Chapter 5

Numerical methods

5.1 Introduction

The numerical methods described in this chapter are based to a large extent upon previous work described in the COHERENS V1 manual (Luyten *et al.*, 1999).

Conservative finite differences (equivalent to a finite volume technique for the Cartesian mesh) are used to discretise the mathematical model in space. The grid chosen for horizontal discretisation is the well known Arakawa "C" grid (Mesinger & Arakawa, 1976) which staggers the currents and pressure/elevation nodes to give a good representation of the crucial gravity waves and provides simple representations of open and coastal boundaries. As discussed in Section 4.1 the model equations are solved on a rectangular or curvilinear grid in the horizontal and a σ- or extended σ-coordinate grid in the vertical, whereby varying surface and bottom boundaries are transformed into constant surfaces. This provides for accurate representation of surface and bottom boundary processes. It also results in an equal number of cells in each vertical water column.

Two options are available to solve the hydrodynamic equations. The original implementation in COHERENS used the mode-splitting technique as in the model of Blumberg & Mellor (1987) to solve the momentum equations. This method consists in solving the depth-integrated momentum and continuity equations for the "external" or barotropic mode with a small time step to satisfy the stringent CFL stability criterium for surface gravity waves, and the 3-D momentum and scalar transport equations for the "internal" or "baroclinic" mode with a larger time step. A "predictor" and a "corrector" step are applied for the horizontal momentum equations to satisfy the basic requirement that the depth-integrated currents obtained from the the 2-D

and 3-D mode equations, have identical values.

Recently, the possibility to solve the momentum equations semi-implicitly as in the model of Chen (2003), based on the original work of Casulli & Cheng (1992) was implemented. With this method, there is no longer need to solve the depth-integrated momentum equations. The stringent CFL stability criterium is relaxed by treating the terms that provoke the barotropic mode in an implicit manner. After an explicit "predictor" step, velocities are corrected with the implicit free surface correction in the "corrector" step. In this method, the free surface correction follows from the inversion of the elliptic free surface correction equation obtained from the 2-D continuity equation.

Much effort has been made to implement suitable schemes for the advection of momentum and scalars. A variety of schemes are available from the literature, e.g. second and higher order central and upwind schemes (see Hirsch, 1990, for a review), Flux Corrected Transport (FCT; Boris & Book, 1979), Total Variation Diminishing (TVD; Roe, 1986; Sweby, 1984), Quadratic Upstream Interpolation for Convective Kinematics (QUICK; Leonard, 1979), Second Order Moments (SOM; Prather, 1986; Hofmann & Maqueda, 2006), Piecewise Parabolic Method (PPM; Colella & Woodward, 1984; James, 1996). Implementing different schemes within the same model code is a tedious task since most higher order schemes impose a coupling between space and time discretisation. The basic choice in the program will therefore be limited to the upwind and the TVD scheme to reduce the programming and computational overhead. The latter scheme is implemented with the symmetrical operator splitting method for time integration and can be considered as a useful tool for the simulation of frontal structures and areas with strong current gradients. The upwind scheme, on the other hand, is only first order accurate and therefore more diffusive, and should be used if CPU time is considered of more importance than accuracy.

The following additional issues are noted:

- When the mode-splitting method is used, scalar quantities are advected with a "filtered" velocity (u_f,v_f) derived from the "corrected" baroclinic currents and the depth-integrated current averaged over the internal time step (Deleersnijder, 1993).

- Sink terms are discretised explicitly in time for cell-centered scalars to make the scheme more conservative, whereas a quasi-implicit formulation is implemented for turbulence transport to ensure positivity (Patankar, 1980).

This chapter is organised as follows:

- The model grid, the grid indexing system and notational conventions are described in Section 5.2.
- The solution of the momentum equations is presented in Section 5.3.
- The scalar transport equations are discussed in Section 5.5.
- Numerical aspects of the turbulence module are given in Section 5.6.
- The discretisations for one-dimensional (water column) and two-dimensional (depth-averaged) applications are discussed in Section 5.7.
- The general solution procedure is summarised in Section 5.8.

5.2 Model grid and discretisations

5.2.1 Grid nodes and indexing system

Figure 5.1 shows the horizontal layout of the C-grid domain as it appears in curvilinear coordinates (ξ_1, ξ_2). A normalisation is applied so that $\Delta\xi_1$=$\Delta\xi_2$=1. For convenience, the notations X and Y will be used for ξ_1 and ξ_2. It is remarked that X and Y do not refer to Cartesian axes in general. The following nodes can be distinguished:

- C-nodes (empty circles): located at the centers of the grid cells, used for 2-D and 3-D scalar quantitities (elevations, water depths, ...) and wind components
- U-nodes (horizontal bars): at the centers of the left (West) and right (East) cell faces, used for the X-components of vectors except the surface wind (transports, depth-mean currents, bottom stress, ...)
- V-nodes (vertical bars): at the centers of the lower (South) and upper (North) cell faces, used for the Y-components of vectors except the surface wind (transports, depth-mean currents, bottom stress, ...)
- UV-nodes (solid circles): at the corners of the grid cells, used for the horizontal coordinate arrays which determine the geographical location of the grid

Each horizontal grid cell has an index, generally denoted by 'i', in the X-direction between 1 and nc and an index ('j') in the Y-direction between 1 and nr. The indices refer to the position of a variable at its "natural" node (C-, U-, V-, UV-node). This is illustrated in Figure 5.2.

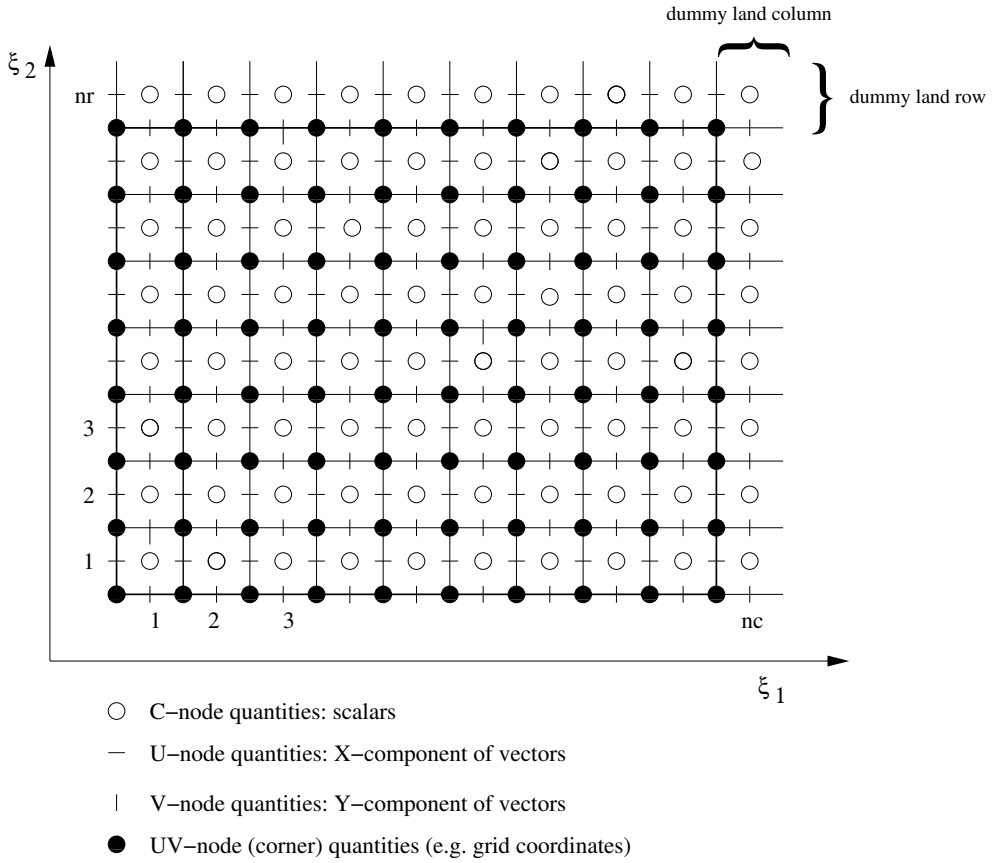

Figure 5.1: Layout of the (global) computational grid in the horizontal.

As shown in Figure 5.1, the last column (to the East) and the last row (to the North) are open ended. In this way the domain contains the same number of C-, U-, V- and UV-nodes. This was not implemented in COHERENS V1 but introduced in the new version to allow a more efficient domain decomposition in case of a parallel application. The drawback is that the C-node grid points with X-index nc or Y-index nr have to be declared as spurious dry cells. This means in practice that, whereas the computational size of the domain is nc×nr, the physical size is (nc-1)×(nr-1) for C-node, nc×(nr-1) for U-node, (nc-1)×nr for V-node and nc×nr for UV-node quantities.

In analogy with the horizontal directions, a staggered grid is used in the vertical as well. The water column is divided into nz layers. The layers, which in transformed vertical coordinates have equal sizes, are illustrated in Figure 5.3. The previous C-nodes are vertically located at the midst of each layer. A new type of node, the W-node, is introduced located at the layer itself, i.e. vertically between the C-nodes and at the bottom and the surface.

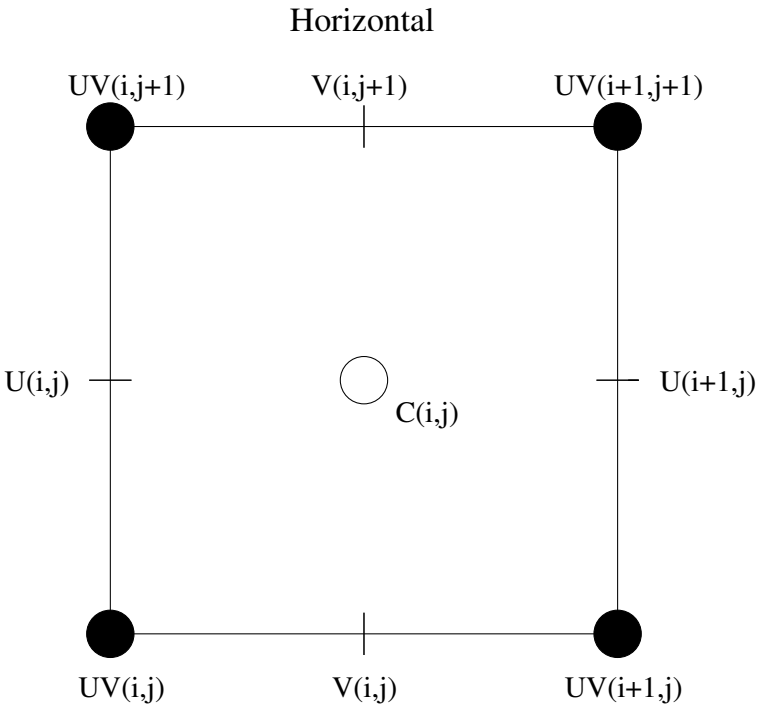

Figure 5.2: Grid indexing in the horizontal plane.

The vertical position of a 3-D model variable is determined by the vertical (Z-)index ("k") which varies between 1 and nz for C-node and between 1 and nz+1 for W-node quantities.

The grid indexing system for the full 3-D mode is shown in Figure 5.4. Combining horizontal and vertical nodes, new types of "combined" nodes arise. The following nodal types are considered in the program:

- C-nodes: at the center of a 3-D grid cell
- U-nodes: at the center of a West/East lateral face
- V-nodes: at the center of a South/North lateral face
- UV-nodes: along the intersection lines of the lateral faces horizontally, halfway between the lower and upper surface vertically
- W-nodes: at the centers of the lower and upper boundary faces
- UW-nodes: as the U-nodes horizontally, as the W-nodes vertically
- VW-nodes: as the V-nodes horizontally, as the W-nodes vertically

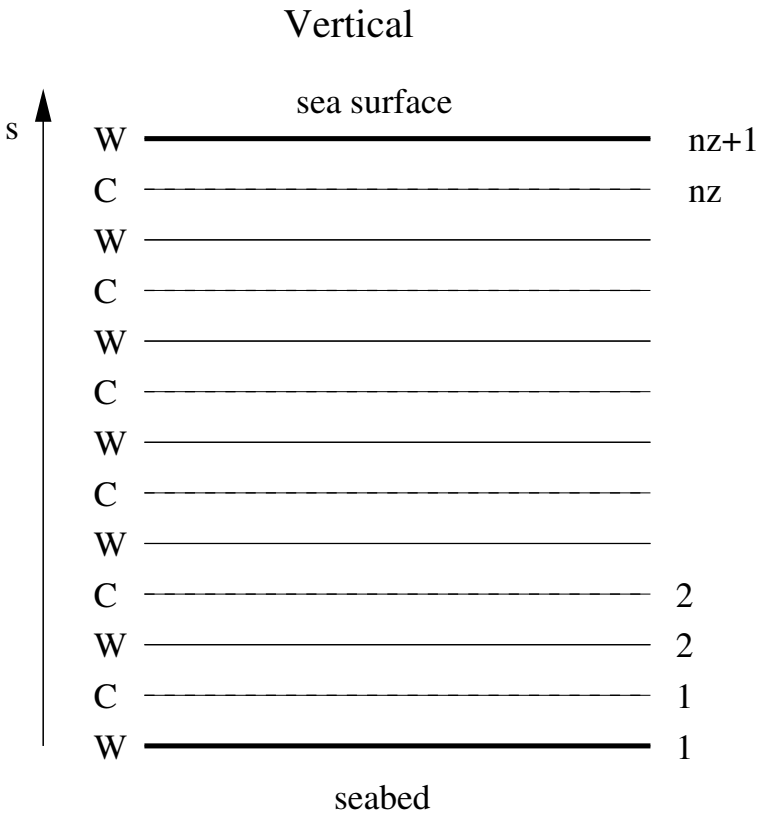

Figure 5.3: Layout of the computational grid in the vertical.

- UVW-nodes: at the corners of a 3-D grid cell (as UV-nodes horizontally and as W-nodes vertically)

The W-nodes are used for the (transformed) vertical current ω and for turbulence variables (k, ε, l, vertical diffusion coefficients and related variables). The UW-, VW- and UVW-nodes are only needed by the program for local internal variables.

The lower bound of all grid indices is 1, the upper boundary depends on the nodal type and on whether it is taken along the computational or physical domain. A complete listing is given in Table 5.1.

5.2.2 Open boundaries

Open boundaries are defined as locations on the model grid where the solution of the discretised model equations requires values of the transport variable(s) located outside the physical domain. Open boundary conditions have to be specified at those locations. The program distinguishes four types of open

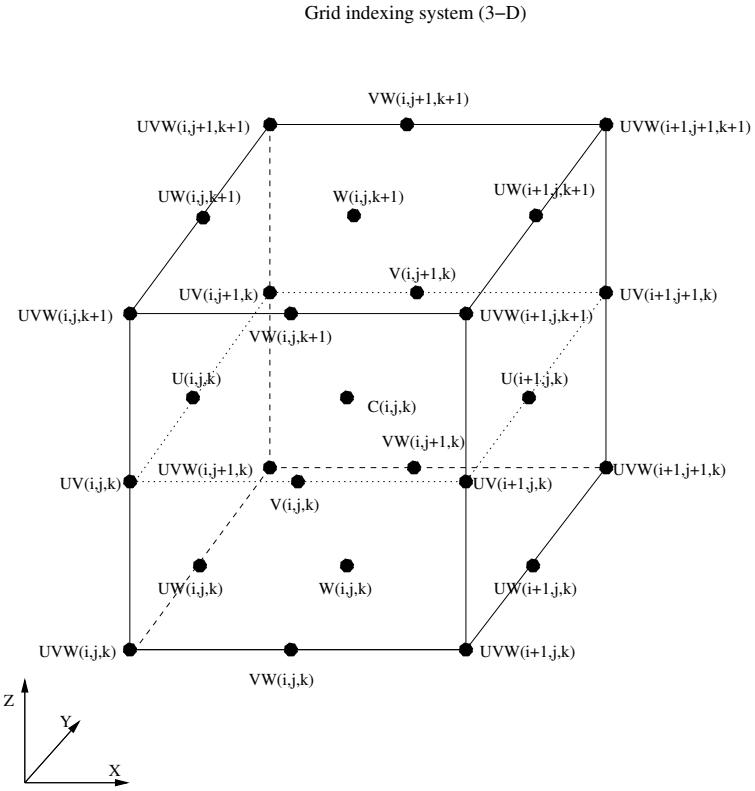

Figure 5.4: Grid indexing in three-dimensional space.

boundaries:

- U-open boundaries at U-velocity nodes needed to determine the values of U,u and the advective/diffusive fluxes of scalars in the X-direction
- V-open boundaries at V-velocity nodes needed to determine the values of V,v and the advective/diffusive fluxes of scalars in the Y-direction
- X-open boundaries at UV-nodes needed to determine the cross-stream advective/diffusive fluxes of v and V
- Y-open boundaries at UV-nodes needed to determine the cross-stream advective/diffusive fluxes of u and U

5.2.3 Conventions

A quantity taken at a grid point on its natural node is written as Q_{ij} for a 2-D or Q_{ijk} for a 3-D variable. To simplify the notations, the indices i, j, k

Table 5.1: Upper bounds for the grid indices (i,j,k) as function of nodal type.

Node	Computational			Physical		
	X-index	Y-index	Z-index	X-index	Y-index	Z-index
C	nc	nr	nz	nc-1	nr-1	nz
U	nc	nr	nz	nc	nr-1	nz
V	nc	nr	nz	nc-1	nr	nz
UV	nc	nr	nz	nc	nr	nz
W	nc	nr	nz+1	nc-1	nr-1	nz+1
UW	nc	nr	nz+1	nc	nr-1	nz+1
VW	nc	nr	nz+1	nc-1	nr	nz+1
UVW	nc	nr	nz+1	nc	nr	nz+1

are omitted if no confusion is possible. This means e.g. that $Q_{i,j+1,k}$ (3-D quantity) can be written as Q_{j+1} or that $\overline{Q}_{i-1}$ (2-D quantity) is the same as $\overline{Q}_{i-1,j}$.

If a quantity needs to be evaluated at a point, different from its natural position, its value is determined by taking an average over the neighbouring points. This is indicated by one of the superscripts ^{c}, ^{u}, ^{v}, ^{w}, ... referring to the point at which the quantity is interpolated. The program allows to use uniform averaging with equal weight factors or non-uniform averaging with unequal weights (see Section 10.2). To illustrate the convention, uniform averaging is assumed here for simplicity. The Coriolis terms in the momentum equations require a 4-point interpolation of the u and v velocities:

$$\begin{aligned} u^v_{ijk} &= \frac{1}{4}(u_{ijk} + u_{i,j-1,k} + u_{i+1,jk} + u_{i+1,j-1,k}) \\ v^u_{ijk} &= \frac{1}{4}(v_{ijk} + v_{i-1,jk} + v_{i,j+1,k} + v_{i-1,j+1,k}) \end{aligned} \tag{5.1}$$

The next example is a centered quantity Q evaluated at respectively the U-, V-, W-, UW- and VW-node with the same index values:

$$\begin{aligned} Q^u_{ijk} &= \frac{1}{2}(Q_{i-1,jk} + Q_{ijk}) \\ Q^v_{ijk} &= \frac{1}{2}(Q_{i,j-1,k} + Q_{ijk}) \\ Q^w_{ijk} &= \frac{1}{2}(Q_{ij,k-1} + Q_{ijk}) \\ Q^{uw}_{ijk} &= \frac{1}{4}(Q_{ij,k-1} + Q_{ijk} + Q_{i-1,j,k-1} + Q_{i-1,j,k}) \end{aligned}$$

$$Q^{vw}_{ijk} = \frac{1}{4}(Q_{ij,k-1} + Q_{ijk} + Q_{i,j-1,k-1} + Q_{i,j-1,k}) \tag{5.2}$$

A double index notation of the form $i_1 : i_2$ or $j_1 : j_2$ is sometimes introduced in expressions related to open boundary conditions, where the first index i_1 (j_1) is used at western (southern) boundaries and the second index i_2 (j_2) at eastern (northern) boundaries, such as in the following example expressions

$$u_{i+1:i-1,jk}\ ,\ v_{i,j:j-1,k}$$

5.2.4 Space discretisation

The grid is defined by specifying the following three arrays:

- the x_1-coordinates (in Cartesian or spherical coordinates) $x_{1;ij}$ of the cell corners (represented by the 2-D array gxcoordglb(nc,nr))
- the x_2-coordinates (in Cartesian or spherical coordinates) $x_{2;ij}$ of the cell corners (represented by the 2-D array gycoordglb(nc,nr))
- the σ-coordinates σ_{ijk} of the W-nodes (represented by the array gscoordglb(nc,nr,nz+1)). Note that $\sigma_{ij1} = 0$ (bottom) and $\sigma_{ij,N_z+1} = 1$ (surface).

As discussed in Sections 4.1.2-4.1.4, the grid spacings Δx_1, Δx_2, Δz are set equal to respectively the metric coefficients h_1, h_2, h_3 by normalisation. The latter notation will be used for convenience in the following.

