mathrm{ll\_tmp1}$, is set to true if and only if the water depth at both neighbouring points is greater than $\mathrm{rn\_wdmin1} + \mathrm{rn\_wdmin2}$ and the minimum height of the sea surface at the two points is greater than the maximum height of the topography at the two points:

$$\begin{aligned}
\mathrm{ll\_tmp1} = &\mathrm{MIN(sshn(ji,jj), sshn(ji+1,jj))} > \\
&\qquad \mathrm{MAX(-ht\_wd(ji,jj), -ht\_wd(ji+1,jj))\ .and.} \\
&\quad \mathrm{MAX(sshn(ji,jj) + ht\_wd(ji,jj),} \\
&\qquad\qquad \mathrm{sshn(ji+1,jj) + ht\_wd(ji+1,jj))} > \\
&\qquad \mathrm{rn\_wdmin1 + rn\_wdmin2}
\end{aligned} \tag{1.10}$$

The second logical, $\mathrm{ll\_tmp2}$, is set to true if and only if the maximum height of the sea surface at the two points is greater than the maximum height of the topography at the two points plus $\mathrm{rn\_wdmin1} + \mathrm{rn\_wdmin2}$

$$\begin{aligned}
\mathrm{ll\_tmp2} = &\mathrm{(ABS(sshn(ji,jj) - sshn(ji+1,jj)) > 1.E-12)\ .AND.} \\
&\mathrm{(MAX(sshn(ji,jj), sshn(ji+1,jj))} > \\
&\ \mathrm{MAX(-ht\_wd(ji,jj), -ht\_wd(ji+1,jj)) + rn\_wdmin1 + rn\_wdmin2).}
\end{aligned} \tag{1.11}$$

If $\mathrm{ll\_tmp1}$ is true then the surface pressure gradient, zhpi at the (i,j) point is unmodified. If both logicals are false zhpi is set to zero.

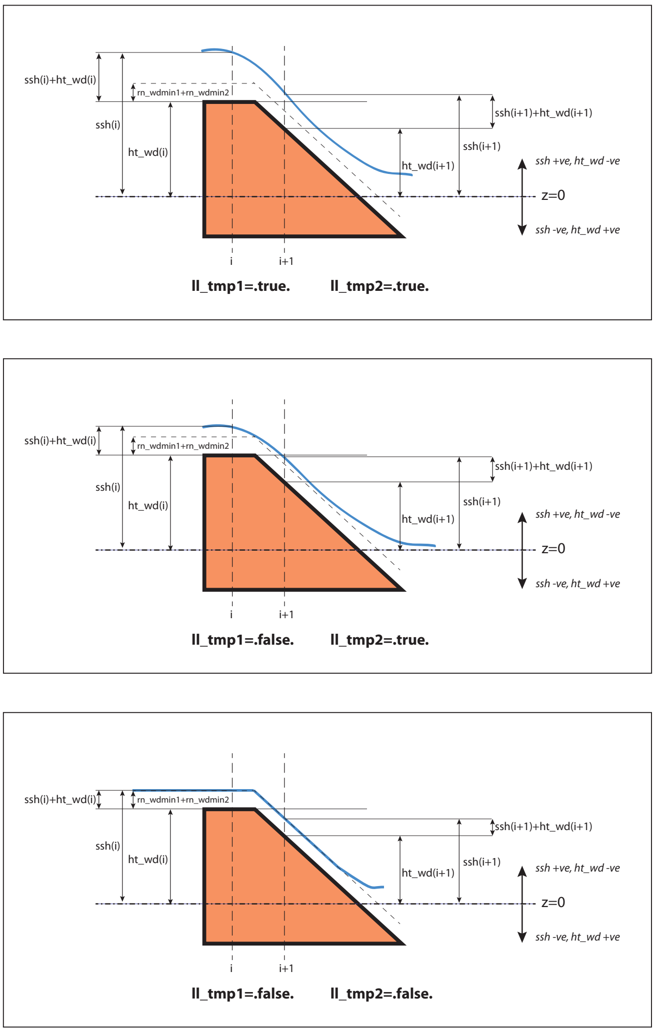

Figure 1.1: Illustrations of the three possible combinations of the logical variables controlling the limiting of the horizontal pressure gradient in wetting and drying regimes

If ll_tmp1 is true and ll_tmp2 is false then the surface pressure gradient is multiplied through by zcpx which is the absolute value of the difference in the water depths at the two points divided by the difference in the surface heights at the two points. Thus the sign of the sea surface height gradient is retained but the magnitude of the pressure force is determined by the difference in water depths rather than the difference in surface height between the two points. Note that dividing by the difference between the sea surface heights can be problematic if the heights approach parity. An additional condition is applied to ll_tmp2 to ensure it is .false.

in such conditions.

