Chapter 15

Model grid and initial conditions

The chapter explains the setup of model grid, bathymetry and physical initial conditions, defined in the *Usrdef_Model.f90* routines:

- usrdef_grid: model grid and bathymetry
- usrdef_phsics: initial conditions

15.1 Model grid and bathymetry

This section describes the arrays defined in the routine usrdef_grid. The routine is called if modfiles(io_modgrd,1,1)%status='N'. If the status attribute equals 'R', the program calls the routines read_grid where the grid and bathymetric arrays are read in standard format.

The list below specifies all arrays which could be defined here.

gdelxglb(1:nc) Grid spacings in the X-direction (meters or fractional degrees longitude) in case of a non-uniform rectangular grid. In the case of a rotated grid the spacings need to be given in transformed coordinates [m or fractional degrees longitude].

gdelyglb(1:nc) Grid spacings in the Y-direction (meters of fractional degrees latitude) in case of a non-uniform rectangular grid. In the case of a rotated grid the spacings need to be given in transformed coordinates [m or fractional degrees latitude].

gxcoordglb(1:nc,1:nr) X-coordinates (Cartesian or spherical) at the UV-nodes (corner points) relative to the reference location in case of a fully curvilinear grid [m or fractional degrees longitude].

gycoordglb(1:nc,1:nr) Y-coordinates (Cartesian or spherical) at the UV-nodes (corner points) relative to the reference location in case of a fully curvilinear grid [m or fractional degrees latitude].

gsigcoord(nz+1) σ-coordinates in case of an horizontally uniform and vertically non-uniform σ-grid. Define only when iopt_grid_vtype=2 and iopt_grid_vtype_transf=0. Note that gsigcoord(1)=0 and gsigcoord(nz+1)=1.

gscoordglb(1:nc-1,1:nr-1,nz+1) σ-coordinates in case of non-uniform σ-grid in both the vertical and horizontal directions. Define only when iopt_grid_vtype=3 and iopt_grid_vtype_transf=0. Note that gscoordglb(:,:,1)=0 and gscoordglb(:,:,nz+1)=1.

depmeanglb(1:nc-1,1:nr-1) mean water depths (bathymetry) [m]

iobu(nobu) (Global) X-index of the (West/East) open boundary points at U-nodes

jobu(nobv) (Global) Y-index of the (West/East) open boundary points at U-nodes

iobv(nobv) (Global) X-index of the (South/North) open boundary points at V-nodes

jobv(nobv) (Global) Y-index of the (South/North) open boundary points at V-nodes

Remarks

- In case of a parallel application, all arrays are defined on the "global" grid.
- Coordinate units in the horizontal are m or fractional degrees depending on whether iopt_grid_sph equals 0 or 1.
- In case of a uniform vertical σ-grid (iopt_grid_vtype=1), the σ-levels are uniformly distributed over the vertical. This is automatically performed by the program.
- In case that a uniform water depth is taken, no bathymetry needs to be defined here. Before calling usrdef_grid the program sets, by default, all water depths to the value of depmean_cst defined in usrdef_mod_params.
- A number of additional (scalar) parameters may need to be defined in usrdef_params (see Section 14.6.2).

- nobu and nobv are the total (open sea and river) number of open boundary points at U- and V-nodes. They are obtained from the values of nosbu, nrvbu, nosbv, nrvbv defined in usrdef_mod_params.

- An open boundary must be located either at a velocity node on the edge of the "physical" domain or a velocity node separating a land and a sea cell. Note that open boundaries cannot be located near cells where a drying process can take place.

- If the model is applied to simulate the inundation of land areas (iopt_fld=2), the bathymetry should include the land topography of the areas which can potentially be flooded. For details see Section 5.4.2.

- Water depths are measured downwards. This means, that in case a inundation scheme is applied, land topography is represented by negative values of depmeanglb. Values equal to the data flag depmean_flag are considered as permanent land and excluded from the calculation. In case no inundation is applied, it is advised to set depmean_flag to zero, all values of the bathymetry at locations above sea level to zero and (eventually) impose a minimum water depth at sea to prevent accidental drying and a crash of the program.

15.2 Initial physical conditions

Routine usrdef_phsics defines the initial conditions for the physical module. The routine is called by *all* processes if modfiles(io_phsics,1,1)%status='N'. If the status attribute equals 'R', the program calls the routine read_phsics where the initial conditions are obtained in standard format.

The initialisation of some variables depends on the values of switches. Some arrays are defined "locally". In that case, the arrays must be given with a different shape, depending on whether the model is applied in parallel or serial mode (see below). Note that the shapes may be different from the ones used in the program itself.

2-D mode

udvel Depth-integrated current in the X-direction [m^2/s]. Shape is (nc,nr) in serial and (ncloc,nrloc) in parallel mode.

vdvel Depth-integrated current in the Y-direction [m^2/s]. Shape is (nc,nr) in serial and (ncloc,nrloc) in parallel mode.

zeta Surface elevation [m]. Shape is (nc-1,nr-1) in serial and (ncloc,nrloc) in parallel mode.