Spatial differences in the x_1-, x_2- or vertical direction are represented respectively by the operators Δ_x, Δ_y, Δ_z. The superscript ^{c}, ^{u}, ^{v}, ^{w}, ^{uw}, ^{vw} or ^{uv} indicates the grid (nodal) location of the result. This is illustrated with the following examples (where Q represents a centered quantity in the third example):

$$\begin{aligned}
\Delta^c_x u_{ijk} &= u_{i+1,j,k} - u_{ijk} \\
\Delta^v_y u^c_{ijk} &= \frac{1}{2}(u_{ijk} + u_{i+1,jk} - u_{i,j-1,k} - u_{i+1,j-1,k}) \\
\Delta_z Q_{ijk} &= Q_{ijk} - Q_{ij,k-1} \\
\Delta^c_y V_{ij} &= V_{i,j+1} - V_{ij}
\end{aligned} \tag{5.3}$$

Grid spacings are "naturally" evaluated at the cell centre. Conforming the previous rules interpolated values at other grid locations are indicated by a superscript, e.g. $h^u_{1;ij}$, $h^w_{3;ijk}$. Note that the grid indices on the left hand side of the expressions (5.3) refer to the destination node and not the source node of the interpolation. An overview of all subscript and superscript notations, used in this chapter, is given in Table 5.2.

Table 5.2: Subscript and superscript notation used in the numerical discretisation formulae.

Type	Purpose
subscripts	
i	X-index of the variable on the model grid (between 1 and either nc-1 or nc)
j	Y-index of the variable on the model grid (between 1 and either nr-1 or nr)
k	vertical index of the variable on the model grid (between 1 and either nz or nz+1)
i_1:i_2	expression used in the spatial discretisation of open boundary conditions, whereby the first index is taken at the western and the second index at the eastern boundary
j_1:j_2	expression used in the spatial discretisation of open boundary conditions, whereby the first index is taken at the southern and the second index at the northern boundary
superscripts	
c	quantity evaluated or interpolated at the cell centre
u	quantity evaluated or interpolated at the U-node
v	quantity evaluated or interpolated at the V-node
uv	quantity evaluated or interpolated at the UV-node
w	quantity evaluated or interpolated at the W-node
uw	quantity evaluated or interpolated at the UW-node
vw	quantity evaluated or interpolated at the VW-node
n	quantity evaluated at the old baroclinic time t^n
$n+1$	quantity evaluated at the new baroclinic time t^{n+1}
m	quantity evaluated at the old barotropic time t^m
$m+1$	quantity evaluated at the new barotropic time t^{m+1}
it	quantity evaluated at the previous iteration level
$it+1$	quantity evaluated at the next iteration level
p	"predicted" value
f	"filtered" value

5.2.5 Time discretisation

The time discretisation of the model equations is summarised below. A detailed description is given in the sections below.

- In case a mode-splitting technique is used (Blumberg & Mellor, 1987), separate time steps are taken for the 2-D "external" barotropic equations ($\Delta\tau$) and the "internal" baroclinic equations (Δt). The 2-D time step $\Delta\tau$ has to be small enough to satisfy the Courant-Friedrichs-Lewy (CFL) criterion (see equation (5.4) below). The 3-D time step is a multiple, M_t, of $\Delta\tau$ (typically of the order of 10–20) and the model is integrated forward in time for N_t baroclinic time steps (equal to $N_t M_t = M_{tot}$ barotropic time steps). From stability analysis for linear surface gravity waves

$$\Delta\tau \leq \frac{\Delta h_{min}}{2\sqrt{g h_{max}}} \tag{5.4}$$

  and

$$\Delta t \leq \frac{\Delta h_{min}}{2\sqrt{g' h_{max}}} \tag{5.5}$$

  where $\Delta h_{min} = \min(h_1, h_2)$ is the minimum horizontal grid spacing, $g' = g\Delta\rho/\rho_0$ the reduced gravity, h_{max} the maximum water depth and $\Delta\rho$ a typical value for the vertical density difference. Since $g' \ll g$ the second condition is less constraining than the first one. A more stringent condition for the 3-D mode, imposed by the explicit schemes for horizontal advection, is that the horizontal distance travelled by a fluid element during the internal time step Δt, must be smaller than the grid spacing, or

$$\left(\frac{u\Delta t}{h_1}, \frac{v\Delta t}{h_2}\right) \leq 1 \tag{5.6}$$

- The semi-implicit hydrodynamic scheme only uses one (3-D) timestep. In this case, $M_t = 1$ and $\Delta\tau = \Delta t$. Because of the implicit treatment of the free surface wave, there is no need for the 2-D CFL time step restriction (5.4) for stability. The convective CFL criterion, eq. (5.6), still needs to be satisfied in all cells at all times.

- All horizontal derivatives are evaluated explicitly while vertical diffusion is computed fully implicitly and vertical advection quasi-implicitly.

- A predictor-corrector method is used to solve the horizontal momentum equations (4.61)–(4.62). This satisfies the requirement (Blumberg & Mellor, 1987) that, when using a mode-splitting technique, the currents

in the 3-D equations should have the same depth integral as the ones obtained from the 2-D depth-integrated equations.

- A quasi-implicit method is implemented for the Coriolis terms.
- Time integration is performed with the operator splitting method in conjunction with the TVD scheme for advection, whereas a simpler forward scheme is considered when advection is discretised with the upwind scheme.
- The sink terms in the momentum and turbulent transport equations, representing e.g. the bottom stress in the momentum equation or the dissipation rate ε and work against stable density gradients in e.g. the k-equation (4.204), are discretised quasi-implicitly to ensure positivity (Patankar, 1980). The sink terms in all other transport equations will be taken explicitly for reasons of conservation.
- The time step at which a quantity is evaluated in the discretised equations, is represented by one of the following superscripts (see also Table 5.2):
  - n: 3-D quantity at the old baroclinic time level $t^n = n\Delta t$
  - n+1: 3-D quantity at the new baroclinic time level $t^{n+1} = (n+1)\Delta t$
  - m: 2-D quantity at the old barotropic time level $t^m = n\Delta t + m\Delta\tau$
  - $m+1$: 2-D quantity at the new barotropic time level $t^{m+1} = n\Delta t + (m+1)\Delta\tau$
  - p: horizontal current at the "predicted" time step

  The superscript is omitted if no confusion is possible. If multiple superscripts appear separated by semicolons, the last superscript represents the spatial node, the one before last the time level. For example, $u^{n;c}$ denotes the value of u at time level n and node "C". In case of multiple subscripts separated by semicolons, the last one(s) is (are) the spatial index (indices).
- The time step notations are the same in the implicit case except that there are no intermediate barotropic time steps. However, there is now a possibility to perform the hydrodynamic solution more than once every time step. Particularly in case of the use of the semi-implicit free surface correction method, the accuracy can be enhanced by applying extra iterations. The values at these extra iteration levels are addressed with the following superscrips (see also Table 5.2):

- it: quantity at the previous iteration level
- $it + 1$: quantity at the present iteration level

5.3 Momentum equations

5.3.1 General procedure for the explicit case

The 3-D momentum equations are solved by a predictor-corrector method in which the sequence of operations for each baroclinic time step is as follows:

1. An initial (predictor) estimate of the currents u^p, v^p is calculated from the equations of three-dimensional motion.
2. An implicit correction is added to the predicted values for the Coriolis terms.
3. The 2-D depth-integrated equations of continuity and momentum are solved for ζ, U and V. This involves M_t integrations in time.
4. An implicit correction is added for the Coriolis terms at each barotropic time step.
5. The 3-D horizontal current u^p and v^p are corrected yielding u^{n+1} and v^{n+1} by adjusting u^p and v^p to ensure that the integrated currents obtained from the 2-D and 3-D momentum equations are identical.
6. The transformed and physical vertical current are obtained by solving (4.102) and (4.73).

Table 5.3: Parameters and variables used in the numerical description.Global and local FORTRAN names refer to the variables as defined on respectively the global and local (parallel) grid.

Symbol	Global name	Local name	Purpose
N_x	nc	ncloc	number of grid cells in the X-direction
N_y	nr	nrloc	number of grid cells in the Y-direction
N_z	nz	nz	number of grid cells in the vertical direction
h_1	—	delxatc	grid spacing in the X-direction at the cell centre
h_2	—	delyatc	grid spacing in the Y-direction at the cell centre

(*Continued*)

Table 5.3: *Continued*

h_3	—	—	grid spacing in the vertical direction at the cell centre (calculated as $H\Delta\sigma_k$)
$\Delta\sigma_k$	—	delsatc	grid spacing in the vertical σ-space at the cell centre
$\Delta\tau$	delt2d	delt2d	(barotropic or external) time step for the 2-D mode equations. In case of an implicit scheme $\Delta\tau = \Delta t$.
Δt	delt3d	—	(baroclinic or internal) time step used for the update of 3-D momentum (3-D mode) and all scalar quantities
M_t	ic3d	ic3d	number of 2-D (barotropic) time step within one 3-D (baroclinic) time step ($= \Delta t/\Delta\tau$). In case of an implicit scheme M_t=1.
N_{tot}	—	—	total number of 3-D time steps used in the simulation
M_{tot}	nstep	nstep	total number of 2-D time steps used in the simulation ($= M_t N_{tot}$)
θ_c	theta_cor	theta_cor	implicity factor for the Coriolis force with a value between 0 (explicit) and 1 (implicit). The default value, currently used in the program, is 0.5.
θ_a	theta_vadv	theta_vadv	implicity factor for vertical advection with a value between 0 (explicit) and 1 (implicit). The default value, currently used in the program, is 0.501.
θ_v	theta_vdif	theta_vdif	implicity factor for vertical diffusion with a value between 0 (explicit) and 1 (implicit). The default value, currently used in the program, is 1.
it_{max}	maxitsimp	maxitsimp	maximum allowed number of outer iterations for the implicit scheme
ϵ_{imp}	dzetaresid_conv	dzetaresid_conv	convergence limit for the free surface correction as used in (5.42)

(*Continued*)

Table 5.3: *Continued*

$\Omega(r)$	—	—	weight function between the upwind and Lax-Wendroff (central) fluxes used in the evaluation of the horizontal (vertical) advective fluxes. Its value depends on the value of the advective switches iopt_adv_3D (3-D currents), iopt_adv_2D (2-D currents), iopt_adv_scal (scalars) and iopt_adv_turb (turbulent variables) as given by (5.50)–(5.53).
u_f	—	ufvel	X-component of the "filtered" advective velocity, used for the advection of scalar quantities
v_f	—	vfvel	Y-component of the "filtered" advective velocity, used for the advection of scalar quantities
U_f	—	udfvel	value of the depth-integrated current U averaged over one baroclinic time step, as given by (5.22)
V_f	—	vdfvel	value of the depth-integrated current V averaged over one baroclinic time step, as given by (5.22)

5.3.1.1 predictor step

1. Firstly, the following terms are evaluated using values of currents, T, S at the old time step (t^n):

   - the density ρ from the equation of state (see Section 4.2.3) if iopt_dens>0
   - the coefficients of vertical diffusion if iopt_vdif_coef>0
   - the baroclinic pressure gradient (see Section 5.3.13) if iopt_dens_grad>0
   - the coeffficients of horizontal diffusion if iopt_hdif_coef=2 (Smagorinsky scheme, see Section 5.3.12.1)

2. The 3-D momentum equations (4.61) and (4.62) are integrated in time at each (internal) grid point (i,j,k). Their discretised forms without operator splitting, is given by

$$\frac{\tilde{u}^p - u^n}{\Delta t} = fv^n - \mathcal{A}_{h1}(u^n) - \mathcal{A}_{h2}(u^n) - \frac{v^{n;u}}{h_1^u h_2^u}(u^n \Delta_y^u h_1^{uv} - v^{n;u} \Delta_x^u h_2^c)$$

$$
\begin{aligned}
&-\theta_a \mathcal{A}_v(\tilde{u}^p) - (1-\theta_a)\mathcal{A}_v(u^n) + \theta_v \mathcal{D}_{mv}(\tilde{u}^p) + (1-\theta_v)\mathcal{D}_{mv}(u^n) \\
&-g\frac{\Delta_x^u \zeta^n}{h_1^u} - \frac{\Delta_x^u P_a}{\rho_0 h_1^u} + F_1^{b;n} + F_1^{t;n+1} + \mathcal{D}_{mh1}(\tau_{11}^n) + \mathcal{D}_{mh2}(\tau_{12}^n)
\end{aligned}
\tag{5.7}
$$

$$
\begin{aligned}
\frac{\tilde{v}^p - v^n}{\Delta t} &= -fu^n - \mathcal{A}_{h1}(v^n) - \mathcal{A}_{h2}(v^n) - \frac{u^{n;v}}{h_1^v h_2^v}(v^n \Delta_x^v h_2^{uv} - u^{n;v}\Delta_y^v h_1^c) \\
&-\theta_a \mathcal{A}_v(\tilde{v}^p) - (1-\theta_a)\mathcal{A}_v(v^n) + \theta_v \mathcal{D}_{mv}(\tilde{v}^p) + (1-\theta_v)\mathcal{D}_{mv}(v^n) \\
&-g\frac{\Delta_y^v \zeta^n}{h_2^v} - \frac{\Delta_y^v P_a}{\rho_0 h_2^v} + F_2^{b;n} + F_2^{t;n+1} + \mathcal{D}_{mh1}(\tau_{21}^n) + \mathcal{D}_{mh2}(\tau_{22}^n)
\end{aligned}
\tag{5.8}
$$

where $(\tilde{u}^p,\ \tilde{v}^p)$ are the "predicted" currents before implicit Coriolis correction, $f = 2\Omega \sin\phi$ is the Coriolis frequency, $\mathcal{A}_{hi}$, $\mathcal{A}_v$ are the horizontal and vertical advection operators defined by (4.64)–(4.66) and $\mathcal{D}_{mhi}$, $\mathcal{D}_{mv}$ the horizontal and vertical diffusion operators defined by (4.67)–(4.69).

3. The predictor currents are obtained by adding an implicit Coriolis correction:

$$
\begin{aligned}
u^p &= \tilde{u}^p \ + \ \frac{f\theta_c \Delta t(\Delta v^u - f\theta_c \Delta t \Delta u)}{1 + (f\theta_c \Delta t)^2} \\
v^p &= \tilde{v}^p \ - \ \frac{f\theta_c \Delta t(\Delta u^v + f\theta_c \Delta t \Delta v)}{1 + (f\theta_c \Delta t)^2}
\end{aligned}
\tag{5.9}
$$

where

$$
\Delta u = \tilde{u}^p - u^n\,, \quad \Delta v = \tilde{v}^p - v^n \tag{5.10}
$$

For details see Appendix C.

4. The "predicted" values for the depth-integrated current are obtained by integrating u^p and v^p over the vertical

$$
U^p = \sum_{k=1}^{N_z} u_k^p h_{3;k}^{n;u}\,, \quad V^p = \sum_{k=1}^{N_z} v_k^p h_{3;k}^{n;v} \tag{5.11}
$$

The following features are to be noted:

- The forward (Euler) scheme for time discretisation in (5.7)–(5.8) is replaced by the operator splitting method, discussed in Section 5.3.3.2, in case the TVD scheme is applied for the advective terms.

- By default the Coriolis terms are evaluated semi-implicitly (θ_c=0.5). The implicity factors for vertical advection θ_a and diffusion θ_v are set to respectively 0.5, 0.501 (semi-implicit) and 1 (fully implicit method). This is further discussed in Section 5.3.3.1.

- The equations are solved at the predictor step with application of surface and bottom boundary conditions, but without open boundary conditions.

5.3.1.2 depth-integrated equations

1. The depth-integrated baroclinic advective and diffusive terms (4.98)–(4.101) are updated using values of the baroclinic current at the old time level t^n.

2. The astronomical tidal force is updated at the new time level t^{n+1} (Section 5.3.14) if iopt_astro_tide=1.

3. The 2-D continuity equation (4.85) for the surface elevation ζ and the depth-integrated momentum equations (4.86)–(4.87) for U, V are solved at each (internal) grid point (i,j) for $M_t = \Delta t/\Delta\tau$ barotropic time steps

$$\frac{\zeta^{m+1}-\zeta^m}{\Delta\tau} = -\frac{1}{h_1 h_2}\Big(\Delta_x^c(h_2^u U^m) + \Delta_y^c(h_1^v V^m)\Big) \tag{5.12}$$

$$\begin{aligned}\frac{\tilde{U}^{m+1}-U^m}{\Delta\tau} + \frac{k_{b2}^u}{H^{m;u}}\tilde{U}^{m+1} &= fV^{m;u} - \overline{\mathcal{A}}_{h1}(U^m) - \overline{\mathcal{A}}_{h2}(U^m)\\ &-\frac{\overline{v}^{m;u}}{h_1^u h_2^u}(U^m\Delta_y^u h_1^{uv} - H^{m+1;u}\overline{v}^{m;u}\Delta_x^u h_2^c) - \frac{gH^{m+1;u}}{h_1^u}\Delta_x^u\zeta^{m+1}\\ &-\frac{H^{m+1;u}}{\rho_0 h_1^u}\Delta_x^u P_a + \overline{F}_1^{b;n} + H^{m+1;u}F_1^{t;m+1} + \tau_{s1}^u - k_{b1}^u(u_b^p - \frac{U^p}{H^{n;u}})\\ &+\overline{\mathcal{D}}_{mh1}(\overline{\tau_{11}})^m + \overline{\mathcal{D}}_{mh2}(\overline{\tau_{12}})^m - \overline{\delta A}_{h1}^n + \overline{\delta D}_{h1}^n\end{aligned} \tag{5.13}$$

$$\begin{aligned}\frac{\tilde{V}^{m+1}-V^m}{\Delta\tau} + \frac{k_{b2}^v}{H^{m;v}}\tilde{V}^{m+1} &= -fU^{m;v} - \overline{\mathcal{A}}_{h1}(V^m) - \overline{\mathcal{A}}_{h2}(V^m)\\ &-\frac{\overline{u}^{m;v}}{h_1^v h_2^v}(V^m\Delta_x^v h_2^{uv} - H^{m+1;v}\overline{u}^{m;v}\Delta_y^v h_1^c) - \frac{gH^{m+1;v}}{h_2^v}\Delta_y^v\zeta^{m+1}\\ &-\frac{H^{m+1;v}}{\rho_0 h_2^v}\Delta_y^v P_a + \overline{F}_2^{b;n} + H^{m+1;v}F_2^{t;m+1} + \tau_{s2}^v - k_{b1}^v(v_b^p - \frac{V^p}{H^{n;v}})\end{aligned}$$

$$+\overline{\mathcal{D}}_{mh1}(\overline{\tau_{21}})^m + \overline{\mathcal{D}}_{mh2}(\overline{\tau_{22}})^m - \overline{\delta A}^n_{h2} + \overline{\delta D}^n_{h2} \tag{5.14}$$

where

$$(\overline{u}, \overline{v}) = (U/H^u, V/H^v) \tag{5.15}$$

are the depth-mean currents and $\mathcal{A}_{hi}$, $\mathcal{D}_{mhi}$ are the 2-D advective and diffusion operators defined by (4.90)–(4.93).

A quasi-implicit formulation is used for the bottom stress in the U-equation of the form

$$\tau^u_{b1} = k^u_{b1}\Big(u^p_b - \frac{U^p}{H^{n;u}}\Big) + k^u_{b2}\frac{\tilde{U}^{m+1}}{H^{m;u}} \tag{5.16}$$

The friction velocities k_{b1} and k_{b2} depend on the formulation for the bottom stress (see equations (4.337)–(4.341)).

$$\begin{array}{lcl}
\text{no bottom stress} & : & k^u_{b1} = k^u_{b2} = 0 \\
\text{linear bottom stress} & : & k^u_{b1} = 0\,,\; k^u_{b2} = k_{lin} \\
\text{3-D quadratic law} & : & k^u_{b1} = k^u_{b2} = C^u_{db}\Big((u^n_b)^2 + (v^{n;u}_b)^2\Big)^{1/2} \\
\text{2-D quadratic law} & : & k^u_{b1} = 0\,,\; k^u_{b2} = C^u_{db}\Big((\overline{u}^m)^2 + (\overline{v}^{m;u})^2\Big)^{1/2}
\end{array} \tag{5.17}$$

The bottom drag coefficient C^u_{db} is calculated from (4.343), giving

$$C^u_{db;ij} = \kappa^2\Big[\ln\Big(\max(0.5h^u_{3;ij1}/z^u_{0;ij}), \xi_{min}\Big)\Big]^{-2} \tag{5.18}$$

or by (4.344)

$$C^u_{db;ij} = \kappa^2\Big[\ln\Big(\max(H^u_{ij}/(ez^u_{0;ij})), \xi_{min}\Big)\Big]^{-2} \tag{5.19}$$

or by interpolating an externally supplied C-node value at the U-node. Note that the discretisations guarantee that C_{db} remains finite, in case of a drying condition (i.e. when $z_r \to z_0$).

The bottom stress at the V-node is treated similarly.

4. An implicit correction is applied for the Coriolis terms:

$$\begin{aligned}
U^{m+1} &= \tilde{U}^{m+1} \;+\; \frac{f\theta_c\Delta\tau(\Delta V^u - f\theta_c\Delta\tau\Delta U)}{1 + (f\theta_c\Delta\tau)^2} \\
V^{m+1} &= \tilde{V}^{m+1} \;-\; \frac{f\theta_c\Delta\tau(\Delta U^v + f\theta_c\Delta\tau\Delta V)}{1 + (f\theta_c\Delta\tau)^2}
\end{aligned} \tag{5.20}$$

where

$$\Delta U = \tilde{U}^{m+1} - U^m\,, \quad \Delta V = \tilde{V}^{m+1} - V^m \tag{5.21}$$

For details see Appendix C.