1.1.3 Additional considerations (*usrdef_zgr.F90*)

In the very shallow water where wetting and drying occurs the parametrisation of bottom drag is clearly very important. In order to promote stability it is sometimes useful to calculate the bottom drag using an implicit time-stepping approach.

Suitable specifcation of the surface heat flux in wetting and drying domains in forced and coupled simulations needs further consideration. In order to prevent freezing or boiling in uncoupled integrations the net surface heat fluxes need to be appropriately limited.

1.1.4 The WAD test cases (*usrdef_zgr.F90*)

This section contains details of the seven test cases that can be run as part of the WAD_TEST_CASES configuration. All the test cases are shallow (less than 10m deep), basins or channels with 4m high walls and some of topography that can wet and dry up to 2.5m above sea-level. The horizontal grid is uniform with a 1km resolution and measures 52km by 34km. These dimensions are determined by a combination of code in the *usrdef_nam.F90* module located in the WAD_TEST_CASES/MY_SRC directory and setting read in from the namusr_def namelist. The first six test cases are closed systems with no rotation or external forcing and motion is simply initiated by an initial ssh slope. The seventh test case introduces and open boundary at the right-hand end of the channel which is forced with sinousoidally varying ssh and barotropic velocities.

```
!
!-----------------------------------------------------------------------
&namusr_def
!-----------------------------------------------------------------------
   rn_dx = 1000.0
   rn_dz = 1.0
   nn_wad_test = 1
/
```

The `nn_wad_test` parameter can takes values 1 to 7 and it is this parameter that determines which of the test cases will be run. Most cases can be run with the default settings but the simple linear slope cases (tests 1 and 5) can be run with lower values of `rn_wdmin1`. Any recommended changes to the default namelist settings will be stated in the individual subsections.

Test case 7 requires additional `namelist_cfg` changes to activate the open boundary and lengthen the duration of the run (in order to demonstrate the full forcing cycle). There is also a simple python script which needs to be run in order to generate the boundary forcing files. Full details are given in subsection (1.1.4).

WAD test case 1 : A simple linear slope

The first test case is a simple linear slope (in the x-direction, uniform in y) with an adverse SSH gradient that, when released, creates a surge up the slope. The parameters are chosen such that the surge rises above sea-level before falling back and oscillating towards an equilibrium position. This case can be run with `rn_wdmin1` values as low as 0.075m. I.e. the following change may be made to the default values in `namelist_cfg` (for this test only):

```
!-----------------------------------------------------------------------
&namusr_def
!-----------------------------------------------------------------------
   nn_wad_test = 1
/
!-----------------------------------------------------------------------
&namwad  !   Wetting and drying
!-----------------------------------------------------------------------
   rn_wdmin1         =  0.075   ! Minimum wet depth on dried cells
/
```

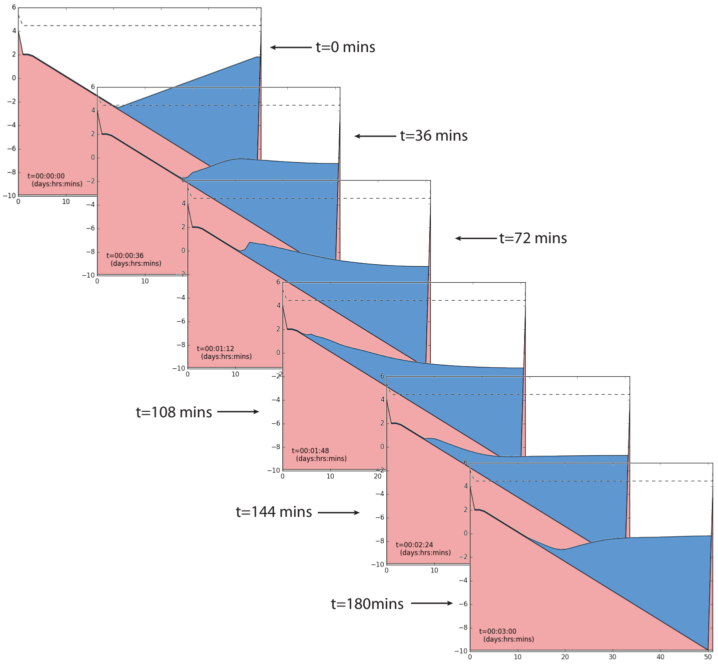

Figure 1.2: The evolution of the sea surface height in WAD_TEST_CASE 1 from the initial state (t=0) over the first three hours of simulation. Note that in this time-frame the resultant surge reaches to nearly 2m above sea-level before retreating.