3-D currents

uvel Current in the X-direction [m/s]. Shape is (nc,nr,nz) in serial and (ncloc,nrloc,nz) in parallel mode. Define only for 1-D and 3-D applications.

vvel Current in the Y-direction [m/s]. Shape is (nc,nr,nz) in serial and (ncloc,nrloc,nz) in parallel mode. Define only for 1-D and 3-D applications.

wvel Transformed vertical current [m/s]. Shape is (nc-1,nr-1,nz+1) in serial and (ncloc,nrloc,nz+1) in parallel mode. Define only for 3-D applications.

density arrays

temp Temperature [deg C]. Define only when iopt_temp>0.

sal Salinity [PSU]. Define only when iopt_sal>0.

The shape of both arrays is (nc-1,nr-1,nz) in serial and (ncloc,nrloc,nz) in parallel mode.

turbulence arrays

Turbulence arrays can be defined if a RANS model (see Section 4.4.3) is selected (iopt_vdif_coef=3).

tke Turbulent kinetic energy [J/kg].

zlmix Mixing length [m]. Define only for a two-equation $k-l$ model (iopt_turb_ntrans=2 and iopt_turb_param=1).

dissip Dissipation of turbulent energy [W/kg]. Define only for a two-equation $k-\varepsilon$ model (iopt_turb_ntrans=2 and iopt_turb_param=2).

The shapes of the arrays are (nc-1,nr-1,2:nz) in serial and (ncloc,nrloc,2:nz) in parallel mode.

arrays for bottom stres

bdragcoefatc Bottom drag coefficient [-]. Define only when iopt_bstres_drag=2.

zroughatc Bottom roughness length [m]. Define only when iopt_bstres_drag=4.

The shapes of the arrays are (nc-1,nr-1) in serial and (ncloc,nrloc) in parallel mode.

tidal arrays

fnode_obc(nconobc) nodal factors of tidal constituents at open boundaries [-]

phase_obc(nconobc) Greenwich arguments of tidal constituents at open boundaries [rad]

fnode_astro(nconastro) nodal factors of tidal constituents used for astronomical forcing [-]

phase_astro(nconastro) Greenwich arguments of tidal constituents used for astronomical forcing [rad]

The phases need to be defined if iopt_astro_pars=0, the nodal factors if iopt_astro_pars=0,1. Otherwise, the phases and nodal factors will be automatically initialised by the program itself.

open boundary arrays

The arrays below represent "storage" arrays used in case the open boundary conditions at specific nodes require the solution of a differential equation in time. The following arrays are used for the 2-D mode

obc2uvatu(nobu,2) 2-D mode at U-open boundaries (iopt_obc_2D=1)

obc2uvatv(nobv,2) 2-D mode at V-open boundaries (iopt_obc_2D=1)

The meaning of the last index in the open boundary arrays depends on the type of open boundary condition[1, 2].

6 : Orlanski

1: U or V at the interior point nearest to the open boundary

2: U or V at the interior point second nearest to the open boundary

7 : Camerlengo-O'Brien

2: U or V at the interior point second nearest to the open boundary

9 : Flather with specified elevation

2: U^L or V^L at the previous time step

10: Flather

1: ζ^L at the previous time step

2: U^L or V^L at the previous time step

[1]The number in the main list below refers to the value of ityp2dobu or ityp2dobv defined in Section 16.1.1.1 or to the description list in Section 4.10.1.

[2]The number in the list refers to the index in the last dimension of the array.

11: Røed and Smedstad

  1: outgoing characteristic

12: characteristic method with specified elevation

  1: outgoing characteristic

13: characteristic method using zero normal gradient

  1: outgoing characteristic

  2: incoming characteristic

The following additional arrays are used for application of the open boundary conditions of scalar quantitities (baroclinic currents, T, S), denoted by ψ below

obc3uvatu(nobu,nz,2)	3-D baroclinic current at U-open boundaries (iopt_obc_3D=1)
obc3uvatv(nobv,nz,2)	3-D baroclinic current at V-open boundaries (iopt_obc_3D=1)
obctmpatu(nobu,nz,0:2)	temperature at U-open boundaries (iopt_obc_temp=1)
obctmpatv(nobv,nz,0:2)	temperature at V-open boundaries (iopt_obc_temp=1)
obcsalatu(nobu,nz,0:2)	salinity at U-open boundaries (iopt_obc_sal=1)
obcsalatv(nobv,nz,0:2)	salinity at V-open boundaries (iopt_obc_sal=1)

The index of the last dimension in the array has the following meaning

0: open boundary value of ψ at the open boundary (scalars only)

1: value of ψ at the interior point nearest to the open boundary

2: value of ψ at the interior point second nearest to the open boundary

The open boundary arrays are usually not created by the user but obtained from a previous run in which case the data are read from an external file.

remarks

- Initial condition arrays can be obtained from an external data file. This may create an extra overhead if the program runs in parallel mode, since the data arrays are usually defined on a global grid. This mean that, after reading the global arrays, they must be distributed onto the local grids. Details are found in the example code of usrdef_phsics in the file *examples/Usrdef.f90*.

- In case an inundation scheme is applied, a negative surface elevation can be supplied in land areas, which are taken intially as dry, but may be inundated at a later time. See Section 5.4.2 for details.

defaults If a variable is not defined, the following defaults are assumed (see Section 4.11):

- transports, currents, elevations: 0
- temperature: uniform value given by temp_ref, defined in usrdef_mod_params
- salinity: uniform value given by sal_ref, defined in usrdef_mod_params
- turbulence: see Section 4.11
- bottom stress arrays: 0
- astronomical phases: 0
- nodal factors: 1
- open boundary arrays: 0