5. The 2-D open boundary conditions are applied (see Section 5.3.16.1).

6. After solving (5.12)–(5.14) M_t times, the solutions are averaged over the baroclinic time step, giving

$$U_f = \frac{1}{M_t}\sum_{m=1}^{M_t} U^m\,, \quad V_f = \frac{1}{M_t}\sum_{m=1}^{M_t} V^m \tag{5.22}$$

where $M_t = \Delta t/\Delta\tau$ is the number of barotropic time steps.

5.3.1.3 corrector step

1. Open boundary conditions are applied for the baroclinic part

$$(\delta u, \delta v) = \left(u^{n+1} - \overline{u}^{n+1}, v^{n+1} - \overline{v}^{n+1}\right) \tag{5.23}$$

2. The predicted values u^p, v^p of the horizontal current are corrected to ensure that the depth-integrated currents obtained from the 2-D mode equations (5.13)–(5.14) are identical to the depth-integrated values of the 3-D current. The corrected values are then given by

$$u^{n+1} = \frac{H^{n;u}u^p + U^{n+1} - U^p}{H^{n+1;u}} \tag{5.24}$$

$$v^{n+1} = \frac{H^{n;v}v^p + V^{n+1} - V^p}{H^{n+1;v}} \tag{5.25}$$

3. The "filtered" advective velocities u_f and v_f, used for the advection of scalar quantities (see Section 5.5), are obtained by adding the depth-integrated current averaged over the baroclinic time step to the baroclinic part of the 3-D corrected current:

$$u_f^{n+1} = \frac{H^{n;u}u^p + U_f - U^p}{H^{n+1;u}} \tag{5.26}$$

$$v_f^{n+1} = \frac{H^{n;v}v^p + V_f - V^p}{H^{n+1;v}} \tag{5.27}$$

For details of the procedures see Ruddick (1995).

5.3.1.4 vertical current

The transformed vertical current ω is obtained by integrating the "baroclinic" continuity equation (4.102) from the bottom. Omitting the i- and j-indices this gives

$$\begin{aligned}
\omega_1^{n+1} &= 0 \\
\mathcal{F}_k &= \frac{1}{h_1^c h_2^c}\Big[\Delta_x^c\Big(h_2^u h_{3;k}^{n+1;u}(u_k^{n+1}-\overline{u}_k^{n+1})\Big)+\Delta_y^c\Big(h_1^v h_{3;k}^{n+1;v}(v_k^{n+1}-\overline{v}_k^{n+1})\Big)\Big] \\
&+ \frac{U^{n+1;c}}{h_1^c}\Delta_x^c\Big(\frac{h_{3;k}^{n+1;u}}{H^{n+1;u}}\Big)+\frac{V^{n+1;c}}{h_2^c}\Delta_y^c\Big(\frac{h_{3;k}^{n+1;v}}{H^{n+1;v}}\Big) \\
\omega_{k+1}^{n+1} &= \omega_k^{n+1}-\mathcal{F}_k \quad \text{for} \quad 2\le k\le N_z \\
\omega_{N_z+1}^{n+1} &= 0
\end{aligned} \tag{5.28}$$

The procedure guarantees that $\omega_{N_z+1}^{n+1}=0$.

The physical vertical current w is computed at the C-nodes from (4.73):

$$\begin{aligned}
w_k^{n+1} &= \frac{2(H^{n+1;c}z_k^{n+1;c}-H^{n;c}z_k^{n;c})}{\Delta t(H^{n;c}+H^{n+1;c})} \\
&+ \frac{1}{h_1^c h_2^c h_{3;k}^{n+1;c}}\Big[\Delta_x^c\Big(h_2^u h_{3;k}^{n+1;u}u_k^{n+1}z_k^{n+1;u}\Big)+\Delta_y^c\Big(h_1^v h_{3;k}^{n+1;v}v_k^{n+1}z_k^{n+1;v}\Big)\Big] \\
&+ \frac{\Delta_z^c(z_k^{n+1;w}\omega_k^{n+1})}{h_{3;k}^{n+1;c}}
\end{aligned} \tag{5.29}$$

where

$$z_k^{n+1;c}=H^{n+1;c}\sigma_k^c-h^c \tag{5.30}$$

and similar expressions at other nodes or time levels.

5.3.2 General procedure for the implicit case

With the implicit method, there is no longer need to solve the depth-integrated momentum equations (unless a 2-D grid has been selected). The stringent CFL stability criterium is relaxed by treating the terms that provoke the barotropic mode in an implicit manner. Difference with the previous explicit version is that the surface slope term is taken at the new time level. Horizontal advection and diffusion are calculated, as before, at the old time level.

After an explicit "predictor" step, velocities are corrected with the implicit free surface correction in the "corrector" step. In this method, the free

surface correction follows from the inversion of the elliptic free surface correction equation obtained from the 2-D continuity equation. Because of the non-linear dependency of the equations on the free surface height through the h_3-term, an iterative scheme has been implemented in addition.

1. At the first iteration $\zeta^{n+1,1} = \zeta^n$ and $h_3^{n+1,1} = (h + \zeta^n)\Delta\sigma$.

2. The momentum equations are solved at the predictor step using the latest values for h_3 and ζ:

$$\begin{aligned}\frac{h_3^{n+1,it}\tilde{u}^p - h_3^n u^n}{h_3^n \Delta t} &= fv^n - \mathcal{A}_{h1}(u^n) - \mathcal{A}_{h2}(u^n) \\ &-\frac{v^{n;u}}{h_1^u h_2^u}(u^n \Delta_y^u h_1^{uv} - v^{n;u}\Delta_x^u h_2^c) - \theta_a \mathcal{A}_v(\tilde{u}^p) - (1-\theta_a)\mathcal{A}_v(u^n) \\ &+\theta_v \mathcal{D}_{mv}(\tilde{u}^p) + (1-\theta_v)\mathcal{D}_{mv}(u^n) - g\frac{h_3^{n+1,it}}{h_3^n}\frac{\Delta_x^u \zeta^{n+1,it}}{h_1^u} \\ &-\frac{\Delta_x^u P_a}{\rho_0 h_1^u} + F_1^{b;n} + F_1^{t;n+1} + \mathcal{D}_{mh1}(\tau_{11}^n) + \mathcal{D}_{mh2}(\tau_{12}^n) \end{aligned} \tag{5.31}$$

$$\begin{aligned}\frac{1}{h_3^n}\frac{h_3^{n+1,it}\tilde{v}^p - h_3^n v^n}{\Delta t} &= -fu^n - \mathcal{A}_{h1}(v^n) - \mathcal{A}_{h2}(v^n) \\ &-\frac{u^{n;v}}{h_1^v h_2^v}(v^n \Delta_x^v h_2^{uv} - u^{n;v}\Delta_y^v h_1^c) - \theta_a \mathcal{A}_v(\tilde{v}^p) - (1-\theta_a)\mathcal{A}_v(v^n) \\ &+\theta_v \mathcal{D}_{mv}(\tilde{v}^p) + (1-\theta_v)\mathcal{D}_{mv}(v^n) - g\frac{h_3^{n+1,it}}{h_3^n}\frac{\Delta_y^v \zeta^{n+1,it}}{h_2^v} \\ &-\frac{\Delta_y^v P_a}{\rho_0 h_2^v} + F_2^{b;n} + F_2^{t;n+1} + \mathcal{D}_{mh1}(\tau_{21}^n) + \mathcal{D}_{mh2}(\tau_{22}^n) \end{aligned} \tag{5.32}$$

   where the surface slope is taken at the previous iteration level. The predicted currents (u^p, v^p) are obtained from $(\tilde{u}^p, \tilde{v}^p)$ after applying the implicit correction for the Coriolis terms, given by (5.9)–(5.10).

3. The free surface correction ζ' is defined as

$$\zeta' = \zeta^{n+1,it+1} - \zeta^{n+1,it} \tag{5.33}$$

   The corrected depth-integrated current is then obtained by adding an implicit correction term

$$U^{n+1,it+1} \;=\; U^p - H^{n+1,it;u}\frac{\Delta t g}{h_1}\frac{\partial \zeta'}{\partial \xi_1} \tag{5.34}$$

$$V^{n+1,it+1} = V^p - H^{n+1,it;v}\frac{\Delta t g}{h_2}\frac{\partial \zeta'}{\partial \xi_2} \tag{5.35}$$

where (U^p,V^p) are the depth integrated values of (u^p,v^p).

The values for ζ' follow from inversion of the elliptic equation that arises by introducing (5.34)–(5.35) into the 2-D continuity equation

$$\frac{\zeta^{n+1,it}-\zeta^n}{\Delta t}+\frac{\zeta'}{\Delta t} = -\frac{1}{h_1 h_2}\left(\Delta_x^c\left(h_2^u U^p\right)+\Delta_y^c\left(h_1^v V^p\right)\right)$$
$$+\frac{1}{h_1 h_2}\left[\Delta_x^c\left(\frac{\Delta t h_2^u g^u H^{n+1,it;u}}{h_1^u}\Delta_x^u\zeta'\right)+\Delta_y^c\left(\frac{\Delta t h_1^v g^v H^{n+1,it;v}}{h_2^v}\Delta_y^v\zeta'\right)\right] \tag{5.36}$$

Equation (5.36) can be written as a linear system of equations with non-zero values only on the diagonal and five sub-diagonals

$$A_{ij}\zeta'_{i-1,j}+B_{i,j}\zeta'_{i,j-1}+C_{ij}\zeta'_{i,j}+D_{ij}\zeta'_{i,j+1}+E_{ij}\zeta'_{i+1,j}=F_{ij} \tag{5.37}$$

Since the decomposition (5.34)–(5.35) can no longer be used at open boundaries, U^{n+1} or V^{n+1} are firstly written as a sum of explicit and implicit (involving ζ') terms which are then substituted into the continuity equation. This is further discussed in Section 5.3.19.1.

4. The free surface elevation is updated

$$\zeta^{n+1,it+1} = \zeta^{n+1,it}+\zeta' \tag{5.38}$$

5. The total water depth is updated

$$H^{n+1,it+1} = H^{n+1,it}+\zeta' \tag{5.39}$$

6. The depth-integrated velocity fields are corrected using (5.34)–(5.35).

7. The values of $U^{n+1,it+1}$ and $V^{n+1,it+1}$ are evaluated at the open boundaries by applying the appropriate boundary conditions.

8. The predicted values u^p, v^p of the horizontal current are corrected to ensure that the depth-integrated currents obtained from equations (5.34)–(5.35) are identical to the depth-integrated values of the 3-D current. The corrected values are then given by

$$u^{n+1} = \frac{H^{n+1,it;u}u^p + U^{n+1,it+1} - U^p}{H^{n+1,it+1;u}} \tag{5.40}$$

$$v^{n+1} = \frac{H^{n+1,it;v}v^p + V^{n+1,it+1} - V^p}{H^{n+1,it+1;v}} \tag{5.41}$$

9. A convergence check is performed by comparing the norm of ζ' with a threshold value ϵ, i.e.

$$\|\zeta'\|_\infty = \max(\zeta') \leq \epsilon_{imp} \tag{5.42}$$

A new iteration is started when the criterion is not satisfied, unless $it > it_{max}$ in which case no further iterations are taken.

10. After completing the iteration loop, the vertical current is obtained by integration of the "baroclinic" continuity equation, as described in Section 5.3.1.4. Since there are no barotropic time steps, one has $u_f = u^{n+1}$, $v_f = v^{n+1}$.

At present, no algorithm has been programmed within the COHERENS source code to solve the linear system, arising from the discretisation of the 2-D continuity equation. Routines have, however, been provided to solve (5.37) with the external PETSc library which is activated in the program by setting the -DPETSC compiler option. Different algorithms (linear solvers and preconditioners) are available, whose default values (Incomplete Cholesky preconditioner in combination with a GMRES solver) can be changed by the user. Since the solvers are iterative, a tolerance level has to be provided.

In summary, application of the implicit scheme involves two iteration loops. The inner loop solves the linear system for ζ' and is controlled by the routines of the PETSc library. The maximum number of iterations of the outer loop (needed for convergence of the h_3-factor) is set by the user with the parameter maxitsimp.

5.3.3 Advection schemes and time discretisation

5.3.3.1 introduction

The time discretisation of the momentum equations depends on the type of advection scheme employed for the spatial discretisation of the horizontal and vertical advection terms. Several schemes are implemented in the program, selected with the model switches iopt_adv_3D and iopt_adv_2D. They may take the following values:

0 : horizontal and vertical advection of momentum disabled

1 : upwind scheme for horizontal and vertical advection

2 : Lax-Wendroff scheme for horizontal, central scheme for vertical advection[1]

3 : TVD (Total Variation Diminishing) scheme using the superbee limiter as a weighting function between the upwind scheme and either the Lax-Wendroff scheme in the horizontal or the central scheme in the vertical

4 : as the previous case now using the monotonic limiter.

The discretisation of the different advection schemes is illustrated with the following simple example, describing the 1-D advection of a scalar ψ:

$$\frac{\partial \psi}{\partial t} + a\frac{\partial \psi}{\partial x} = 0 \tag{5.43}$$

where a is a constant advecting velocity and the equation is spatially integrated for the interval $x_a \leq x \leq x_b$. The equation can then be rewritten in flux form

$$\frac{\partial \psi}{\partial t} + \frac{\partial F}{\partial x} = 0 \tag{5.44}$$

where $F = a\psi$ is the advective flux. The discretised form of (5.44), using forward Euler time integration, is given by

$$\frac{\psi^{n+1} - \psi^n}{\Delta t} + \frac{F_{i+1} - F_i}{\Delta x} = 0 \tag{5.45}$$

where Δt is the time step and Δx a uniform grid spacing. The quantities ψ and F_i are evaluated on a uniform staggered grid (see Figure 5.5) with ψ-points located halfway between the F-points. Boundary conditions at x_a and x_b are needed to determine the fluxes F_1 and F_{N+1}. At interior points, i.e. for $2 \leq i \leq N+1$, the fluxes F_i are then written as a weighting between the upwind and Lax-Wendroff fluxes $F_{up;i}$ and $F_{lw;i}$:

$$F_i = \Big((1 - \Omega(r_i))F_{up;i} + \Omega(r_i)F_{lw;i}\Big) \tag{5.46}$$

where

$$F_{up;i} = \frac{1}{2}a\Big((1 + s_i)\psi_{i-1} + (1 - s_i)\psi_i\Big) \tag{5.47}$$

$$F_{lw;i} = \frac{1}{2}a\Big((1 + c_i)\psi_{i-1} + (1 - c_i)\psi_i\Big) \tag{5.48}$$

[1]The "pure" Lax-Wendroff and central schemes have only been implemented for illustrative purposes and should be avoided in realistic simulations.

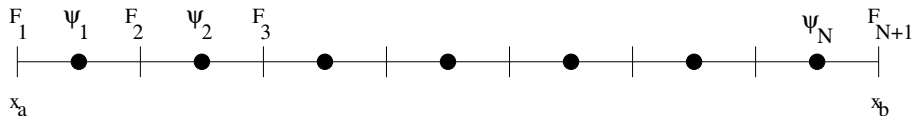

Figure 5.5: Numerical grid for the 1-D advection problem.

where s_i and c_i are the sign and CFL number of the advecting current

$$s_i = \mathrm{Sign}(a)\,, \quad c_i = \frac{a\Delta t}{\Delta x} \tag{5.49}$$

The weight function Ω depends on the type of advection scheme:

- upwind
$$\Omega(r) = 0 \tag{5.50}$$
- Lax-Wendroff
$$\Omega(r) = 1 \tag{5.51}$$
- TVD with superbee limiter
$$\Omega(r) = \max(0, \min(2r, 1), \min(r, 2)) \tag{5.52}$$
- TVD with monotonic limiter
$$\Omega(r) = \frac{r + |r|}{1 + |r|} \tag{5.53}$$

The argument r of Ω is defined by

$$\begin{aligned} r_i &= \frac{(1 + s_i)\Delta F_{i-1} + (1 - s_i)\Delta F_{i+1}}{2\Delta F_i} \\ \Delta F_i &= F_{lw;i} - F_{up;i} \end{aligned} \tag{5.54}$$

The discretisation scheme for vertical advection is similar, except that the Lax-Wendroff flux $F_{lw;k}$ is replaced by the central flux

$$F_{ce;k} = \frac{1}{2} a(\psi_{k-1} + \psi_k) \tag{5.55}$$

The discretisation schemes, actually applied in the model, need to take account of the following additional complexities

- non-uniform grids

- space and time dependent currents
- grid staggering (advected quantities and advecting currents can, for example, be located at the same locations)
- extensions to 2-D and 3-D grids
- time integration using operator splitting (see below) to improve the time accuracy of the TVD scheme

Explicit expressions of each discretisation will be presented below for the advective terms of all model equations.

The upwind scheme has the interesting property to preserve monotonicity, but has the disadvantage of being only first order accurate. The Lax-Wendroff scheme, on the other hand, is accurate to second order in space and time but non-monotone which means that spurious over- and undershootings are created in regimes of strong gradients. This is clearly illustrated by the results of the test cases ***cones*** and ***front*** described in Sections 23.1 and 23.2. The TVD scheme has the advantage of combining the monotonicity of the upwind scheme with the second order accuracy of the Lax-Wendroff scheme.

Horizontal advection is evaluated explicitly to prevent the solution of large-banded matrix systems. A necessary stability condition for both the upwind and the Lax-Wendroff scheme is given by the criterion (5.6) (see Hirsch, 1990). The restriction to explicit schemes does not apply for the vertical since the discretised equations can be written into a simpler tridiagonal form (see Section 5.3.18). A semi-implicit scheme in the vertical allows to replace the Lax-Wendroff by the central scheme which is a monotone scheme and stable provided that the implicity factor $\theta_a \geq 0.5$.

The aim of the limiter function is to reduce the numerical diffusion due to the upwind scheme in areas of low gradients and to provide sufficiently large diffusion in regions of large gradients so that over- and undershooting due to the non-monotonicity of the Lax-Wendroff scheme are suppressed. Both the superbee (Roe, 1985) as the monotonic limiter are available in the program. The ***cones*** and ***front*** test case simulations (see Sections 23.1 and 23.2) showed that the superbee limiter is the least diffusive and is therefore taken as the default formulation in the program. The spatial discretisation of the advective terms in the momentum equations and the form of the limiter function are further discussed in the subsections below.

In the absence of advection or when the upwind or Lax-Wendroff/central scheme is selected, the momentum equations are solved by forward time-stepping as given by the time-discretised forms (5.7)–(5.8) or (5.31)–(5.32). In case of the TVD scheme, the spatial discretisation of the advective terms

involves the Lax-Wendroff and central schemes which are both second order accurate in space. The equations are then integrated in time with the aid of the "fractional step" or "operator splitting" method as proposed by Yanenko (1971). The procedure consists in splitting the time integration into three fractional steps. During the first and second step only the advection-diffusion terms in respectively the X- and Y-direction are taken into account. The vertical advection and diffusion terms and all other terms (Coriolis force, pressure gradient and tidal force) are included during the third time step. To preserve the second-order accuracy of the 1-D schemes in the fractional step approach the method of symmetric splitting (e.g. Hirsch, 1990) is implemented. This means that the previous procedure ("A"-steps) is repeated now in reverse order ("B"-steps) , i.e. vertical advection/diffusion and other terms followed by advection-diffusion in the Y-direction, followed by advection-diffusion in the X-direction. The final "predicted" value of u^p or v^p is then obtained by taking the average of the values at the end of the A- and B-steps. The same method is applied for scalar quantities.

The implicity factors θ_a and θ_v have a range between 0 and 1 where a 0 corresponds to a fully explicit, 1 a fully implicit and 0.5 a semi-implicit (Crank-Nicholson) method. The schemes are stable provided that $\theta_a, \theta_v \geq 0.5$. To retain the same accuracy in time for horizontal as well as vertical advection the defaults are a semi-implicit option for vertical advection, i.e. $\theta_a = 0.501$[2] and a fully implicit treatment of vertical diffusion ($\theta_v = 1$). Contrary to COHERENS V1, these defaults can be changed by the user and can take any value between 0 and 1.

For a more detailed account of advection schemes and the time splitting method see Ruddick (1995).

[2]The central scheme is second accurate in time if $\theta_a = 0.5$.

Table 5.4: Overview of the operators used in the numerical discretisations.

Type	Purpose
difference operators	
Δ_x	difference operator in the X-direction
Δ_y	difference operator in the Y-direction
Δ_z	difference operator in the vertical direction
advective operators	
$\mathcal{A}_{h1}$	horizontal advection in the X-direction[3] $\mathcal{A}_{h1}(F) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_1}(h_2 h_3 u_f F)$
$\mathcal{A}_{h2}$	horizontal advection in the Y-direction[3] $\mathcal{A}_{h2}(F) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_2}(h_1 h_3 v_f F)$
$\mathcal{A}_v$	vertical advection (u, v and scalars) $\mathcal{A}_v(F) = \frac{1}{h_3}\frac{\partial}{\partial s}(\omega F)$
$\overline{\mathcal{A}}_{h1}$	horizontal advection in the X-direction (2-D mode) $\overline{\mathcal{A}}_{h1}(F) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_1}\left(\frac{h_2 U F}{H}\right)$
$\overline{\mathcal{A}}_{h2}$	horizontal advection in the Y-direction (2-D mode) $\overline{\mathcal{A}}_{h2}(F) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_2}\left(\frac{h_1 V F}{H}\right)$
extended advective operators for currents including curvature terms	

(*Continued*)

[3]Note that (u_f,v_f) is replaced by (u,v) if F represents u, v or a turbulent transport variable (k,ε,kl).