WAD test case 2 : A parabolic channel

The second and third test cases use a closed channel which is parabolic in x and uniform in y. Test case 2 uses a gentler initial SSH slope which nevertheless demonstrates the ability to wet and dry on both sides of the channel. This solution requires values of rn_wdmin1 at least 0.3m (*Q.: A function of the maximum topographic slope?*)

```
!-----------------------------------------------------------------------
&namusr_def
!-----------------------------------------------------------------------
   nn_wad_test = 2
/
```

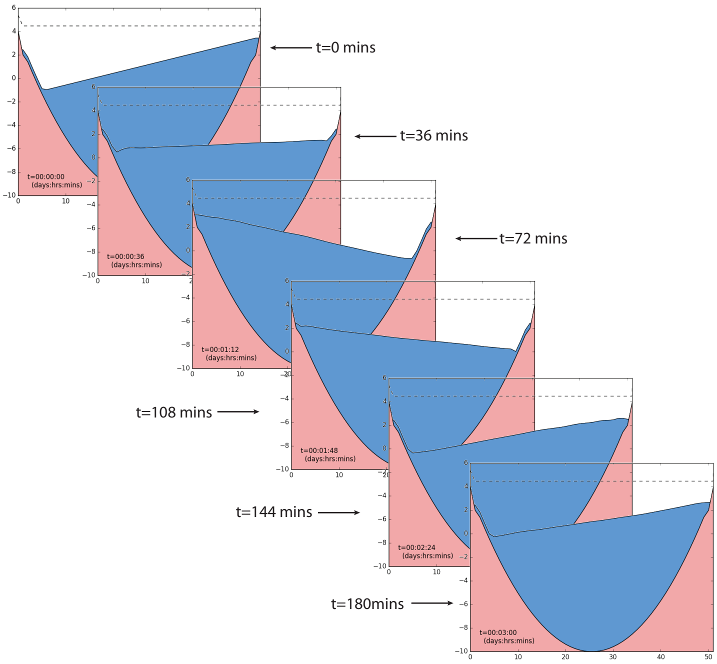

Figure 1.3: The evolution of the sea surface height in WAD_TEST_CASE 2 from the initial state (t=0) over the first three hours of simulation. Note that in this time-frame the resultant sloshing causes wetting and drying on both sides of the parabolic channel.

WAD test case 3 : A parabolic channel (extreme slope)

Similar to test case 2 but with a steeper initial SSH slope. The solution is similar but more vigorous.

```
!-----------------------------------------------------------------------
&namusr_def
!-----------------------------------------------------------------------
   nn_wad_test = 3
/
```

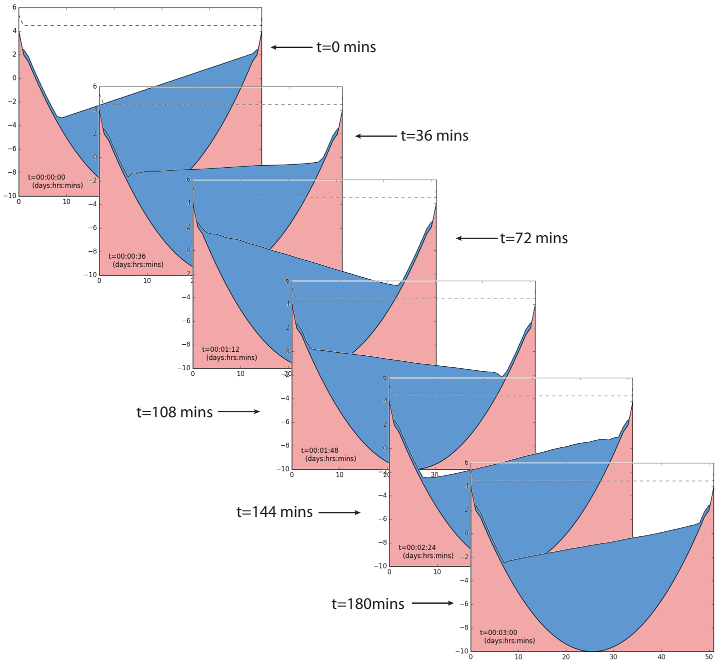

Figure 1.4: The evolution of the sea surface height in WAD_TEST_CASE 3 from the initial state (t=0) over the first three hours of simulation. Note that in this time-frame the resultant sloshing causes wetting and drying on both sides of the parabolic channel.

WAD test case 4 : A parabolic bowl

Test case 4 includes variation in the y-direction in the form of a parabolic bowl. The initial condition is now a raised bulge centred over the bowl. Figure 1.5 shows a cross-section of the SSH in the X-direction but features can be seen to propagate in all directions and interfere when return paths cross.

```
!-----------------------------------------------------------------------
&namusr_def
!-----------------------------------------------------------------------
   nn_wad_test = 4
/
!-----------------------------------------------------------------------
&namwad  !   Wetting and drying
!-----------------------------------------------------------------------
   rn_wdmin1          =  0.45    ! Minimum wet depth on dried cells
/
```

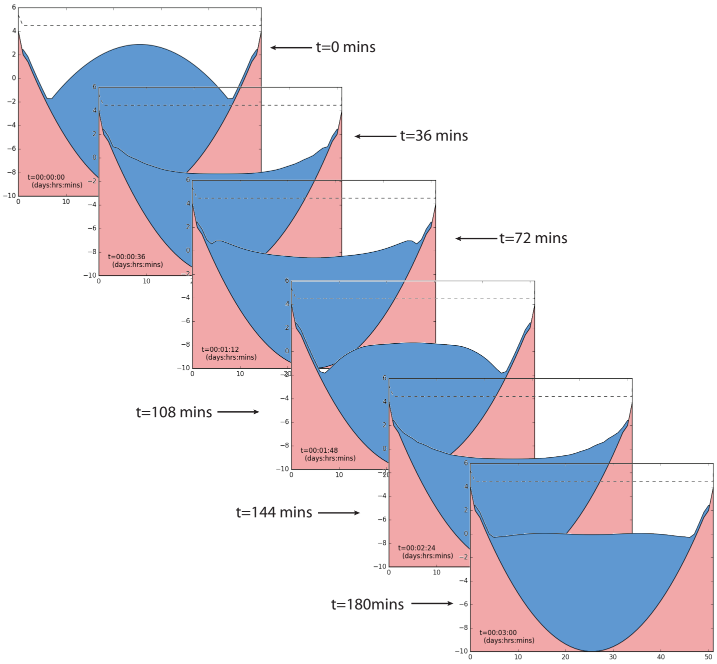

Figure 1.5: The evolution of the sea surface height in WAD_TEST_CASE 4 from the initial state (t=0) over the first three hours of simulation. Note that this test case is a parabolic bowl with variations occurring in the y-direction too (not shown here).

WAD test case 5 : A double slope with shelf channel

Similar in nature to test case 1 but with a change in slope and a mid-depth shelf.

```
!------------------------------------------------------------------------
&namusr_def
!------------------------------------------------------------------------
   nn_wad_test = 5
/
!------------------------------------------------------------------------
&namwad  !   Wetting and drying
!------------------------------------------------------------------------
   rn_wdmin1         =  0.15     ! Minimum wet depth on dried cells
/
```

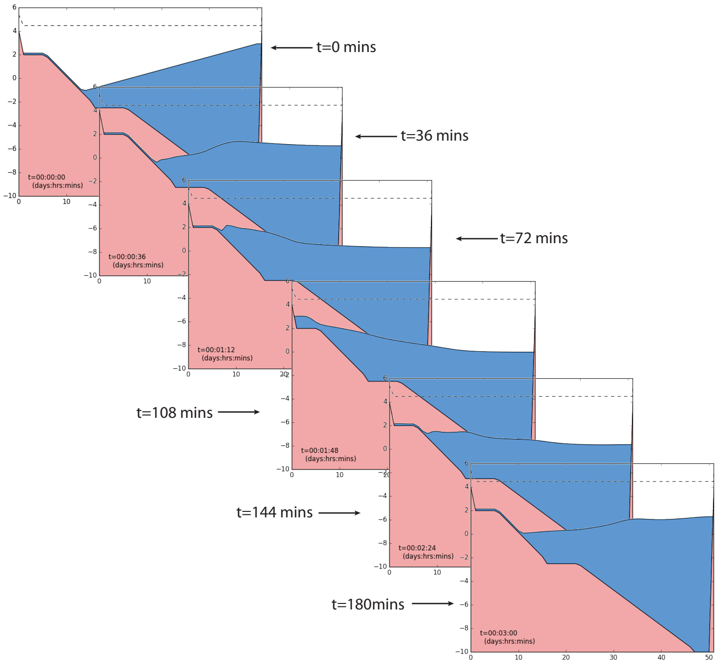

Figure 1.6: The evolution of the sea surface height in WAD_TEST_CASE 5 from the initial state (t=0) over the first three hours of simulation. The surge resulting in this case wets to the full depth permitted (2.5m above sea-level) and is only halted by the 4m high side walls.