Table 5.4: *Continued*

$\tilde{\mathcal{A}}_{h1}(u)$	extended horizontal advection of u in the X-direction $$\tilde{\mathcal{A}}_{h1}(u) = \mathcal{A}_{h1}(u) - \frac{v^2}{h_1 h_2}\frac{\partial h_2}{\partial \xi_1}$$
$\tilde{\mathcal{A}}_{h2}(u)$	extended horizontal advection of u in the Y-direction $$\tilde{\mathcal{A}}_{h2}(u) = \mathcal{A}_{h2}(u) + \frac{uv}{h_1 h_2}\frac{\partial h_1}{\partial \xi_2}$$
$\tilde{\mathcal{A}}_{h1}(v)$	extended horizontal advection of v in the X-direction $$\tilde{\mathcal{A}}_{h1}(v) = \mathcal{A}_{h1}(v) + \frac{uv}{h_1 h_2}\frac{\partial h_2}{\partial \xi_1}$$
$\tilde{\mathcal{A}}_{h2}(v)$	extended horizontal advection of v in the Y-direction $$\tilde{\mathcal{A}}_{h2}(v) = \mathcal{A}_{h2}(v) - \frac{u^2}{h_1 h_2}\frac{\partial h_1}{\partial \xi_2}$$
$\overline{\tilde{\mathcal{A}}}_{h1}(U)$	extended horizontal advection of U in the X-direction $$\overline{\tilde{\mathcal{A}}}_{h1}(U) = \overline{\mathcal{A}}_{h1}(U) - \frac{\overline{v}V}{h_1 h_2}\frac{\partial h_2}{\partial \xi_1}$$
$\overline{\tilde{\mathcal{A}}}_{h2}(U)$	extended horizontal advection of U in the Y-direction $$\overline{\tilde{\mathcal{A}}}_{h2}(U) = \overline{\mathcal{A}}_{h2}(U) + \frac{\overline{v}U}{h_1 h_2}\frac{\partial h_1}{\partial \xi_2}$$
$\overline{\tilde{\mathcal{A}}}_{h1}(V)$	extended horizontal advection of V in the X-direction $$\overline{\tilde{\mathcal{A}}}_{h1}(V) = \overline{\mathcal{A}}_{h1}(V) + \frac{\overline{u}V}{h_1 h_2}\frac{\partial h_2}{\partial \xi_1}$$
$\overline{\tilde{\mathcal{A}}}_{h2}(V)$	extended horizontal advection of V in the Y-direction $$\overline{\tilde{\mathcal{A}}}_{h2}(V) = \overline{\mathcal{A}}_{h2}(V) - \frac{\overline{u}U}{h_1 h_2}\frac{\partial h_1}{\partial \xi_2}$$
diffusion operators	

(*Continued*)

Table 5.4: *Continued*

$\mathcal{D}_{sh1}$	horizontal diffusion in the X-direction (scalars) $$\mathcal{D}_{sh1}(\psi) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_1}\Big(\lambda_H \frac{h_2 h_3}{h_1}\frac{\partial \psi}{\partial \xi_1}\Big)$$
$\mathcal{D}_{sh2}$	horizontal diffusion in the Y-direction (scalars) $$\mathcal{D}_{sh2}(\psi) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(\lambda_H \frac{h_1 h_3}{h_2}\frac{\partial \psi}{\partial \xi_2}\Big)$$
$\mathcal{D}_{mh1}$	horizontal diffusion in the X-direction (3-D momentum) $$\mathcal{D}_{mh1}(F) = \frac{1}{h_1 h_2^2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2^2 h_3 F\Big)$$
$\mathcal{D}_{mh2}$	horizontal diffusion in the Y-direction (3-D momentum) $$\mathcal{D}_{mh2}(F) = \frac{1}{h_1^2 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1^2 h_3 F\Big)$$
τ_{ij}	3-D horizontal shear stress tensor $$\begin{aligned} \tau_{11} &= -\tau_{22} = \nu_H D_T \\ \tau_{12} &= \tau_{21} = \nu_H D_S \\ D_T &= \frac{h_2}{h_1}\frac{\partial}{\partial \xi_1}\left(\frac{u}{h_2}\right) - \frac{h_1}{h_2}\frac{\partial}{\partial \xi_2}\left(\frac{v}{h_1}\right) \\ D_S &= \frac{h_1}{h_2}\frac{\partial}{\partial \xi_2}\left(\frac{u}{h_1}\right) + \frac{h_2}{h_1}\frac{\partial}{\partial \xi_1}\left(\frac{v}{h_2}\right) \end{aligned}$$
$\mathcal{D}_{sv}$	vertical diffusion (scalars) $$\mathcal{D}_{sv}(F) = \frac{1}{h_3}\frac{\partial}{\partial s}\Big(\frac{\lambda_T^{\psi}}{h_3}\frac{\partial F}{\partial s}\Big)$$
$\mathcal{D}_{mv}$	vertical diffusion (momentum) $$\mathcal{D}_{mv}(F) = \frac{1}{h_3}\frac{\partial}{\partial s}\Big(\frac{\nu_T}{h_3}\frac{\partial F}{\partial s}\Big)$$
$\overline{\mathcal{D}}_{mh1}$	horizontal diffusion in the X-direction (2-D momentum) $$\overline{\mathcal{D}}_{mh1}(F) = \frac{1}{h_1 h_2^2}\frac{\partial}{\partial \xi_1}\Big(h_2^2 F\Big)$$
$\overline{\mathcal{D}}_{mh2}$	horizontal diffusion in the Y-direction (2-D momentum)

(*Continued*)

Table 5.4: *Continued*

	$$\overline{\mathcal{D}}_{mh2}(F) = \frac{1}{h_1^2 h_2}\frac{\partial}{\partial \xi_2}\left(h_1^2 F\right)$$
$\overline{\tau_{ij}}$	2-D horizontal shear stress tensor $$\begin{aligned}\overline{\tau_{11}} &= -\overline{\tau_{22}} = \overline{\nu_H}\overline{D}_T \\ \overline{\tau_{12}} &= \overline{\tau_{21}} = \overline{\nu_H}\overline{D}_S \\ \overline{D}_T &= \frac{h_2}{h_1}\frac{\partial}{\partial \xi_1}\left(\frac{\overline{u}}{h_2}\right) - \frac{h_1}{h_2}\frac{\partial}{\partial \xi_2}\left(\frac{\overline{v}}{h_1}\right) \\ \overline{D}_S &= \frac{h_1}{h_2}\frac{\partial}{\partial \xi_2}\left(\frac{\overline{u}}{h_1}\right) + \frac{h_2}{h_1}\frac{\partial}{\partial \xi_1}\left(\frac{\overline{v}}{h_2}\right)\end{aligned}$$
other operators	
$\mathcal{P}$	production terms in the scalar transport equations
$\mathcal{S}$	sink terms in the scalar transport equations
$\mathcal{T}$	production minus sink terms in the scalar transport equations $$\mathcal{T} = \mathcal{P} - \mathcal{S}$$
$\mathcal{C}_{s1}^f$	X-corrector term in the scalar transport equations $$\mathcal{C}_{s1}^f(\psi) = \frac{\psi}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_1}(h_2 h_3 u_f)$$
$\mathcal{C}_{s2}^f$	Y-corrector term in the scalar transport equations $$\mathcal{C}_{s2}^f(\psi) = \frac{\psi}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_2}(h_1 h_3 v_f)$$
$\mathcal{C}_{s3}$	Z-corrector term in the scalar transport equations $$\mathcal{C}_{s3}(\psi) = \frac{\psi}{h_3}\frac{\partial \omega}{\partial s}$$

5.3.3.2 mode splitting scheme for the 3-D momentum equations

The time-discretised form of the u-equation (4.61) with mode splitting is given by

- Part A

$$\frac{u_A^{n+1/3}-u^n}{\Delta t} = -\mathcal{A}_{h1}(u^n) + \frac{(v^{n;u})^2\Delta_x^u h_2^c}{h_1^u h_2^u} + \mathcal{D}_{mh1}(\nu_H D_T(u^n, v^n)) \quad (5.56)$$

$$\begin{aligned}\frac{u_A^{n+2/3}-u_A^{n+1/3}}{\Delta t} &= -\mathcal{A}_{h2}(u_A^{n+1/3}) - \frac{u_A^{n+1/3}v^{n;u}\Delta_y^u h_1^{uv}}{h_1^u h_2^u} \\ &\quad + \mathcal{D}_{mh2}(\nu_H D_S(u_A^{n+1/3}, v^n)) \end{aligned} \quad (5.57)$$

$$\begin{aligned}\frac{\tilde{u}_A^p - u_A^{n+2/3}}{\Delta t} &= -\theta_a\mathcal{A}_v(\tilde{u}_A^p) - (1-\theta_a)\mathcal{A}_v(u_A^{n+2/3}) \\ &\quad +\theta_v\mathcal{D}_{mv}(\tilde{u}_A^p) + (1-\theta_v)\mathcal{D}_{mv}(u_A^{n+2/3}) + \mathcal{O}_1 \end{aligned} \quad (5.58)$$

- Part B

$$\begin{aligned}\frac{u_B^{n+1/3}-u^n}{\Delta t} &= -\theta_a\mathcal{A}_v(u_B^{n+1/3}) - (1-\theta_a)\mathcal{A}_v(u^n) \\ &\quad +\theta_v\mathcal{D}_{mv}(u_B^{n+1/3}) + (1-\theta_v)\mathcal{D}_{mv}(u^n) + \mathcal{O}_1 \end{aligned} \quad (5.59)$$

$$\begin{aligned}\frac{u_B^{n+2/3}-u_B^{n+1/3}}{\Delta t} &= -\mathcal{A}_{h2}(u_B^{n+1/3}) - \frac{u_B^{n+1/3}v^{n;u}\Delta_y^u h_1^{uv}}{h_1^u h_2^u} \\ &\quad + \mathcal{D}_{mh2}(\nu_H D_S(u_B^{n+1/3}, v^n)) \end{aligned} \quad (5.60)$$

$$\frac{\tilde{u}_B^p - u_B^{n+2/3}}{\Delta t} = -\mathcal{A}_{h1}(u_B^{n+2/3}) + \frac{(v^{n;u})^2\Delta_x^u h_2^c}{h_1^u h_2^u} + \mathcal{D}_{mh1}(\nu_H D_T(u_B^{n+2/3}, v^n)) \quad (5.61)$$

- Predictor value

$$\tilde{u}^p = \frac{1}{2}(\tilde{u}_A^p + \tilde{u}_B^p) \quad (5.62)$$

The $\mathcal{O}_1$-terms are defined by

$$\mathcal{O}_1 = fv^{n;u} - g\frac{\Delta_x^u\zeta^n}{h_1^u} - \frac{\Delta_x^u P_a}{\rho_0 h_1^u} + F_1^{b;n} + F_1^{t;n+1} \quad (5.63)$$

A similar procedure is applied for the v-equation (4.62).

- Part A

$$\frac{v_A^{n+1/3} - v^n}{\Delta t} = -\mathcal{A}_{h1}(v^n) - \frac{u^{n;v} v^n \Delta_x^v h_2^{uv}}{h_1^v h_2^v} + \mathcal{D}_{mh1}(\nu_H D_S(u^n, v^n)) \quad (5.64)$$

$$\begin{aligned} \frac{v_A^{n+2/3} - v_A^{n+1/3}}{\Delta t} &= -\mathcal{A}_{h2}(v_A^{n+1/3}) + \frac{(u^{n;v})^2 \Delta_y^v h_1^c}{h_1^v h_2^v} \\ &\quad - \mathcal{D}_{mh2}(\nu_H D_T(u^n, v_A^{n+1/3})) \end{aligned} \quad (5.65)$$

$$\begin{aligned} \frac{\tilde{v}_A^p - v_A^{n+2/3}}{\Delta t} &= -\theta_a \mathcal{A}_v(\tilde{v}_A^p) - (1-\theta_a)\mathcal{A}_v(v_A^{n+2/3}) \\ &\quad + \theta_v \mathcal{D}_{mv}(\tilde{v}_A^p) + (1-\theta_v)\mathcal{D}_{mv}(v_A^{n+2/3}) + \mathcal{O}_2 \end{aligned} \quad (5.66)$$

- Part B

$$\begin{aligned} \frac{v_B^{n+1/3} - v^n}{\Delta t} &= -\theta_a \mathcal{A}_v(v_B^{n+1/3}) - (1-\theta_a)\mathcal{A}_v(v^n) \\ &\quad + \theta_v \mathcal{D}_{mv}(v_B^{n+1/3}) + (1-\theta_v)\mathcal{D}_{mv}(v^n) + \mathcal{O}_2 \end{aligned} \quad (5.67)$$

$$\begin{aligned} \frac{v_B^{n+2/3} - v_B^{n+1/3}}{\Delta t} &= -\mathcal{A}_{h2}(v_B^{n+1/3}) + \frac{(u^{n;v})^2 \Delta_y^v h_1^c}{h_1^v h_2^v} \\ &\quad - \mathcal{D}_{mh2}(\nu_H D_T(u^n, v_B^{n+1/3})) \end{aligned} \quad (5.68)$$

$$\begin{aligned} \frac{\tilde{v}_B^p - v_B^{n+2/3}}{\Delta t} &= -\mathcal{A}_{h1}(v_B^{n+2/3}) - \frac{u^{n;v} v_B^{n+2/3} \Delta_x^v h_2^{uv}}{h_1^v h_2^v} \\ &\quad + \mathcal{D}_{mh1}(\nu_H D_S(u^n, v_B^{n+2/3})) \end{aligned} \quad (5.69)$$

- Predictor value

$$\tilde{v}^p = \frac{1}{2}(\tilde{v}_A^p + \tilde{v}_B^p) \quad (5.70)$$

The $\mathcal{O}_2$-terms are defined by

$$\mathcal{O}_2 = -f u^{n;v} - g\frac{\Delta_y^v \zeta^n}{h_2^v} - \frac{\Delta_y^v P_a}{\rho_0 h_2^v} + F_2^{b;n} + F_2^{t;n+1} \quad (5.71)$$

Once $\tilde{u}^p$ and $\tilde{v}^p$ are obtained, an implicit correction is applied as described in Section 5.3.1.1.

Important to note again is that, compared to the simpler forward scheme, the computation using symmetrical operator splitting increases the CPU time for the circulation module by a factor two, but has the advantage of being more accurate which is an important property in regions of strong horizontal and vertical shear.

5.3.3.3 mode splitting scheme for the 2-D momentum equations

The operator splitting is applied for the 2-D case if the advective terms are discretised with the TVD scheme (iopt_adv_2D=3). Since the 2-D mode equations are solved with a much smaller time step than the 3-D mode, second-order accuracy is of less relevance. Contrary to the 3-D case, the simpler upwind scheme, using only a forward Euler time integration, can be recommended for 2-D applications.

The method is analogous to the 3-D case, but given here in detail for completeness. Firstly, the U-equation (4.86) is solved as follows:

- Part A

$$\begin{aligned}\frac{U_A^{m+1/3}-U^m}{\Delta\tau} &= -\overline{\mathcal{A}}_{h1}(U^m)+\frac{H^{m;u}(\overline{v}^{m;u})^2\Delta_x^u h_2^c}{h_1^u h_2^u}\\ &\quad+\overline{\mathcal{D}}_{mh1}(\overline{\nu_H}\overline{D}_T(U^m,V^m))\end{aligned} \tag{5.72}$$

$$\begin{aligned}\frac{U_A^{m+2/3}-U_A^{m+1/3}}{\Delta\tau} &= -\overline{\mathcal{A}}_{h2}(U_A^{m+1/3})-\frac{U_A^{m+1/3}\overline{v}^{m;u}\Delta_y^u h_1^{uv}}{h_1^u h_2^u}\\ &\quad+\overline{\mathcal{D}}_{mh2}(\overline{\nu_H}\overline{D}_S(U_A^{m+1/3},V^m))\end{aligned} \tag{5.73}$$

$$\frac{\tilde{U}_A^{m+1}-U_A^{m+2/3}}{\Delta\tau}+\frac{k_{b2}^u}{H^{m;u}}\tilde{U}_A^{m+1}=\overline{\mathcal{O}}_1 \tag{5.74}$$

- Part B

$$\frac{U_B^{m+1/3}-U^m}{\Delta\tau}+\frac{k_{b2}^u}{H^{m;u}}U_B^{m+1/3}=\overline{\mathcal{O}}_1 \tag{5.75}$$

$$\begin{aligned}\frac{U_B^{m+2/3}-U_B^{m+1/3}}{\Delta\tau} &= -\overline{\mathcal{A}}_{h2}(U_B^{m+1/3})-\frac{U_B^{m+1/3}\overline{v}^{m;u}\Delta_y^u h_1^{uv}}{h_1^u h_2^u}\\ &\quad+\overline{\mathcal{D}}_{mh2}(\overline{\nu_H}\overline{D}_S(U_B^{m+1/3},V^m))\end{aligned} \tag{5.76}$$

$$\begin{aligned}\frac{\tilde{U}_B^{m+1}-U_B^{m+2/3}}{\Delta\tau} &= -\overline{\mathcal{A}}_{h1}(U^{m+2/3})+\frac{H^{m+1;u}(\overline{v}^{m;u})^2\Delta_x^u h_2^c}{h_1^u h_2^u}\\ &\quad+\overline{\mathcal{D}}_{mh1}(\overline{\nu_H}\overline{D}_T(U_B^{m+2/3},V^m))\end{aligned} \tag{5.77}$$

- Value at new time step

$$\tilde{U}^{m+1}=\frac{1}{2}(\tilde{U}_A^{m+1}+\tilde{U}_B^{m+1}) \tag{5.78}$$

The $\overline{\mathcal{O}}_1$-terms are defined by

$$\begin{aligned}\overline{\mathcal{O}}_1 &= fV^{m;u} - \frac{gH^{m+1;u}}{h_1^u}\Delta_x^u\zeta^{m+1} - \frac{H^{m+1;u}}{\rho_0 h_1^u}\Delta_x P_a + \overline{F_1}^{b;n} + H^{m+1;u}F_1^{t;m+1} \\ &\quad + \tau_{s1}^u - k_{b1}^{p;u}\Big(u_b^p - \frac{U^p}{H^{n;u}}\Big) - \overline{\delta A}_{h1}^n + \overline{\delta D}_{h1}^n \qquad (5.79)\end{aligned}$$

A similar procedure is followed for the V-equation (4.87):

- Part A

$$\begin{aligned}\frac{V_A^{m+1/3} - V^m}{\Delta\tau} &= -\overline{\mathcal{A}}_{h1}(V^m) - \frac{\overline{u}^{m;v}V^m\Delta_x^v h_2^{uv}}{h_1^v h_2^v} \\ &\quad + \overline{\mathcal{D}}_{mh1}(\overline{\nu_H}\overline{D}_T(U^m, V^m)) \qquad (5.80)\end{aligned}$$

$$\begin{aligned}\frac{V_A^{m+2/3} - V_A^{m+1/3}}{\Delta\tau} &= -\overline{\mathcal{A}}_{h2}(V_A^{m+1/3}) + \frac{H^{m;v}(\overline{u}^{m;v})^2\Delta_y^v h_1^c}{h_1^v h_2^v} \\ &\quad - \overline{\mathcal{D}}_{mh2}(\overline{\nu_H}\overline{D}_S(U^m, V_A^{m+1/3})) \qquad (5.81)\end{aligned}$$

$$\frac{\tilde{V}_A^{m+1} - V_A^{m+2/3}}{\Delta\tau} + \frac{k_{b2}^v}{H^{m;v}}\tilde{V}_A^{m+1} = \overline{\mathcal{O}}_2 \qquad (5.82)$$

- Part B

$$\frac{V_B^{m+1/3} - V^m}{\Delta\tau} + \frac{k_{b2}^v}{H^{m;v}}V_B^{m+1/3} = \overline{\mathcal{O}}_2 \qquad (5.83)$$

$$\begin{aligned}\frac{V_B^{m+2/3} - V_B^{m+1/3}}{\Delta\tau} &= -\overline{\mathcal{A}}_{h2}(V_B^{m+1/3}) + \frac{H^{m;v}(\overline{u}^{m;v})^2\Delta_y^v h_1^c}{h_1^v h_2^v} \\ &\quad - \overline{\mathcal{D}}_{mh2}(\overline{\nu_H}\overline{D}_S(U^m, V_B^{m+1/3})) \qquad (5.84)\end{aligned}$$

$$\begin{aligned}\frac{\tilde{V}_B^{m+1} - V_A^{m+2/3}}{\Delta\tau} &= -\overline{\mathcal{A}}_{h1}(V^{m+2/3}) - \frac{\overline{u}^{m;v}V^{m+2/3}\Delta_x^v h_2^{uv}}{h_1^v h_2^v} \\ &\quad + \overline{\mathcal{D}}_{mh1}(\overline{\nu_H}\overline{D}_T(U^m, V_B^{m+2/3})) \qquad (5.85)\end{aligned}$$

- Value at new time step

$$\tilde{V}^{m+1} = \frac{1}{2}(\tilde{V}_A^{m+1} + \tilde{V}_B^{m+1}) \qquad (5.86)$$

The $\overline{\mathcal{O}}_2$-terms are defined by

$$\begin{aligned}\overline{\mathcal{O}}_2 &= -fU^{m;v} - \frac{gH^{m+1;v}}{h_2^v}\Delta_y^v\zeta^{m+1} - \frac{H^{m+1;v}}{\rho_0 h_2^v}\Delta_y P_a + \overline{F_2}^{b;n} + H^{m+1;v}F_2^{t;m+1} \\ &\quad + \tau_{s2}^v - k_{b1}^v\left(v_b^p - \frac{V^p}{H^{n;v}}\right) - \overline{\delta A}_{h2}^n + \overline{\delta D}_{h2}^n \end{aligned} \tag{5.87}$$

Once $\tilde{U}^{m+1}$ and $\tilde{V}^{m+1}$ are obtained, an implicit correction is applied as described in Section 5.3.1.2.