WAD test case 6 : A parabolic channel with central bar

Test cases 1 to 5 have all used uniform T and S conditions. The dashed line in each plot shows the surface salinity along the y=17 line which remains satisfactorily constant. Test case 6 introduces variation in salinity by taking a parabolic channel divided by a central bar (gaussian) and using two different salinity values in each half of the channel. This step change in salinity is initially enforced by the central bar but the bar is subsequently over-topped after the initial SSH gradient is released. The time series in this case shows the SSH evolution with the water coloured according to local salinity values. Encroachment of the high salinity (red) waters into the low salinity (blue) basin can clearly be seen.

```
!-------------------------------------------------------------------------
&namusr_def
!-------------------------------------------------------------------------
   nn_wad_test = 6
/
```

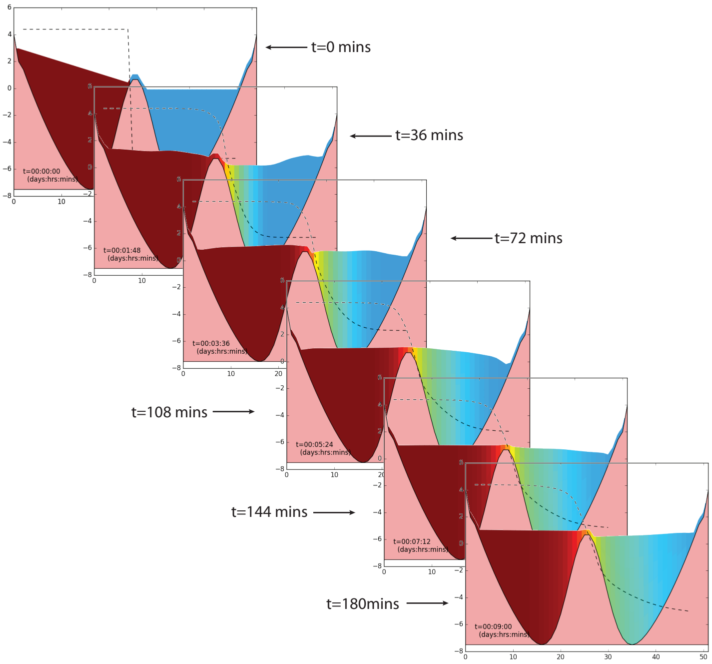

Figure 1.7: The evolution of the sea surface height in WAD_TEST_CASE 6 from the initial state (t=0) over the first three hours of simulation. Water is coloured according to local salinity values. Encroachment of the high salinity (red) waters into the low salinity (blue) basin can clearly be seen although the largest influx occurs early in the sequence between the frames shown.

WAD test case 7 : A double slope with shelf, open-ended channel

Similar in nature to test case 5 but with an open boundary forced with a sinusoidally varying ssh. This test case has been introduced to emulate a typical coastal application with a tidally forced open boundary. The bathymetry and setup is identical to test case 5 except the right hand end of the channel is now open and has simple ssh and barotropic velocity boundary conditions applied at the open boundary. Several additional steps and namelist changes are required to run this test.

```
!------------------------------------------------------------------------
&namusr_def
!------------------------------------------------------------------------
   nn_wad_test = 7
/
!------------------------------------------------------------------------
&namrun        !   parameters of the run
!------------------------------------------------------------------------
   nn_itend    =      9600  !  last  time step
/
!------------------------------------------------------------------------
&nambdy        !  unstructured open boundaries
!------------------------------------------------------------------------
    ln_bdy         = .true.
    nb_bdy         = 1                    !  number of open boundary sets
/
!------------------------------------------------------------------------
&namwad  !   Wetting and drying
!------------------------------------------------------------------------
   rn_wdmin1         =  0.150   ! Minimum wet depth on dried cells
/
```

In addition, the boundary condition files must be generated using the python script provided.

```
python ./makebdy_tc7.py
```

will create the following boundary files for this test (assuming a suitably configured python environment: python2.7 with netCDF4 and numpy):

```
bdyssh_tc7_m12d30.nc   bdyuv_tc7_m12d30.nc
bdyssh_tc7_m01d01.nc   bdyuv_tc7_m01d01.nc
bdyssh_tc7_m01d02.nc   bdyuv_tc7_m01d02.nc
bdyssh_tc7_m01d03.nc   bdyuv_tc7_m01d03.nc
```

These are sufficient for up to a three day simulation; the script is easily adapted if longer periods are required.