Table 5.5: Definitions of the fluxes used in the numerical discretisations.

Type	Purpose
advective fluxes	
F_1	advective flux of a scalar in the X-direction at the U-node $F_1 = u\psi$
F_2	advective flux of a scalar in the Y-direction at the V-node $F_2 = v\psi$
F_3	advective flux of a scalar in the vertical direction at the W-node $F_3 = \omega\psi$
F_{11}	advective flux of u in the X-direction at the C-node $F_{11} = uu$
F_{12}	advective flux of a u in the Y-direction at the UV-node $F_{12} = vu$
F_{21}	advective flux of a v in the X-direction at the UV-node $F_{21} = uv$
F_{22}	advective flux of v in the Y-direction at the C-node $F_{22} = vv$
F_{13}	advective flux of u in the vertical direction at the UW-node $F_{13} = \omega u$
F_{23}	advective flux of v in the vertical direction at the VW-node $F_{23} = \omega v$
$\overline{F}_{11}$	advective flux of U in the X-direction at the C-node $\overline{F}_{11} = \overline{u}U$
$\overline{F}_{12}$	advective flux of U in the Y-direction at the UV-node $\overline{F}_{12} = \overline{v}U$
$\overline{F}_{21}$	advective flux of V in the X-direction at the UV-node $\overline{F}_{21} = \overline{u}V$
$\overline{F}_{22}$	advective flux of V in the Y-direction at the C-node $\overline{F}_{22} = \overline{v}V$

(*Continued*)

Table 5.5: *Continued*

diffusive fluxes	
D_1	diffusive flux of a scalar in the X-direction at the U-node $$D_1 = \frac{\lambda_H}{h_1}\frac{\partial \psi}{\partial \xi_1}$$
D_2	diffusive flux of a scalar in the Y-direction at the V-node $$D_2 = \frac{\lambda_H}{h_2}\frac{\partial \psi}{\partial \xi_2}$$
D_3	diffusive flux of a scalar in the vertical direction at the W-node $$D_3 = \frac{\lambda_T^{\psi}}{h_3}\frac{\partial \psi}{\partial s}$$
D_{11}	diffusive flux in the X-direction (u-equation) at the C-node $$D_{11} = h_2\tau_{11} = \nu_H h_2 D_T$$
D_{12}	diffusive flux in the Y-direction (u-equation) at the UV-node $$D_{12} = h_1\tau_{12} = \nu_H h_1 D_S$$
D_{21}	diffusive flux in the X-direction (v-equation) at the UV-node $$D_{21} = h_2\tau_{21} = \nu_H h_2 D_S$$
D_{22}	diffusive flux in the Y-direction (v-equation) at the C-node $$D_{22} = h_1\tau_{22} = -\nu_H h_1 D_T$$
D_{13}	diffusive flux in the vertical direction (u-equation) at the UW-node $$D_{13} = \frac{\nu_T}{h_3}\frac{\partial u}{\partial s}$$
D_{23}	diffusive flux in the vertical direction (v-equation) at the VW-node $$D_{23} = \frac{\nu_T}{h_3}\frac{\partial v}{\partial s}$$
$\overline{D}_{11}$	diffusive flux in the X-direction (U-equation) at the C-node $$\overline{D}_{11} = h_2\overline{\tau_{11}} = \overline{\nu_H} h_2 \overline{D}_T$$
$\overline{D}_{12}$	diffusive flux in the Y-direction (U-equation) at the UV-node $$\overline{D}_{12} = h_1\overline{\tau_{12}} = \overline{\nu_H} h_1 \overline{D}_S$$
$\overline{D}_{21}$	diffusive flux in the X-direction (V-equation) at the UV-node $$\overline{D}_{21} = h_2\overline{\tau_{21}} = \overline{\nu_H} h_2 \overline{D}_S$$
$\overline{D}_{22}$	diffusive flux in the Y-direction (V-equation) at the C-node $$\overline{D}_{22} = h_1\overline{\tau_{22}} = -\overline{\nu_H} h_1 \overline{D}_T$$

(*Continued*)

Table 5.5: *Continued*

5.3.4 Discretisation of 3-D horizontal advection

The four horizontal advective terms in the 3-D momentum equations are written as the divergence of the horizontal fluxes F_{11}, F_{12}, F_{21}, F_{22}, defined in Table 5.5:

$$\mathcal{A}_{h1}(u) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}\Big(h_2h_3u^2\Big) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}\Big(h_2h_3F_{11}\Big) \tag{5.88}$$

$$\mathcal{A}_{h2}(u) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}\Big(h_1h_3uv\Big) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}\Big(h_1h_3F_{12}\Big) \tag{5.89}$$

$$\mathcal{A}_{h1}(v) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}\Big(h_2h_3uv\Big) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}\Big(h_2h_3F_{21}\Big) \tag{5.90}$$

$$\mathcal{A}_{h2}(v) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}\Big(h_1h_3v^2\Big) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}\Big(h_1h_3F_{22}\Big) \tag{5.91}$$

For simplicity, the k-index and time level will be omitted from the discretisation formulae.

Extended forms of the above operators which include the appropriate curvature term, are defined by (see Table 5.4):

$$\tilde{\mathcal{A}}_{h1}(u) = \mathcal{A}_{h1}(u) - \frac{v^2}{h_1h_2}\frac{\partial h_2}{\partial\xi_1} \tag{5.92}$$

$$\tilde{\mathcal{A}}_{h2}(u) = \mathcal{A}_{h2}(u) + \frac{uv}{h_1h_2}\frac{\partial h_1}{\partial\xi_2} \tag{5.93}$$

$$\tilde{\mathcal{A}}_{h1}(v) = \mathcal{A}_{h1}(v) + \frac{uv}{h_1h_2}\frac{\partial h_2}{\partial\xi_1} \tag{5.94}$$

$$\tilde{\mathcal{A}}_{h2}(v) = \mathcal{A}_{h2}(v) - \frac{u^2}{h_1h_2}\frac{\partial h_1}{\partial\xi_2} \tag{5.95}$$

5.3.4.1 alongstream advection of u

The alongstream advective term in the u-equation (4.61) is obtained by differencing the flux F^c_{11} at the U-node

$$\mathcal{A}_{h1}(u)^u_{ij} = \frac{h^c_{2;ij}h^c_{3;ij}F^c_{11;ij} - h^c_{2;i-1,j}h^c_{3;i-1,j}F^c_{11;i-1,j}}{h^u_{1;ij}h^u_{2;ij}h^u_{3;ij}} \tag{5.96}$$

The flux is calculated from

$$F^c_{11;ij} = \Big(1 - \Omega(r^c_{ij})\Big)F^c_{up;ij} + \Omega(r^c_{ij})F^c_{lw;ij} \tag{5.97}$$

where $F^c_{up;ij}$ and $F^c_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the C-node:

$$F^c_{up;ij} = \frac{1}{2}u^c_{ij}\Big((1+s_{ij})u_{ij} + (1-s_{ij})u_{i+1,j}\Big) \tag{5.98}$$

$$F^c_{lw;ij} = \frac{1}{2}u^c_{ij}\Big((1+c_{ij})u_{ij} + (1-c_{ij})u_{i+1,j}\Big) \tag{5.99}$$

where s_{ij} and c_{ij} are the sign and CFL number of the advecting current

$$s_{ij} = \mathrm{Sign}(u^c_{ij})\,, \quad c_{ij} = \frac{u^c_{ij}\Delta t}{h^c_{1;ij}} \tag{5.100}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_3D. The argument r of the weight function is defined by

$$\begin{aligned} r^c_{ij} &= \frac{(1+s_{ij})\Delta F^c_{i-1,j} + (1-s_{ij})\Delta F^c_{i+1,j}}{2\Delta F^c_{ij}} \\ \Delta F^c_{ij} &= F^c_{lw;ij} - F^c_{up;ij} \end{aligned} \tag{5.101}$$

The extended advective term is discretised as

$$\tilde{\mathcal{A}}_{h1}(u)^u_{ij} = \mathcal{A}_{h1}(u)^u_{ij} - \frac{(v^u_{ij})^2\Delta^u_x h^c_{2;ij}}{h^u_{1;ij}h^u_{2;ij}} \tag{5.102}$$

5.3.4.2 cross-stream advection of u

The cross-stream advective term in the u-equation (4.61) is obtained by differencing the flux F^{uv}_{12} at the U-node

$$\mathcal{A}_{h2}(u)^u_{ij} = \frac{h^{uv}_{1;i,j+1}h^{uv}_{3;i,j+1}F^{uv}_{12;i,j+1} - h^{uv}_{1;ij}h^{uv}_{3;ij}F^{uv}_{12;ij}}{h^u_{1;ij}h^u_{2;ij}h^u_{3;ij}} \tag{5.103}$$

The flux is calculated from

$$F^{uv}_{12;ij} = \Big(1-\Omega(r^{uv}_{ij})\Big)F^{uv}_{up;ij} + \Omega(r^{uv}_{ij})F^{uv}_{lw;ij} \tag{5.104}$$

where $F^{uv}_{up;ij}$ and $F^{uv}_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the UV-node:

$$F^{uv}_{up;ij} = \frac{1}{2}v^{uv}_{ij}\Big((\alpha_{ij}+s_{ij})u_{i,j-1} + (\beta_{ij}-s_{ij})u_{ij}\Big) \tag{5.105}$$

$$F^{uv}_{lw;ij} = \frac{1}{2}v^{uv}_{ij}\Big((\alpha_{ij}+c_{ij})u_{i,j-1} + (\beta_{ij}-c_{ij})u_{ij}\Big) \tag{5.106}$$

where

$$s_{ij} = \text{Sign}(v^{uv}_{ij})\,, \quad c_{ij} = \frac{v^{uv}_{ij}\Delta t}{h^{uv}_{2;ij}} \tag{5.107}$$

$$\alpha_{ij} = \frac{h^{u}_{2;ij}}{h^{uv}_{2;ij}}\,, \quad \beta_{ij} = \frac{h^{u}_{2;i,j-1}}{h^{uv}_{2;ij}} \tag{5.108}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_3D. The argument r of the weight function is defined by

$$\begin{aligned} r^{uv}_{ij} &= \frac{(\alpha_{ij}+s_{ij})\Delta F^{uv}_{i,j-1} + (\beta_{ij}-s_{ij})\Delta F^{uv}_{i,j+1}}{2\Delta F^{uv}_{ij}} \\ \Delta F^{uv}_{ij} &= F^{uv}_{lw;ij} - F^{uv}_{up;ij} \end{aligned} \tag{5.109}$$

The extended advective term is discretised as

$$\tilde{\mathcal{A}}_{h2}(u)^{u}_{ij} = \mathcal{A}_{h2}(u)_{ij} + \frac{u_{ij}v^{u}_{ij}\Delta^{u}_{y}h^{uv}_{1;ij}}{h^{u}_{1;ij}h^{u}_{2;ij}} \tag{5.110}$$

5.3.4.3 cross-stream advection of v

The cross-stream advective term in the v-equation (4.62) is obtained by differencing the flux F^{uv}_{21} at the V-node

$$\mathcal{A}_{h1}(v)^{v}_{ij} = \frac{h^{uv}_{2;i+1,j}h^{uv}_{3;i+1,j}F^{uv}_{21;i+1,j} - h^{uv}_{2;ij}h^{uv}_{3;ij}F^{uv}_{21;ij}}{h^{v}_{1;ij}h^{v}_{2;ij}h^{v}_{3;ij}} \tag{5.111}$$

The flux is calculated from

$$F^{uv}_{21;ij} = \Big(1-\Omega(r^{uv}_{ij})\Big)F^{uv}_{up;ij} + \Omega(r^{uv}_{ij})F^{uv}_{lw;ij} \tag{5.112}$$

where $F^{uv}_{up;ij}$ and $F^{uv}_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the UV-node:

$$F^{uv}_{up;ij} = \frac{1}{2}u^{uv}_{ij}\Big((\alpha_{ij}+s_{ij})v_{i-1,j} + (\beta_{ij}-s_{ij})v_{ij}\Big) \tag{5.113}$$

$$F^{uv}_{lw;ij} = \frac{1}{2}u^{uv}_{ij}\Big((\alpha_{ij}+c_{ij})v_{i-1,j} + (\beta_{ij}-c_{ij})v_{ij}\Big) \tag{5.114}$$

where

$$s_{ij} = \text{Sign}(u^{uv}_{ij})\,, \quad c_{ij} = \frac{u^{uv}_{ij}\Delta t}{h^{uv}_{1;ij}} \tag{5.115}$$

$$\alpha_{ij} = \frac{h^{v}_{1;ij}}{h^{uv}_{1;ij}}\,, \quad \beta_{ij} = \frac{h^{v}_{1;i-1,j}}{h^{uv}_{1;ij}} \tag{5.116}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_3D. The argument r of the weight function is defined by

$$\begin{aligned} r^{uv}_{ij} &= \frac{(\alpha_{ij}+s_{ij})\Delta F^{uv}_{i-1,j} + (\beta_{ij}-s_{ij})\Delta F^{uv}_{i+1,j}}{2\Delta F^{uv}_{ij}} \\ \Delta F^{uv}_{ij} &= F^{uv}_{lw;ij} - F^{uv}_{up;ij} \end{aligned} \tag{5.117}$$

The extended advective term is discretised as

$$\tilde{\mathcal{A}}_{h1}(v)^v_{ij} = \mathcal{A}_{h1}(v)^v_{ij} + \frac{u^v_{ij} v_{ij} \Delta^v_x h^{uv}_{2;ij}}{h^v_{1;ij} h^v_{2;ij}} \tag{5.118}$$

5.3.4.4 alongstream advection of v

The alongstream advective term in the v-equation (4.62) is obtained by differencing the flux F^c_{22} at the V-node

$$\mathcal{A}_{h2}(v)^v_{ij} = \frac{h^c_{1;ij} h^c_{3;ij} F^c_{22;ij} - h^c_{1;i,j-1} h^c_{3;i,j-1} F^c_{22;i,j-1}}{h^v_{1;ij} h^v_{2;ij} h^v_{3;ij}} \tag{5.119}$$

The flux is calculated from

$$F^c_{22;ij} = \Big(1 - \Omega(r^c_{ij})\Big) F^c_{up;ij} + \Omega(r^c_{ij}) F^c_{lw;ij} \tag{5.120}$$

where $F^c_{up;ij}$ and $F^c_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the C-node:

$$F^c_{up;ij} = \frac{1}{2} v^c_{ij} \Big((1+s_{ij}) v_{ij} + (1-s_{ij}) v_{i,j+1} \Big) \tag{5.121}$$

$$F^c_{lw;ij} = \frac{1}{2} v^c_{ij} \Big((1+c_{ij}) v_{ij} + (1-c_{ij}) v_{i,j+1} \Big) \tag{5.122}$$

where s_{ij} and c_{ij} are the sign and CFL number of the advecting current

$$s_{ij} = \mathrm{Sign}(v^c_{ij}), \quad c_{ij} = \frac{v^c_{ij} \Delta t}{h^c_{2;ij}} \tag{5.123}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_3D. The argument r of the weight function is defined by

$$\begin{aligned} r^c_{ij} &= \frac{(1+s_{ij})\Delta F^c_{i,j-1} + (1-s_{ij})\Delta F^c_{i,j+1}}{2\Delta F^c_{ij}} \\ \Delta F^c_{ij} &= F^c_{lw;ij} - F^c_{up;ij} \end{aligned} \tag{5.124}$$

The extended advective term is discretised as

$$\tilde{\mathcal{A}}_{h2}(v)^v_{ij} = \mathcal{A}_{h2}(v)^v_{ij} - \frac{(u^v_{ij})^2 \Delta^v_y h^c_{1;ij}}{h^v_{1;ij} h^v_{2;ij}} \tag{5.125}$$

5.3.5 Discretisation of 2-D horizontal advection

The four horizontal advective terms in the 2-D momentum equations are written as the divergence of the horizontal fluxes $\overline{F}_{11}$, $\overline{F}_{12}$, $\overline{F}_{21}$, $\overline{F}_{22}$, defined in Table 5.5:

$$\overline{\mathcal{A}}_{h1}(U) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_1}(h_2 \overline{u} U) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_1}(h_2 \overline{F}_{11}) \tag{5.126}$$

$$\overline{\mathcal{A}}_{h2}(U) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_2}(h_1 \overline{v} U) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_2}(h_1 \overline{F}_{12}) \tag{5.127}$$

$$\overline{\mathcal{A}}_{h1}(V) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_1}(h_2 \overline{u} V) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_1}(h_2 \overline{F}_{21}) \tag{5.128}$$

$$\overline{\mathcal{A}}_{h2}(V) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_2}(h_1 \overline{v} V) = \frac{1}{h_1 h_2}\frac{\partial}{\partial \xi_2}(h_1 \overline{F}_{22}) \tag{5.129}$$

Extended forms of the above operators which include the appropriate curvature term, are defined by (see Table 5.4):

$$\overline{\tilde{\mathcal{A}}}_{h1}(U) = \overline{\mathcal{A}}_{h1}(U) - \frac{\overline{v} V}{h_1 h_2}\frac{\partial h_2}{\partial \xi_1} \tag{5.130}$$

$$\overline{\tilde{\mathcal{A}}}_{h2}(U) = \overline{\mathcal{A}}_{h2}(U) + \frac{\overline{v} U}{h_1 h_2}\frac{\partial h_1}{\partial \xi_2} \tag{5.131}$$

$$\overline{\tilde{\mathcal{A}}}_{h1}(V) = \overline{\mathcal{A}}_{h1}(V) + \frac{\overline{u} V}{h_1 h_2}\frac{\partial h_2}{\partial \xi_1} \tag{5.132}$$

$$\overline{\tilde{\mathcal{A}}}_{h2}(V) = \overline{\mathcal{A}}_{h2}(V) - \frac{\overline{u} U}{h_1 h_2}\frac{\partial h_1}{\partial \xi_2} \tag{5.133}$$

5.3.5.1 alongstream advection of U

The alongstream advective term in the U-equation (4.86) is obtained by differencing the flux $\overline{F}^c_{11}$ at the U-node

$$\overline{\mathcal{A}}_{h1}(U)^u_{ij} = \frac{h^c_{2;ij}\overline{F}^c_{11;ij} - h^c_{2;i-1,j}\overline{F}^c_{11;i-1,j}}{h^u_{1;ij} h^u_{2;ij}} \tag{5.134}$$

The flux is calculated from

$$\overline{F}^c_{11;ij} = \Big(1 - \Omega(r^c_{ij})\Big)\overline{F}^c_{up;ij} + \Omega(r^c_{ij})\overline{F}^c_{lw;ij} \tag{5.135}$$

where $\overline{F}^c_{up;ij}$ and $\overline{F}^c_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the C-node:

$$\overline{F}^c_{up;ij} = \frac{1}{2}\overline{u}^c_{ij}\Big((1 + s_{ij})U_{ij} + (1 - s_{ij})U_{i+1,j}\Big) \tag{5.136}$$

$$\overline{F}^c_{lw;ij} = \frac{1}{2}\overline{u}^c_{ij}\Big((1+c_{ij})U_{ij} + (1-c_{ij})U_{i+1,j}\Big) \tag{5.137}$$

where s_{ij} and c_{ij} are the sign and CFL number of the advecting current

$$s_{ij} = \text{Sign}(\overline{u}^c_{ij})\,, \quad c_{ij} = \frac{\overline{u}^c_{ij}\Delta\tau}{h^c_{1;ij}} \tag{5.138}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_2D. The argument r of the weight function is defined by

$$\begin{aligned} r^c_{ij} &= \frac{(1+s_{ij})\Delta\overline{F}^c_{i-1,j} + (1-s_{ij})\Delta\overline{F}^c_{i+1,j}}{2\Delta\overline{F}^c_{ij}} \\ \Delta\overline{F}^c_{ij} &= \overline{F}^c_{lw;ij} - \overline{F}^c_{up;ij} \end{aligned} \tag{5.139}$$

The extended advective term is discretised by

$$\overline{\tilde{\mathcal{A}}}_{h1}(U)^u_{ij} = \overline{\mathcal{A}}_{h1}(U)_{ij} - \frac{\overline{v}^u_{ij}V^u_{ij}\Delta^u_x h^c_{2;ij}}{h^u_{1;ij}h^u_{2;ij}} \tag{5.140}$$

5.3.5.2 cross-stream advection of U

The cross-stream advective term in the U-equation (4.86) is obtained by differencing the flux $\overline{F}^{uv}_{12}$ at the U-node

$$\overline{\mathcal{A}}_{h2}(U)^u_{ij} = \frac{h^{uv}_{1;i,j+1}\overline{F}^{uv}_{12;i,j+1} - h^{uv}_{1;ij}\overline{F}^{uv}_{12;ij}}{h^u_{1;ij}h^u_{2;ij}} \tag{5.141}$$

The flux is calculated from

$$\overline{F}^{uv}_{12;ij} = \Big(1-\Omega(r^{uv}_{ij})\Big)\overline{F}^{uv}_{up;ij} + \Omega(r^{uv}_{ij})\overline{F}^{uv}_{lw;ij} \tag{5.142}$$

where $\overline{F}^{uv}_{up;ij}$ and $\overline{F}^{uv}_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the UV-node:

$$\overline{F}^{uv}_{up;ij} = \frac{1}{2}\overline{v}^{uv}_{ij}\Big((\alpha_{ij}+s_{ij})U_{i,j-1} + (\beta_{ij}-s_{ij})U_{ij}\Big) \tag{5.143}$$

$$\overline{F}^{uv}_{lw;ij} = \frac{1}{2}\overline{v}^{uv}_{ij}\Big((\alpha_{ij}+c_{ij})U_{i,j-1} + (\beta_{ij}-c_{ij})U_{ij})\Big) \tag{5.144}$$

where

$$s_{ij} = \text{Sign}(\overline{v}^{uv}_{ij})\,, \quad c_{ij} = \frac{\overline{v}^{uv}_{ij}\Delta\tau}{h^{uv}_{2;ij}} \tag{5.145}$$

$$\alpha_{ij} = \frac{h^u_{2;ij}}{h^{uv}_{2;ij}}\,, \quad \beta_{ij} = \frac{h^u_{2;i,j-1}}{h^{uv}_{2;ij}} \tag{5.146}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_2D. The argument r of the weight function is defined by

$$\begin{aligned} r^{uv}_{ij} &= \frac{(\alpha_{ij}+s_{ij})\Delta\overline{F}^{uv}_{i,j-1} + (\beta_{ij}-s_{ij})\Delta\overline{F}^{uv}_{i,j+1}}{2\Delta\overline{F}^{uv}_{ij}} \\ \Delta\overline{F}^{uv}_{ij} &= \overline{F}^{uv}_{lw;ij} - \overline{F}^{uv}_{up;ij} \end{aligned} \tag{5.147}$$

The extended advective term is discretised by

$$\overline{\tilde{\mathcal{A}}}_{h2}(U)^u_{ij} = \overline{\mathcal{A}}_{h2}(U)_{ij} + \frac{\overline{v}^u_{ij}U_{ij}\Delta^u_y h^{uv}_{1;ij}}{h^u_{1;ij}h^u_{2;ij}} \tag{5.148}$$

5.3.5.3 cross-stream advection of V

The cross-stream advective term in the V-equation (4.87) is obtained by differencing the flux $\overline{F}^{uv}_{21}$ at the V-node

$$\overline{\mathcal{A}}_{h1}(V)^v_{ij} = \frac{h^{uv}_{2;i+1,j}\overline{F}^{uv}_{21;i+1,j} - h^{uv}_{2;ij}\overline{F}^{uv}_{21;ij}}{h^v_{1;ij}h^v_{2;ij}} \tag{5.149}$$

The flux is calculated from

$$\overline{F}^{uv}_{21;ij} = \Big(1-\Omega(r^{uv}_{ij})\Big)\overline{F}^{uv}_{up;ij} + \Omega(r^{uv}_{ij})\overline{F}^{uv}_{lw;ij} \tag{5.150}$$

where $\overline{F}^{uv}_{up;ij}$ and $\overline{F}^{uv}_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the UV-node:

$$\overline{F}^{uv}_{up;ij} = \frac{1}{2}\overline{u}^{uv}_{ij}\Big((\alpha_{ij}+s_{ij})V_{i-1,j} + (\beta_{ij}-s_{ij})V_{i,j}\Big) \tag{5.151}$$

$$\overline{F}^{uv}_{lw;ij} = \frac{1}{2}\overline{u}^{uv}_{ij}\Big((\alpha_{ij}+c_{ij})V_{i-1,j} + (\beta_{ij}-c_{ij})V_{ij}\Big) \tag{5.152}$$

where

$$s_{ij} = \mathrm{Sign}(\overline{u}^{uv}_{ij})\,, \quad c_{ij} = \frac{\overline{u}^{uv}_{ij}\Delta\tau}{h^{uv}_{1;ij}} \tag{5.153}$$

$$\alpha_{ij} = \frac{h^v_{1;ij}}{h^{uv}_{1;ij}}\,, \quad \beta_{ij} = \frac{h^v_{1;i,j-1}}{h^{uv}_{1;ij}} \tag{5.154}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_2D. The argument r of the weight function is defined by

$$\begin{aligned} r^{uv}_{ij} &= \frac{(\alpha_{ij}+s_{ij})\Delta\overline{F}^{uv}_{i-1,j}+(\beta_{ij}-s_{ij})\Delta\overline{F}^{uv}_{i+1,j}}{2\Delta\overline{F}^{uv}_{ij}} \\ \Delta\overline{F}^{uv}_{ij} &= \overline{F}^{uv}_{lw;ij}-\overline{F}^{uv}_{up;ij} \end{aligned} \tag{5.155}$$

The extended advective term is discretised as

$$\overline{\tilde{\mathcal{A}}}_{h1}(V)^{v}_{ij}=\overline{\mathcal{A}}_{h1}(V)_{ij}+\frac{\overline{u}^{v}_{ij}V_{ij}\Delta^{v}_{x}h^{uv}_{2;ij}}{h^{v}_{1;ij}h^{v}_{2;ij}} \tag{5.156}$$

5.3.5.4 alongstream advection of V

The alongstream advective term in the V-equation (4.87) is obtained by differencing the flux $\overline{F}^{c}_{22}$ at the V-node

$$\overline{\mathcal{A}}_{h2}(V)^{v}_{ij}=\frac{h^{c}_{1;ij}\overline{F}^{c}_{22;ij}-h^{c}_{1;i,j-1}\overline{F}^{c}_{22;i,j-1}}{h^{v}_{1;ij}h^{v}_{2;ij}} \tag{5.157}$$

The flux is calculated from

$$\overline{F}^{c}_{22;ij}=\Big(1-\Omega(r^{c}_{ij})\Big)\overline{F}^{c}_{up;ij}+\Omega(r^{c}_{ij})\overline{F}^{c}_{lw;ij} \tag{5.158}$$

where $\overline{F}^{c}_{up;ij}$ and $\overline{F}^{c}_{lw;ij}$ are the upwind and Lax-Wendroff fluxes at the C-node:

$$\overline{F}^{c}_{up;ij} = \frac{1}{2}\overline{v}^{c}_{ij}\Big((1+s_{ij})V_{ij}+(1-s_{ij})V_{i,j+1}\Big) \tag{5.159}$$

$$\overline{F}^{c}_{lw;ij} = \frac{1}{2}\overline{v}^{c}_{ij}\Big((1+c_{ij})V_{ij}+(1-c_{ij})V_{i,j+1}\Big) \tag{5.160}$$

where s_{ij} and c_{ij} are the sign and CFL number of the advecting current

$$s_{ij}=\mathrm{Sign}(\overline{v}^{c}_{ij})\,,\quad c_{ij}=\frac{\overline{v}^{c}_{ij}\Delta\tau}{h^{c}_{2;ij}} \tag{5.161}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_2D. The argument r of the weight function is defined by

$$r^{c}_{ij} = \frac{(1+s_{ij})\Delta\overline{F}^{c}_{i,j-1}+(1-s_{ij})\Delta\overline{F}^{c}_{i,j+1}}{2\Delta\overline{F}^{c}_{ij}}$$

$$\Delta\overline{F}^c_{ij} = \overline{F}^c_{lw;ij} - \overline{F}^c_{up;ij} \tag{5.162}$$

The extended advective term is discretised as

$$\overline{\tilde{\mathcal{A}}}_{h2}(V)^v_{ij} = \overline{\mathcal{A}}_{h2}(V)_{ij} - \frac{\overline{u}^v_{ij} U^v_{ij} \Delta^v_y h^c_{1;ij}}{h^v_{1;ij} h^v_{2;ij}} \tag{5.163}$$

5.3.6 Integrals of the baroclinic advection terms

The discretised versions of the advective integrals in the 2-D momentum equations at time step t^n are given by

$$\overline{\delta A}^u_{h1;ij} = \sum_{k=1}^{N_z} \Big(\tilde{\mathcal{A}}_{h1}(u)^u_{ijk} + \tilde{\mathcal{A}}_{h2}(u)^u_{ijk} \Big) h^u_{3;ijk} - \overline{\tilde{\mathcal{A}}}_{h1}(U)^u_{ij} - \overline{\tilde{\mathcal{A}}}_{h2}(U)^u_{ij} \tag{5.164}$$

$$\overline{\delta A}^v_{h2;ij} = \sum_{k=1}^{N_z} \Big(\tilde{\mathcal{A}}_{h1}(v)^v_{ijk} + \tilde{\mathcal{A}}_{h2}(v)^v_{ijk} \Big) h^v_{3;ijk} - \overline{\tilde{\mathcal{A}}}_{h1}(V)^v_{ij} - \overline{\tilde{\mathcal{A}}}_{h2}(V)^v_{ij} \tag{5.165}$$

5.3.7 Discretisation of vertical advection

The vertical advection terms in the 3-D momentum equations are written as the divergence of the of the vertical fluxes F_{13}, F_{23}, defined in Table 5.5:

$$\mathcal{A}_v(u) = \frac{1}{h_3}\frac{\partial}{\partial s}(\omega u) = \frac{1}{h_3}\frac{\partial F_{13}}{\partial s} \tag{5.166}$$

$$\mathcal{A}_v(v) = \frac{1}{h_3}\frac{\partial}{\partial s}(\omega v) = \frac{1}{h_3}\frac{\partial F_{23}}{\partial s} \tag{5.167}$$

5.3.7.1 vertical advection of u

The vertical advective term in the u-equation (4.61) is obtained by differencing the flux F^{uw}_{13} at the U-node

$$\mathcal{A}_v(u)^u_{ijk} = \frac{F^{uw}_{13;ij,k+1} - F^{uw}_{13;ijk}}{h^u_{3;ijk}} \tag{5.168}$$

The flux is calculated from

$$F^{uw}_{13;ijk} = \Big(1 - \Omega(r^{uw}_{ijk})\Big) F^{uw}_{up;ijk} + \Omega(r^{uw}_{ijk}) F^{uw}_{ce;ijk} \tag{5.169}$$

where $F^{uw}_{up;ijk}$ and $F^{uw}_{ce;ijk}$ are the upwind and central fluxes at the UW-node:

$$F^{uw}_{up;ijk} = \frac{1}{2}\omega^{uw}_{ijk}\Big((\alpha_{ijk} + s_{ijk}) u_{ij,k-1} + (\beta_{ijk} - s_{ijk}) u_{ijk} \Big) \tag{5.170}$$

$$F^{uw}_{ce;ijk} = \frac{1}{2}\omega^{uw}_{ijk}(\alpha_{ijk}u_{ij,k-1} + \beta_{ijk}u_{ijk}) \tag{5.171}$$

where

$$s_{ijk} = \text{Sign}(\omega^{uw}_{ijk})\,, \quad \alpha_{ijk} = \frac{h^{u}_{3;ijk}}{h^{uw}_{3;ijk}}\,, \quad \beta_{ijk} = \frac{h^{u}_{3;ij,k-1}}{h^{uw}_{3;ijk}} \tag{5.172}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_3D. The argument r of the weight function is defined by

$$\begin{aligned} r^{uw}_{ijk} &= \frac{(\alpha_{ijk} + s_{ijk})\Delta F^{uw}_{ij,k-1} + (\beta_{ijk} - s_{ijk})\Delta F^{uw}_{ij,k+1}}{2\Delta F^{uw}_{ijk}} \\ \Delta F^{uw}_{ijk} &= F^{uw}_{ce;ijk} - F^{uw}_{up;ijk} \end{aligned} \tag{5.173}$$

5.3.7.2 vertical advection of v

The vertical advective term in the v-equation (4.62) is obtained by differencing the flux F^{vw}_{23} at the V-node

$$\mathcal{A}_v(v)^{v}_{ijk} = \frac{F^{vw}_{23;ij,k+1} - F^{vw}_{23;ijk}}{h^{v}_{3;ijk}} \tag{5.174}$$

The flux is calculated from

$$F^{vw}_{23;ijk} = \Big(1 - \Omega(r^{vw}_{ijk})\Big)F^{vw}_{up;ijk} + \Omega(r^{vw}_{ijk})F^{vw}_{ce;ijk} \tag{5.175}$$

where $F^{vw}_{up;ijk}$ and $F^{vw}_{ce;ijk}$ are the upwind and central fluxes at the VW-node:

$$F^{vw}_{up;ijk} = \frac{1}{2}\omega^{vw}_{ijk}\Big((\alpha_{ijk} + s_{ijk})v_{ij,k-1} + (\beta_{ijk} - s_{ijk})v_{ijk}\Big) \tag{5.176}$$

$$F^{vw}_{ce;ijk} = \frac{1}{2}\omega^{vw}_{ijk}(\alpha_{ijk}v_{ij,k-1} + \beta_{ijk}v_{ijk}) \tag{5.177}$$

where

$$s_{ijk} = \text{Sign}(\omega^{vw}_{ijk})\,, \quad \alpha_{ijk} = \frac{h^{v}_{3;ijk}}{h^{vw}_{3;ijk}}\,, \quad \beta_{ijk} = \frac{h^{v}_{3;ij,k-1}}{h^{vw}_{3;ijk}} \tag{5.178}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_3D. The argument r of the weight function is defined by

$$\begin{aligned} r^{vw}_{ijk} &= \frac{(\alpha_{ijk} + s_{ijk})\Delta F^{vw}_{ij,k-1} + (\beta_{ijk} - s_{ijk})\Delta F^{vw}_{ij,k+1}}{2\Delta F^{vw}_{ijk}} \\ \Delta F^{vw}_{ijk} &= F^{vw}_{ce;ijk} - F^{vw}_{up;ijk} \end{aligned} \tag{5.179}$$

5.3.8 Discretisation of 3-D horizontal diffusion

The four horizontal diffusion terms in the 3-D momentum equations are written as the divergence of the horizontal fluxes D_{11}, D_{12}, D_{21}, D_{22}, defined in Table 5.5:

$$\mathcal{D}_{mh1}(\tau_{11}) = \frac{1}{h_1 h_2^2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2^2 h_3 \tau_{11}\Big) = \frac{1}{h_1 h_2^2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2 h_3 D_{11}\Big) \quad (5.180)$$

$$\mathcal{D}_{mh2}(\tau_{12}) = \frac{1}{h_1^2 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1^2 h_3 \tau_{12}\Big) = \frac{1}{h_1^2 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1 h_3 D_{12}\Big) \quad (5.181)$$

$$\mathcal{D}_{mh1}(\tau_{21}) = \frac{1}{h_1 h_2^2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2^2 h_3 \tau_{21}\Big) = \frac{1}{h_1 h_2^2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2 h_3 D_{21}\Big) \quad (5.182)$$

$$\mathcal{D}_{mh2}(\tau_{22}) = \frac{1}{h_1^2 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1^2 h_3 \tau_{22}\Big) = \frac{1}{h_1^2 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1 h_3 D_{22}\Big) \quad (5.183)$$

Discretisations for the horizontal diffusion terms in the 3-D momentum equations are given below. For simplicity, the k-index and time level will be omitted.

- alongstream diffusion in the u-equation (4.61) at the U-node

$$\mathcal{D}_{mh1}(\tau_{11})^u_{ij} = \frac{h^c_{2;ij}h^c_{3;ij}D^c_{11;ij} - h^c_{2;i-1,j}h^c_{3;i-1,j}D^c_{11;i-1,j}}{h^u_{1;ij}(h^u_{2;ij})^2 h^u_{3;ij}} \quad (5.184)$$

$$D^c_{11;ij} = \nu^c_{H;ij}h^c_{2;ij}\Big[\frac{h^c_{2;ij}}{h^c_{1;ij}}\Delta^c_x\Big(\frac{u_{ij}}{h^u_{2;ij}}\Big) - \frac{h^c_{1;ij}}{h^c_{2;ij}}\Delta^c_y\Big(\frac{v_{ij}}{h^v_{1;ij}}\Big)\Big] \quad (5.185)$$

- cross-stream diffusion in the u-equation (4.61) at the U-node

$$\mathcal{D}_{mh2}(\tau_{12})^u_{ij} = \frac{h^{uv}_{1;i,j+1}h^{uv}_{3;i,j+1}D^{uv}_{12;i,j+1} - h^{uv}_{1;ij}h^{uv}_{3;ij}D^{uv}_{12;ij}}{(h^u_{1;ij})^2 h^u_{2;ij}h^u_{3;ij}} \quad (5.186)$$

$$D^{uv}_{12;ij} = \nu^{uv}_{H;ij}h^{uv}_{1;ij}\Big[\frac{h^{uv}_{1;ij}}{h^{uv}_{2;ij}}\Delta^{uv}_y\Big(\frac{u_{ij}}{h^u_{1;ij}}\Big) + \frac{h^{uv}_{2;ij}}{h^{uv}_{1;ij}}\Delta^{uv}_x\Big(\frac{v_{ij}}{h^v_{2;ij}}\Big)\Big] \quad (5.187)$$

- cross-stream diffusion in the v-equation (4.62) at the V-node

$$\mathcal{D}_{mh1}(\tau_{21})^v_{ij} = \frac{h^{uv}_{2;i+1,j}h^{uv}_{3;i+1,j}D^{uv}_{21;i+1,j} - h^{uv}_{2;ij}h^{uv}_{3;ij}D^{uv}_{21;ij}}{h^v_{1;ij}(h^v_{2;ij})^2 h^v_{3;ij}} \quad (5.188)$$

$$D^{uv}_{21;ij} = \nu^{uv}_{H;ij}h^{uv}_{2;ij}\Big[\frac{h^{uv}_{1;ij}}{h^{uv}_{2;ij}}\Delta^{uv}_y\Big(\frac{u_{ij}}{h^u_{1;ij}}\Big) + \frac{h^{uv}_{2;ij}}{h^{uv}_{1;ij}}\Delta^{uv}_x\Big(\frac{v_{ij}}{h^v_{2;ij}}\Big)\Big] \quad (5.189)$$

- alongstream diffusion in the v-equation (4.62) at the V-node

$$\mathcal{D}_{mh2}(\tau_{22})^v_{ij} = \frac{h^c_{1;ij}h^c_{3;ij}D^c_{22;ij} - h^c_{1;i,j-1}h^c_{3;i,j-1}D^c_{22;i,j-1}}{(h^v_{1;ij})^2 h^v_{2;ij}h^v_{3;ij}} \tag{5.190}$$

$$D^c_{22;ij} = \nu^c_{H;ij}h^c_{1;ij}\Big[\frac{h^c_{1;ij}}{h^c_{2;ij}}\Delta^c_y\Big(\frac{v_{ij}}{h^v_{1;ij}}\Big) - \frac{h^c_{2;ij}}{h^c_{1;ij}}\Delta^c_x\Big(\frac{u_{ij}}{h^u_{2;ij}}\Big)\Big] \tag{5.191}$$

5.3.9 Discretisation of 2-D horizontal diffusion

The four horizontal diffusion terms in the 2-D momentum equations are written as the divergence of the horizontal fluxes $\overline{D}_{11}$, $\overline{D}_{12}$, $\overline{D}_{21}$, $\overline{D}_{22}$, defined in Table 5.5:

$$\overline{\mathcal{D}}_{mh1}(\overline{\tau_{11}}) = \frac{1}{h_1h_2^2}\frac{\partial}{\partial\xi_1}\Big(h_2^2\overline{\tau_{11}}\Big) = \frac{1}{h_1h_2^2}\frac{\partial}{\partial\xi_1}\Big(h_2\overline{D}_{11}\Big) \tag{5.192}$$

$$\overline{\mathcal{D}}_{mh2}(\overline{\tau_{12}}) = \frac{1}{h_1^2h_2}\frac{\partial}{\partial\xi_2}\Big(h_1^2\overline{\tau_{12}}\Big) = \frac{1}{h_1^2h_2}\frac{\partial}{\partial\xi_2}\Big(h_1\overline{D}_{12}\Big) \tag{5.193}$$

$$\overline{\mathcal{D}}_{mh1}(\overline{\tau_{21}}) = \frac{1}{h_1h_2^2}\frac{\partial}{\partial\xi_1}\Big(h_2^2\overline{\tau_{21}}\Big) = \frac{1}{h_1h_2^2}\frac{\partial}{\partial\xi_1}\Big(h_2\overline{D}_{21}\Big) \tag{5.194}$$

$$\overline{\mathcal{D}}_{mh2}(\overline{\tau_{22}}) = \frac{1}{h_1^2h_2}\frac{\partial}{\partial\xi_2}\Big(h_1^2\overline{\tau_{22}}\Big) = \frac{1}{h_1^2h_2}\frac{\partial}{\partial\xi_2}\Big(h_1\overline{D}_{22}\Big) \tag{5.195}$$

Discretisations for the horizontal diffusion terms in the 2-D momentum equations are given below.