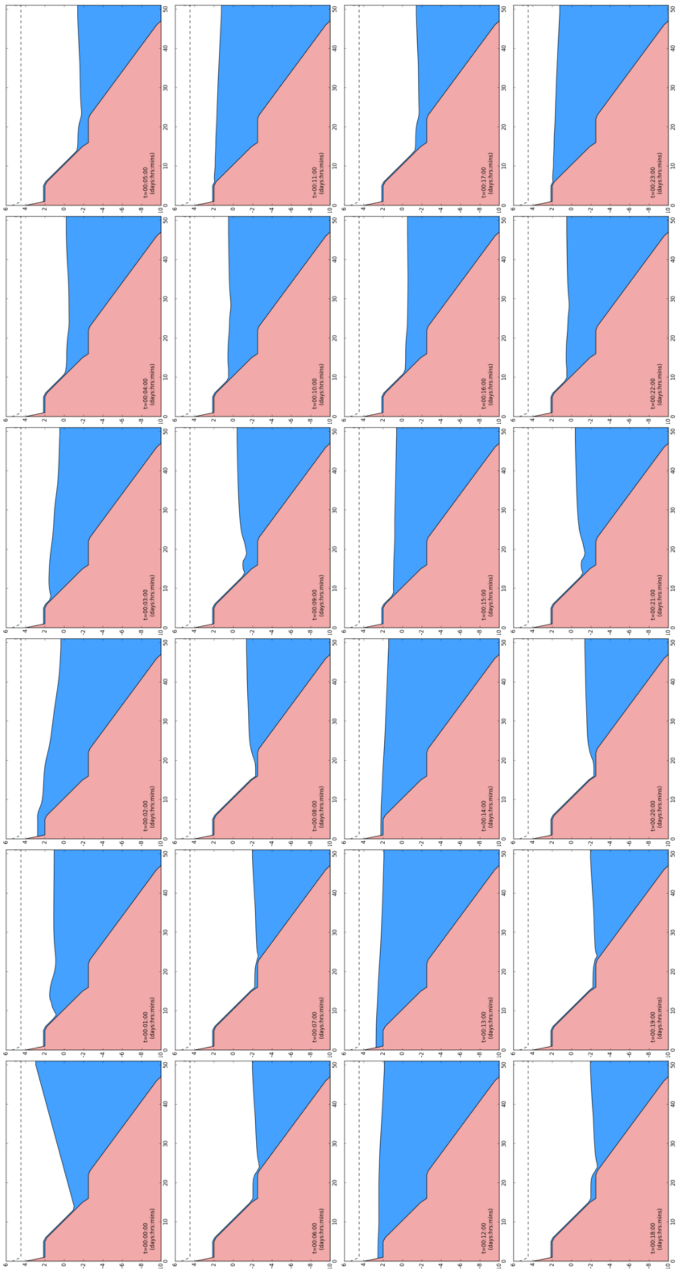

Figure 1.8: The evolution of the sea surface height in WAD_TEST_CASE 7 from the initial state (t=0) over the first 24 hours of simulation. After the initial surge the solution settles into a simulated tidal cycle with an amplitude of 5m. This is enough to repeatedly wet and dry both shelves.

Bibliography

Oey, L.-Y., 2006: An ogcm with movable land-sea boundaries. *Ocean Modelling*, **13** (**2**), 176 – 195, doi:https://doi.org/10.1016/j.ocemod.2006.01.001, URL `http://www.sciencedirect.com/science/article/pii/S1463500306000084`.

Warner, J. C., Z. Defne, K. Haas, and H. G. Arango, 2013: A wetting and drying scheme for roms. *Computers & Geosciences*, **58**, 54 – 61, doi:https://doi.org/10.1016/j.cageo.2013.05.004, URL `http://www.sciencedirect.com/science/article/pii/S0098300413001362`.