- alongstream diffusion in the U-equation (4.86) at the U-node

$$\overline{\mathcal{D}}_{mh1}(\overline{\tau_{11}})^u_{ij} = \frac{h^c_{2;ij}\overline{D}^c_{11;ij} - h^c_{2;i-1,j}\overline{D}^c_{11;i-1,j}}{h^u_{1;ij}(h^u_{2;ij})^2} \tag{5.196}$$

$$\overline{D}^c_{11;ij} = \overline{\nu_H}^c_{ij}h^c_{2;ij}\Big[\frac{h^c_{2;ij}}{h^c_{1;ij}}\Delta^c_x\Big(\frac{\overline{u}_{ij}}{h^u_{2;ij}}\Big) - \frac{h^c_{1;ij}}{h^c_{2;ij}}\Delta^c_y\Big(\frac{\overline{v}_{ij}}{h^v_{1;ij}}\Big)\Big] \tag{5.197}$$

- cross-stream diffusion in the U-equation (4.86) at the U-node

$$\overline{\mathcal{D}}_{mh2}(\overline{\tau_{12}})^u_{ij} = \frac{h^{uv}_{1;i,j+1}\overline{D}^{uv}_{12;i,j+1} - h^{uv}_{1;ij}\overline{D}^{uv}_{12;ij}}{(h^u_{1;ij})^2 h^u_{2;ij}} \tag{5.198}$$

$$\overline{D}^{uv}_{12;ij} = \overline{\nu_H}^{uv}_{ij}h^{uv}_{1;ij}\Big[\frac{h^{uv}_{1;ij}}{h^{uv}_{2;ij}}\Delta^{uv}_y\Big(\frac{\overline{u}_{ij}}{h^u_{1;ij}}\Big) + \frac{h^{uv}_{2;ij}}{h^{uv}_{1;ij}}\Delta^{uv}_x\Big(\frac{\overline{v}_{ij}}{h^v_{2;ij}}\Big)\Big] \tag{5.199}$$

- cross-stream diffusion in the V-equation (4.87) at the V-node

$$\overline{\mathcal{D}}_{mh1}(\overline{\tau_{21}})^v_{ij} = \frac{h^{uv}_{2;i+1,j}\overline{D}^{uv}_{21;i+1,j} - h^{uv}_{2;ij}\overline{D}^{uv}_{21;ij}}{h^v_{1;ij}(h^v_{2;ij})^2} \tag{5.200}$$

$$\overline{D}^{uv}_{21;ij} = \overline{\nu_H}^{uv}_{ij} h^{uv}_{2;ij}\Big[\frac{h^{uv}_{1;ij}}{h^{uv}_{2;ij}}\Delta^{uv}_y\Big(\frac{\overline{u}_{ij}}{h^u_{1;ij}}\Big) + \frac{h^{uv}_{2;ij}}{h^{uv}_{1;ij}}\Delta^{uv}_x\Big(\frac{\overline{v}_{ij}}{h^v_{2;ij}}\Big)\Big] \tag{5.201}$$

- alongstream diffusion in the V-equation (4.87) at the V-node

$$\overline{\mathcal{D}}_{mh2}(\overline{\tau_{22}})^v_{ij} = \frac{h^c_{1;ij}\overline{D}^c_{22;ij} - h^c_{2;i,j-1}\overline{D}^c_{22;i,j-1}}{(h^v_{1;ij})^2 h^v_{2;ij}} \tag{5.202}$$

$$\overline{D}^c_{22;ij} = \overline{\nu_H}^c_{ij} h^c_{1;ij}\Big[\frac{h^c_{1;ij}}{h^c_{2;ij}}\Delta^c_y\Big(\frac{\overline{v}_{ij}}{h^v_{1;ij}}\Big) - \frac{h^c_{2;ij}}{h^c_{1;ij}}\Delta^c_x\Big(\frac{\overline{u}_{ij}}{h^u_{2;ij}}\Big)\Big] \tag{5.203}$$

5.3.10 Integrals of the baroclinic diffusion terms

The discretised versions of the diffusion integrals in the 2-D momentum equations are given by

$$\overline{\delta D}^u_{h1;ij} = \sum_{k=1}^{N_z}\Big(\mathcal{D}_{mh1}(\tau_{11})^u_{ijk} + \mathcal{D}_{mh2}(\tau_{12})^u_{ijk}\Big)h^u_{3;ijk} - \overline{\mathcal{D}}_{mh1}(\overline{\tau_{11}})^u_{ij} - \overline{\mathcal{D}}_{mh2}(\overline{\tau_{12}})^u_{ij} \tag{5.204}$$

$$\overline{\delta D}^v_{h2;ij} = \sum_{k=1}^{N_z}\Big(\mathcal{D}_{mh1}(\tau_{21})^v_{ijk} + \mathcal{D}_{mh2}(\tau_{22})^v_{ijk}\Big)h^v_{3;ijk} - \overline{\mathcal{D}}_{mh1}(\overline{\tau_{21}})^v_{ij} - \overline{\mathcal{D}}_{mh2}(\overline{\tau_{22}})^v_{ij} \tag{5.205}$$

5.3.11 Discretisation of vertical diffusion

The vertical diffusion terms in the 3-D momentum equations are written as the divergence of the vertical fluxes D_{13}, D_{23}, defined in Table 5.5:

$$\mathcal{D}_{mv}(u) = \frac{1}{h_3}\frac{\partial}{\partial s}\left(\frac{\nu_T}{h_3}\frac{\partial u}{\partial s}\right) = \frac{1}{h_3}\frac{\partial D_{13}}{\partial s} \tag{5.206}$$

$$\mathcal{D}_{mv}(v) = \frac{1}{h_3}\frac{\partial}{\partial s}\left(\frac{\nu_T}{h_3}\frac{\partial v}{\partial s}\right) = \frac{1}{h_3}\frac{\partial D_{23}}{\partial s} \tag{5.207}$$

$$\tag{5.208}$$

- The vertical diffusion term in the u-equation (4.61) is obtained by differencing the flux D^{uw}_{13} at the U-node

$$\mathcal{D}_{mv}(u)^u_{ijk} = \frac{D^{uw}_{13;ij,k+1} - D^{uw}_{13;ijk}}{h^u_{3;ijk}} \tag{5.209}$$

  The flux is calculated from

$$D^{uw}_{13;ijk} = \frac{\nu^{uw}_{T;ijk}}{h^{uw}_{3;ijk}} \Delta^u_z u_{ijk} \tag{5.210}$$

- The vertical diffusion term in the v-equation (4.62) is obtained by differencing the flux D^{vw}_{23} at the V-node

$$\mathcal{D}_{mv}(v)^v_{ijk} = \frac{D^{vw}_{23;ij,k+1} - D^{vw}_{23;ijk}}{h^v_{3;ijk}} \tag{5.211}$$

  The flux is calculated from

$$D^{vw}_{23;ijk} = \frac{\nu^{vw}_{T;ijk}}{h^{vw}_{3;ijk}} \Delta^v_z v_{ijk} \tag{5.212}$$

5.3.12 Diffusion coefficients for momentum

5.3.12.1 horizontal diffusion coefficients

This section describes the discretisation of the horizontal diffusion coefficients for the case that a Smagorinsky scheme has been selected. Firstly, the horizontal tension and shearing are calculated at their "natural" node

$$D^c_{T;ijk} = \frac{h^c_{2;ij}}{h^c_{1;ij}} \Delta^c_x \Big(\frac{u_{ijk}}{h^u_{2;ij}} \Big) - \frac{h^c_{1;ij}}{h^c_{2;ij}} \Delta^c_y \Big(\frac{v_{ijk}}{h^v_{1;ij}} \Big) \tag{5.213}$$

$$D^{uv}_{S;ijk} = \frac{h^{uv}_{1;ij}}{h^{uv}_{2;ij}} \Delta^{uv}_y \Big(\frac{u_{ijk}}{h^u_{1;ij}} \Big) + \frac{h^{uv}_{2;ij}}{h^{uv}_{1;ij}} \Delta^{uv}_x \Big(\frac{v_{ijk}}{h^v_{2;ij}} \Big) \tag{5.214}$$

The discretised values of the horizontal diffusion coefficient at the C- and corner nodes are obtained by applying (4.52)

$$\nu^c_{H;ijk} = C_m h^c_{1;ij} h^c_{2;ij} \sqrt{\Big(D^c_{T;ijk} \Big)^2 + \Big(D^c_{S;ijk} \Big)^2} \tag{5.215}$$

$$\nu^{uv}_{H;ijk} = C_m h^{uv}_{1;ij} h^{uv}_{2;ij} \sqrt{\Big(D^{uv}_{T;ijk} \Big)^2 + \Big(D^{uv}_{S;ijk} \Big)^2} \tag{5.216}$$

Note that the (5.215) and (5.216) only require the interpolation of either D_S at the C-node or D_T at the UV-node but not both.

The 2-D coefficients are obtained by vertical integration

$$\overline{\nu_H}^c_{ij} = \sum_{k=1}^{N_z} \nu^c_{H;ijk} h^c_{3;ijk} \tag{5.217}$$

$$\overline{\nu_H}^{uv}_{ij} = \sum_{k=1}^{N_z} \nu^{uv}_{H;ijk} h^{uv}_{3;ijk} \tag{5.218}$$

5.3.12.2 vertical diffusion coefficient

The vertical diffusion coefficient for momentum ν_T is obtained from one of the available turbulence schemes, described in Section 4.4. Values are first stored at the W-nodes and interpolated afterwards at the UW- and VW-nodes for the calculation of the vertical diffusion fluxes in the momentum equations. The evaluation of ν_T only involves algebraic expressions so that the discretisation procedure is straightforward.

The following comments are to be given

- To avoid spurious numerical oscillations the squared buoyancy frequency N^2 is spatially discretised by averaging over the neighbouring cells in the horizontal:

$$\begin{aligned}\left(N^w_{ijk}\right)^2 = & \Big[2w_{ij}(\tilde{N}_{ijk})^2 + w_{i-1,j}(\tilde{N}_{i-1,jk})^2 + w_{i+1,j}(\tilde{N}_{i+1,jk})^2 \\ & w_{i,j-1}(\tilde{N}_{i,j-1,k})^2 + w_{i,j+1}(\tilde{N}_{i,j+1,k})^2\Big] \\ & \Big[2w_{ij} + w_{i-1,j} + w_{i+1,j} + w_{i,j-1} + w_{i,j+1}\Big]^{-1}\end{aligned} \tag{5.219}$$

  where

$$\left(\tilde{N}_{ijk}\right)^2 = \frac{\beta^w_{T;ijk}(T^c_{ijk} - T^c_{ij,k-1}) - \beta^w_{S;ijk}(S^c_{ijk} - S^c_{ij,k-1})}{h^w_{3;ijk}} \tag{5.220}$$

  is the unfiltered value of N^2 and w_{ij} equals 0 on land and 1 on sea cells. The expansion coefficients β_T and β_S are first obtained from the equation of state at the C-node and then interpolated at the W-nodes.

- The squared shear frequency M^2 is discretised using

$$\left(M^w_{ijk}\right)^2 = \frac{(u^c_{ijk} - u^c_{ij,k-1})^2 + (v^c_{ijk} - v^c_{ij,k-1})^2}{(h^w_{3;ijk})^2} \tag{5.221}$$

Note that the currents are interpolated first at the C-nodes before the vertical derivative is taken.

- The Richardson number is obtained using its definition $Ri = N^2/M^2$. An upper limit of 1000 is imposed to prevent division by zero if $N^2 \gg M^2$.

- If the vertical diffusion coefficient is derived from a RANS model (see Section 4.4.3), its value is not known at the surface and the bottom. Its evaluation involves the turbulent parameters k, ε or l which are obtained from algebraic relations or by solving additional transport equations. This is further discussed in Section 5.6.

5.3.13 Discretisation of the baroclinic pressure gradient

A known problem is the numerical treatment of the baroclinic pressure gradient for σ-coordinate models. Two types of errors may occur

- The two terms on the right hand side of (4.74) may have the same magnitude and different signs. Significant rounding errors may arise, especially in case of large bathymetric gradients.

- Violation of the hydrostatic consistency condition which states that a σ-surface immediately below (above) a given σ-surface remains below (above) the given σ-surface within a horizontal distance of one grid interval

$$\left|\frac{\sigma}{H}\frac{\partial H}{\partial x_i}\right|\Delta x_i < \Delta\sigma \tag{5.222}$$

  where Δx_i is the grid resolution in the X- or Y-direction and $\Delta\sigma$ the vertical resolution in σ-space.

For further discussion and examples are found in Haney (1991); Kliem & Pietrzak (1999).

Several solutions have been proposed: fourth order (McCalpin, 1994) or sixth order (Chu & Fan, 1997) discretisations, "z"-level based methods (Beckmann & Haidvogel, 1993; Stelling & Van Kester, 1994; Slørdal, 1997), second order method using unequal weighting (Song, 1998), cubic polynomial interpolation using harmonic averaging (Shchepetkin & McWilliams, 2003). Three algorithms are implemented in the code: the "traditional" second-order discretisation, a simple z-level method and the Shchepetkin & McWilliams (2003) approach.

Before discussing the implemented algorithms, the baroclinic pressure gradient is rewritten in a more convenient form. In Cartesian coordinates, the last term on the right of (4.56) can be written as

$$\begin{aligned} -\frac{\partial q}{\partial x_i} &= -g\frac{\partial}{\partial x_i}\int_z^\zeta \Big(\frac{\rho}{\rho_0}-1\Big)\,\mathrm{d}z' \\ &= -g\int_z^\zeta \frac{\partial}{\partial x_i}\Big(\frac{\rho}{\rho_0}-1\Big)\,\mathrm{d}z' - g\Big(\frac{\rho_s}{\rho_0}-1\Big)\frac{\partial \zeta}{\partial x_i} \\ &\simeq -\frac{g}{\rho_0}\int_z^\zeta \frac{\partial \rho}{\partial x_i}\,\mathrm{d}z' \\ &\simeq g\int_z^\zeta \Big(\beta_T\frac{\partial T}{\partial x_i}-\beta_S\frac{\partial S}{\partial x_i}\Big)\,\mathrm{d}z' \end{aligned} \tag{5.223}$$

where use is made of the Boussinesq approximaion. The last line is obtained by applying the equation of state, taking account that T represents potential and not *in situ* temperature. The latter approximation is at least valid for non-oceanic waters.

In transformed coordinates, equation (5.223) becomes

$$\begin{aligned} F_i^b &= g\int_z^\zeta \beta_T\Big(\frac{\partial T}{\partial x_i}\Big|_\sigma - \frac{\partial T}{\partial z'}\frac{\partial z'}{\partial x_i}\Big|_\sigma\Big)\,\mathrm{d}z' - g\int_z^\zeta \beta_S\Big(\frac{\partial S}{\partial x_i}\Big|_\sigma - \frac{\partial S}{\partial z'}\frac{\partial z'}{\partial x_i}\Big|_\sigma\Big)\,\mathrm{d}z' \\ &= F_i^T + F_i^S \end{aligned} \tag{5.224}$$

where a $\Big|_\sigma$ means a derivative along a surface of constant σ.

The implemented discretisations for F_1^T are discussed below. Algorithms for the salinity and Y-component are obtained in a similar way.

5.3.13.1 second-order method

The scheme uses a straightforward discretisation of (5.224)

$$\begin{aligned} F^T_{ijN_z} &= \frac{g^u_{ij}}{2h^u_{1;ij}}\beta^u_{T;ijN_z}\Delta^u_x(T_{ijN_z})h^u_{3;ijN_z} \\ F^T_{ij,k-1} &= F^T_{ijk} + \frac{g^u_{ij}}{h^u_{1;ij}}\beta^{uw}_{T;ijk}\Big(h^{uw}_{3;ijk}\Delta^{uw}_x(T^w_{ijk}) - \Delta^{uw}_z(T^u_{ijk})\Delta^{uw}_x(z^w_{ijk})\Big) \end{aligned} \tag{5.225}$$

for $N_z \geq k \geq 2$.

5.3.13.2 z-level method

The z-level scheme evaluates the horizontal gradient by vertically interpolating the values of T at the C-node to the vertical level of the U-node (see Figure 5.6). If z^u_{ijk} denotes the "physical" z-level of the U-node where the gradient has to be evaluated, the X-component of the gradient is obtained by vertically interpolating the C-node temperature values in the adjacent columns (i-1,j) and (i,j) to the vertical level of the U-node. Denoting this values by respectively T^L_{ijk} and T^R_{ijk}, the pressure gradient is readily evaluated using (5.223):

$$\begin{aligned} F^T_{ijN_z} &= \frac{g^u_{ij}}{2h^u_{1;ij}}\beta^u_{T;ijN_z}(T^R_{ijk}-T^L_{ijk})h^u_{3;ijN_z} \\ F^T_{ij,k-1} &= F^T_{ijk}+\frac{g^u_{ij}}{h^u_{1;ij}}\beta^{uw}_{T;ijk}(T^R_{ijk}-T^L_{ijk})h^{uw}_{3;ijk} \end{aligned} \tag{5.226}$$

The following restrictions apply:

- If the vertical position of T^L_{ijk} or T^R_{ijk} is below the lowest grid point, its value is set to T_{ij1}.
- If the vertical position of T^L_{ijk} or T^R_{ijk} is above the highest grid point, its value is set to T_{ijN_z}.

Despite its simplicity and the fact that it avoids the truncation problem of the σ-grid, the scheme may produce unrealistic results near the bottom (surface) or adjacent to a sloping boundary.

5.3.13.3 cube-H method

The "cube-H" algorithm is probably the most robust scheme, but at the expense of a larger CPU time. The method uses a cubic spline formalism together with harmonic averaging. Details of the scheme are not given here but can be found in the paper of Shchepetkin & McWilliams (2003).

Omitting the j-index for simplicity, the following procedure is taken

1. Evaluate the integral at the W-nodes

$$FW_{ik}=\int_{\sigma^c_{ik}}^{\sigma^c_{i,k+1}} T\frac{\partial z}{\partial \sigma}\,\mathrm{d}\sigma \tag{5.227}$$

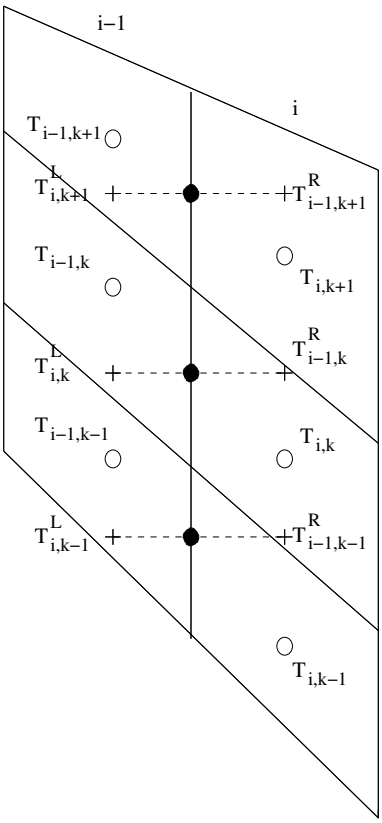

Figure 5.6: Illustration of the z-level interpolation scheme. C- and U-nodes are represented by empty, respectively solid circles.

giving

$$FW_{i,N_z+1} = \frac{1}{2} T_{iN_z} h^c_{3;iN_z} \tag{5.228}$$

$$\begin{aligned} FW_{ik} &= \frac{1}{2}(T_{i,k-1} + T_{ik})(z^c_{ik} - z^c_{i,k-1}) \\ &\quad - \frac{1}{10}\Big[(d_z T_{ik} - d_z T_{i,k-1})\Big(z^c_{ik} - z^c_{i,k-1} - \frac{1}{12}(d_z z^c_{i,k-1} + d_z z^c_{ik})\Big) \\ &\quad - (d_z z^c_{ik} - d_z z^c_{i,k-1})\Big(T_{ik} - T_{i,k-1} - \frac{1}{12}(d_z T_{i,k-1} + d_z T_{ik})\Big)\Big] \end{aligned} \tag{5.229}$$

where $2 \le k \le N_z$.

2. Evaluate the integral at the UW-nodes

$$FU_{ik} = \int_{\xi^c_{1;ik}}^{\xi^c_{1;i+1,k}} T \frac{\partial z}{\partial \xi_1} \, \mathrm{d}\xi_1 \tag{5.230}$$

giving

$$FU_{i,N_z+1} = 0 \tag{5.231}$$

$$\begin{aligned} FU_{ik} \;=\; & \frac{1}{2}(T_{i-1,k} + T_{ik})(z^c_{ik} - z^c_{i-1,k}) \\ & - \frac{1}{10}\Big[(d_x T_{ik} - d_x T_{i-1,k})\Big(z^c_{ik} - z^c_{i-1,k} - \frac{1}{12}(d_x z^c_{i-1,k} + d_x z^c_{ik})\Big) \\ & - (d_x z^c_{ik} - d_x z^c_{i-1,k})\Big(T_{ik} - T_{i-1,k} - \frac{1}{12}(d_x T_{i-1,k} + d_x T_{ik})\Big)\Big] \end{aligned} \tag{5.232}$$

where $2 \le k \le N_z$.

3. The algorithms for the derivatives in (5.229) and (5.232) are given by:

   - Vertical derivatives

$$\begin{aligned} & d_z f_{ik} = \frac{2(f_{i,k+1} - f_{ik})(f_{ik} - f_{i,k-1})}{f_{i,k+1} - f_{i,k-1}} && \text{if} \quad (f_{i,k+1} - f_{ik})(f_{ik} - f_{i,k-1}) > 0 \\ & d_z f_{ik} = 0 && \text{otherwise} \end{aligned} \tag{5.233}$$

   Values at the boundaries are

$$\begin{aligned} d_z f_{i1} &= \frac{6}{5}(f_{i2} - f_{i1}) - \frac{7}{15} d_z f_{i2} \\ d_z f_{iN_z} &= \frac{6}{5}(f_{iN_z} - f_{i,N_z-1}) - \frac{7}{15} d_z f_{i,N_z-1} \end{aligned} \tag{5.234}$$

   - Horizontal derivatives

$$\begin{aligned} & d_x f_{ik} = \frac{2(f_{i+1,k} - f_{ik})(f_{ik} - f_{i-1,k})}{f_{i+1,k} - f_{i-1,k}} && \text{if} \quad (f_{i+1,k} - f_{ik})(f_{ik} - f_{i-1,k}) > 0 \\ & d_x f_{ik} = 0 && \text{otherwise} \end{aligned} \tag{5.235}$$

   Boundary conditions are

$$\begin{aligned} i-1 \text{ dry}, \, i{+}1 \text{ wet} \;&:\; d_x f_{ik} = \tfrac{6}{5}(f_{i+1,k} - f_{ik}) - \tfrac{7}{15} d_x f_{i+1,k} \\ i-1 \text{ wet}, \, i{+}1 \text{ dry} \;&:\; d_x f_{ik} = \tfrac{6}{5}(f_{ik} - f_{i-1,k}) - \tfrac{7}{15} d_x f_{i-1,k} \\ i-1 \text{ dry}, \, i{+}1 \text{ dry} \;&:\; d_x f_{ik} = 0 \end{aligned} \tag{5.236}$$

4. Evaluate the "temperature Jacobian" at the UW-node

$$\mathcal{I}(T,z) = \frac{1}{h_1}\frac{\partial T}{\partial \xi_1}\Big|_\sigma \frac{\partial z}{\partial \sigma} - \frac{1}{h_1}\frac{\partial T}{\partial \sigma}\frac{\partial z}{\partial \xi_1}\Big|_\sigma \tag{5.237}$$

giving

$$\mathcal{I}^{uw}_{ik} = \frac{g^u_i \beta^{uw}_{T;ik}}{h^u_{1;i}}(FW_{ik} - FW_{i-1,k} - FU_{ik} + FU_{i,k-1}) \tag{5.238}$$

for $2 \le k \le N_z$.

5. Evaluate the baroclinic gradient

$$F^T_{ik} = \sum_{k'=k+1}^{N_z+1} \mathcal{I}^{uw}_{ik'} \tag{5.239}$$

for $1 \le k \le N_z$, or

$$\begin{aligned} F^T_{iN_z} &= \frac{g^u_i}{2h^u_{1;i}}\beta^u_{T;iN_z}\Delta^u_x(T_{iN_z})h^u_{3;iN_z} \\ F^T_{i,k-1} &= F^T_{ik} + \mathcal{I}_{ik} \end{aligned} \tag{5.240}$$

for $2 \le k \le N_z$.

5.3.14 Tidal force

If the astronomical tidal force is included in the momentum equations[4], the components of the force are updated at each internal (3-D) time step. Tidal constituents for the harmonic expansion are selected by the user.

The procedure is the following:

- The astronomical Greenwich arguments and nodal factors are updated using the expressions given in Section 4.5 if requested. Otherwise

$$\begin{aligned} f_{qn}(t^{m+1}) &\simeq f_{qn}(t^m) \\ P^{m+1}_{qn} &\simeq P^m_{qn} + \Delta\tau\omega_{qn} \end{aligned} \tag{5.241}$$

  where $P^m_{qn} = V_{qn}(t^m) + u_{qn}(t^m)$ and ω_{qn} are the frequencies of the tidal constituents[5][6].

[4]The astronomical tidal force can only be taken into account if a spherical grid is selected.

[5]Time is converted to GMT if necessary.

[6]Equation (5.241) is applied in double precision arithmetic to avoid truncation errors.

- The tidal amplitudes A_{qn}^{m+1} are calculated using (4.232).
- The components of the tidal force are determined using (4.231) and (4.230)

$$\begin{aligned}
F_{1;ij}^{t;u} = -\frac{g_{ij}^u}{h_{1;ij}^u}\Big[&\frac{3}{2}\sin(2\phi_{ij}^u)\Delta_x^u(\phi_{ij}^c)\sum_{n=1}^{N_0}A_{0n}\cos(P_{0n;ij}) \\
&+\sin(2\phi_{ij}^u)\Delta_x^u(\lambda_{ij}^c)\sum_{n=1}^{N_1}A_{1n}\sin(\lambda_{ij}^u+P_{1n;ij}) \\
&-2\cos(2\phi_{ij}^u)\Delta_x^u(\phi_{ij}^c)\sum_{n=1}^{N_1}A_{1n}\cos(\lambda_{ij}^u+P_{1n;ij}) \\
&+(1+\cos(2\phi_{ij}^u))\Delta_x^u(\lambda_{ij}^c)\sum_{n=1}^{N_2}A_{2n}\sin(2\lambda_{ij}^u+P_{2n;ij}) \\
&+\sin(2\phi_{ij}^u)\Delta_x^u(\phi_{ij}^c)\sum_{n=1}^{N_2}A_{2n}\cos(2\lambda_{ij}^u+P_{2n;ij}) \\
&+\frac{3}{4}(3\cos\phi_{ij}^u+\cos(3\phi_{ij}^u))\Delta_x^u(\lambda_{ij}^c)\sum_{n=1}^{N_3}A_{3n}\sin(3\lambda_{ij}^u+P_{3n;ij}) \\
&+\frac{3}{4}(\sin\phi_{ij}^u+\sin(3\phi_{ij}^u))\Delta_x^u(\phi_{ij}^c)\sum_{n=1}^{N_3}A_{3n}\cos(3\lambda_{ij}^u+P_{3n;ij})\Big]
\end{aligned} \tag{5.242}$$

and

$$\begin{aligned}
F_{2;ij}^{t;v} = -\frac{g_{ij}^v}{h_{2;ij}^v}\Big[&\frac{3}{2}\sin(2\phi_{ij}^v)\Delta_y^v(\phi_{ij}^c)\sum_{n=1}^{N_0}A_{0n}\cos(P_{0n;ij}) \\
&+\sin(2\phi_{ij}^v)\Delta_y^v(\lambda_{ij}^c)\sum_{n=1}^{N_1}A_{1n}\sin(\lambda_{ij}^v+P_{1n;ij}) \\
&-2\cos(2\phi_{ij}^v)\Delta_y^v(\phi_{ij}^c)\sum_{n=1}^{N_1}A_{1n}\cos(\lambda_{ij}^v+P_{1n;ij}) \\
&+(1+\cos(2\phi_{ij}^v))\Delta_y^v(\lambda_{ij}^c)\sum_{n=1}^{N_2}A_{2n}\sin(2\lambda_{ij}^v+P_{2n;ij}) \\
&+\sin(2\phi_{ij}^v)\Delta_y^v(\phi_{ij}^c)\sum_{n=1}^{N_2}A_{2n}\cos(2\lambda_{ij}^v+P_{2n;ij})
\end{aligned}$$

$$+\frac{3}{4}(3\cos\phi_{ij}^{v}+\cos(3\phi_{ij}^{v}))\Delta_{y}^{v}(\lambda_{ij}^{c})\sum_{n=1}^{N_3}A_{3n}\sin(3\lambda_{ij}^{v}+P_{3n;ij})$$
$$+\frac{3}{4}(\sin\phi_{ij}^{v}+\sin(3\phi_{ij}^{v}))\Delta_{y}^{v}(\phi_{ij}^{c})\sum_{n=1}^{N_3}A_{3n}\cos(3\lambda_{ij}^{v}+P_{3n;ij})\Big] \tag{5.243}$$

5.3.15 Surface and bottom boundary conditions

5.3.15.1 surface boundary conditions

Application of the surface boundary conditions (4.266) and (4.265) gives

$$F_{13;ij,N_z+1}=0\,,\quad F_{23;ij,N_z+1}=0 \tag{5.244}$$

and

$$D_{13;ij,N_z+1}=\tau_{s1;ij}^{u}\,,\quad D_{23;ij,N_z+1}=\tau_{s2;ij}^{v} \tag{5.245}$$

for respectively the vertical advective and diffusive fluxes of momentum at the surface.

The components of the surface stress are calculated as function of meteorological variables. Different options are available and discussed in Section 4.8.

The following steps are taken

- The meteorological data are interpolated at the C-nodes of the model grid.
- The surface drag coefficient C_{ds} and the components of the surface stress are determined at the C-nodes.
- The stress components are interpolated at respectively the U- and V-node.

In the case that the surface drag and exchange are calculated using the Monin-Obukhov formulation, described in Section 4.8.3, the system of four equations (4.327), (4.330), (4.331) and (4.332) needs to solved at each model grid point and time step using a time-consuming iteration procedure. The following simplifying procedure is taken in the program

- The equations generally depend on five meteorological variables: the magnitude of the surface wind W, air temperature T_a, sea surface temperature T_s, relative humidity RH and atmospheric pressure P_a. The following simplications are made

- The atmospheric pressure enters the equations only indirectly to evaluate the surface humidity and is taken as constant in the equations, i.e. $P_a = P_{ref}$.
- Tests showed a larger dependence on the air minus sea temperature difference ΔT than on the individual values of T_a and T_s itself. Letting $T_s = T_{s;ref}$ one has $T_a = T_{s;ref} + \Delta T$.

The equations now only depend on three variables: W, ΔT, RH

- The equations are solved for C_{ds}, C_e, C_h on a "discretised" grid at the initial time

$$\begin{aligned} W &= W_{min} + l\delta W \quad \text{for} \quad l = 0, N_W\,; \quad N_W = \frac{W_{max} - W_{min}}{\delta W} \\ \Delta T &= \Delta T_{min} + m\delta(\Delta T) \quad \text{for} \quad m = 0, N_T\,; \quad N_T = \frac{\Delta T_{max} - \Delta T_{min}}{\delta(\Delta T)} \\ RH &= RH_{min} + n\delta RH \quad \text{for} \quad n = 0, N_R\,; \quad N_R = \frac{RH_{max} - RH_{min}}{\delta R} \end{aligned} \tag{5.246}$$

The following default values are taken

$$\begin{aligned} W_{min} &= 3\,, \quad W_{max} = 50\,, \quad \delta W = 0.25 \quad (\text{m/s}) \\ \Delta T_{min} &= -5\,, \quad \Delta T_{max} = 5\,, \quad \delta(\Delta T) = 1 \quad (^0\text{C}) \\ RH_{min} &= 0.5\,, \quad RH_{max} = 1.0\,, \quad \delta R = 0.05 \end{aligned} \tag{5.247}$$

The lower limit W_{min}=3 m/s is taken since the equations diverge in the case of the free-convection limit $W \to 0$ and $\Delta T > 0$. The computed values are stored in 3-D arrays.

- The values of the drag and exchange coefficients are then obtained at a specific time by a tri-linear inperpolation from the discretised values. Extrapolation is used if necessary.

5.3.15.2 bottom boundary conditions

The bottom boundary condition for vertical advection is the same as the one applied at surface (see equation (4.345)) so that

$$F_{13;ij1} = 0\,, \quad F_{23;ij1} = 0 \tag{5.248}$$

If the bottom stress is parameterised using the bottom values of the 3-D current, vertical diffusion is treated implicitly. The flux bottom boundary conditions (4.340) are then discretised as

$$D_{13;ij1} = \theta_v k^{n;u}_{b;ij} u^{n+1}_{ij1} + (1 - \theta_v) k^{n;u}_{b;ij} u^n_{ij1} \tag{5.249}$$

$$D_{23;ij1} = \theta_v k^{n;v}_{b;ij} v^{n+1}_{ij1} + (1-\theta_v) k^{n;v}_{b;ij} v^{n}_{ij1} \tag{5.250}$$

where θ_v is the implicity factor for vertical diffusion and the friction velocities are defined by

$$\begin{array}{lll} \text{no bottom stress (4.337)} & : & k^u_{b;ij} = k^v_{b;ij} = 0 \\ \text{linear bottom stress (4.338)} & : & k^u_{b;ij} = k^v_{b;ij} = k_{lin} \\ \text{3-D quadratic law (4.340)} & : & k^u_{b;ij} = C^u_{db;ij}\left((u^n_{ij1})^2 + (v^{n;u}_{ij1})^2\right)^{1/2} \\ & : & k^v_{b;ij} = C^v_{db;ij}\left((u^{n;v}_{ij1})^2 + (v^n_{ij1})^2\right)^{1/2} \end{array} \tag{5.251}$$

If the bottom stress is expressed in terms of the depth mean current, the bottom flux is discretised explicitly

$$D_{13;ij1} = k^{n;u}_{b;ij} \overline{u}^n_{ij} \tag{5.252}$$

$$D_{23;ij1} = k^{n;v}_{b;ij} \overline{v}^n_{ij} \tag{5.253}$$

where the friction velocity is given by (5.251) in the absence of a bottom stress or a linear friction law and

$$\begin{aligned} k^u_{b;ij} &= C^u_{db;ij}\left((\overline{u}^n_{ij})^2 + (\overline{v}^{n;u}_{ij})^2\right)^{1/2} \\ k^v_{b;ij} &= C^v_{db;ij}\left((\overline{u}^{n;v}_{ij})^2 + (\overline{v}^n_{ij})^2\right)^{1/2} \end{aligned} \tag{5.254}$$

The bottom drag coefficient is discretised using (5.18)-(5.19) or specified externally.

5.3.16 Lateral boundary conditions for the 2-D mode

5.3.16.1 open boundary conditions for transports

Open boundary conditions for the 2-D mode are discussed in Section 4.10.1. The aim is to provide values of U at U- and of V at V-open boundaries. External data can be generally expressed as the sum of a non-harmonic and an harmonic part as given by (4.354). The expression is updated at each time step for the requested locations and variables (U, V or ζ) depending on the type of conditions, prior to the application of the boundary conditions itself.

The tidal constituents are selected by the user. In analogy with the astronomical tide (see Section 5.3.14), the space-independent astronomical phases are calculated by using either the expressions in Section 4.5 with time converted to GMT if needed, or the linear approximation

$$P^{m+1}_l = V^{m+1}_l + u^{m+1}_l \simeq V^m_l + u^m_l + \omega_l \delta\tau \tag{5.255}$$

where ω_l are the tidal frequencies. Note that (5.255) is evaluated in double precision to avoid truncation errors for long integration periods.

The following notations are introduced

- $\pm$ or $\mp$: The upper (lower) sign applies at a western/southern (eastern/northern) open boundary.
- $X_{i:i-1,j}$: The quantity X is evaluated at grid point (i,j) for a western and at $(i-1,j)$ for an eastern boundary.
- $Y_{i,j:j-1}$: The quantity Y is evaluated at grid point (i,j) for a southern and at $(i,j-1)$ for a northern boundary.
- U^e_{ij}, V^e_{ij}, ζ^e_{ij} denote externally specified values (harmonic or time series data).
- s_{ij} equals 1 if ζ^e_{ij} is defined at an exterior C-node and 2 if ζ^e_{ij} is defined at an open boundary U- or V-node.
- The gravity wave speed at the C-node nearest to the U- or V-open boundary location (i,j) is defined by

$$c^u_{ij} = \sqrt{g^c_{i:i-1,j} H^c_{i:i-1,j}}\,, \quad c^v_{ij} = \sqrt{g^c_{i,j:j-1} H^c_{i,j:j-1}} \tag{5.256}$$

- The following auxiliary parameters are defined at U- or V-nodes

$$\alpha^u_{ij} = \frac{\Delta\tau c^u_{ij}}{h^c_{1;i:i-1,j}}\,, \quad \alpha^v_{ij} = \frac{\Delta\tau c^v_{ij}}{h^c_{2;i,j:j-1}} \tag{5.257}$$

$$r^u_{ij} = \frac{h^c_{1;i:i-1,j}}{h^u_{1;ij}}\,, \quad r^v_{ij} = \frac{h^c_{2;i,j:j-1}}{h^v_{2;ij}} \tag{5.258}$$

The discretised versions of all available open boundary conditions are listed below using the same numbering system as in Section 4.10.1.

0. Clamped (see equation (4.357)).

$$U^{m+1}_{ij} = U^m_{ij}\,, \quad V^{m+1}_{ij} = V^m_{ij} \tag{5.259}$$

1. Zero slope (see equation (4.358)).

$$U^{m+1}_{ij} \;=\; U^m_{ij} + \Delta\tau\Big(f^u_{ij}(\theta_c V^{m+1;c}_{i:i-1,j} + (1-\theta_c) V^{m;c}_{i:i-1,j})$$

$$+ H_{ij}^{m+1;u} F_{1;ij}^{t;m+1} + \tau_{s1;ij}^{c} - \tau_{b1;ij}^{n;u}\Big) \tag{5.260}$$

$$\begin{aligned} V_{ij}^{m+1} &= V_{ij}^{m} - \Delta\tau\Big(f_{ij}^{v}(\theta_c U_{i,j:j-1}^{m+1;c} + (1-\theta_c)U_{i,j:j-1}^{m;c}) \\ &\quad - H_{ij}^{m+1;v} F_{2;ij}^{t;m+1} - \tau_{s2;ij}^{c} + \tau_{b2;ij}^{n;u}\Big) \end{aligned} \tag{5.261}$$

Note that the bottom stress components are evaluated at the old time t^n.

2. Zero volume flux (see equation (4.359)).

$$U_{ij}^{m+1} = U_{i+1:i-1,j}^{m+1}\,, \quad V_{ij}^{m+1} = V_{i,j+1:j-1}^{m+1} \tag{5.262}$$

3. Specified elevation (see (equation 4.360)).

$$\begin{aligned} U_{ij}^{m+1} &= U_{ij}^{L;m+1} \\ &= U_{ij}^{L;m} \mp s_{ij}^{u}\alpha_{ij}^{u} c_{ij}^{u} r_{ij}^{u}(\zeta_{i:i-1,j}^{m+1} - \zeta_{ij}^{e}) \\ &\quad + \Delta\tau\Big(f_{ij}^{u}(\theta_c V_{i:i-1,j}^{m+1;c} + (1-\theta_c)V_{i:i-1,j}^{m;c}) \\ &\quad + H_{ij}^{m+1;u} F_{1;ij}^{t;m+1} + \tau_{s1;ij}^{c} - \tau_{b1;ij}^{n;u}\Big) \end{aligned} \tag{5.263}$$

$$\begin{aligned} V_{ij}^{m+1} &= V_{ij}^{L;m+1} \\ &= V_{ij}^{L;m} \mp s_{ij}^{v}\alpha_{ij}^{v} c_{ij}^{v} r_{ij}^{v}(\zeta_{i,j:j-1}^{m+1} - \zeta_{ij}^{e}) \\ &\quad - \Delta\tau\Big(f_{ij}^{v}(\theta_c U_{i,j:j-1}^{m+1;c} + (1-\theta_c)V_{i,j:j-1}^{m;c}) \\ &\quad - H_{ij}^{m+1;v} F_{2;ij}^{t;m+1} - \tau_{s2;ij}^{c} + \tau_{b2;ij}^{n;v}\Big) \end{aligned} \tag{5.264}$$

4. Specified transport (see equation (4.361)).

$$U_{ij}^{m+1} = U_{ij}^{e}\,, \quad V_{ij}^{m+1} = V_{ij}^{e} \tag{5.265}$$

5. Radiation condition using shallow water speed (see equation (4.363)).

$$\begin{aligned} U_{ij}^{m+1} &= \frac{U_{ij}^{m} + \alpha_{ij}^{u} U_{i+1:i-1,j}^{m+1}}{1+\alpha_{ij}^{u}} \\ V_{ij}^{m+1} &= \frac{V_{ij}^{m} + \alpha_{ij}^{v} V_{i,j+1:j-1}^{m+1}}{1+\alpha_{ij}^{v}} \end{aligned} \tag{5.266}$$

6. Orlanski (1976) condition (see equation (4.364)).

$$U_{ij}^{m+1} = \Big(1 - O_R(r_{1;ij}^u, r_{2;ij}^u, r_{3,ij}^u)\Big) U_{ij}^m + O_R(r_{1;ij}^u, r_{2;ij}^u, r_{3,ij}^u) U_{i+1:i-1,j}^m \tag{5.267}$$

$$V_{ij}^{m+1} = \Big(1 - O_R(r_{1;ij}^v, r_{2;ij}^v, r_{3,ij}^v)\Big) V_{ij}^m + O_R(r_{1;ij}^v, r_{2;ij}^v, r_{3,ij}^v) V_{i,j+1:j-1}^m \tag{5.268}$$

The Orlanski weight function is defined by

$$\begin{array}{ll} O_R(r_1, r_2, r_3) = \min\Big(\max(\dfrac{r_1 - r_2}{r_3 - r_2}, 0), 1\Big) & \text{if } r_2 \neq r_3 \\ O_R(r_1, r_2, r_3) = 0 & \text{if } r_2 = r_3 \text{ and } r_1 \leq r_2 \\ O_R(r_1, r_2, r_3) = 1 & \text{if } r_2 = r_3 \text{ and } r_1 > r_2 \end{array} \tag{5.269}$$

The arguments of the weight functions are defined by

$$r_{1;ij}^u = U_{i+1:i-1,j}^m\,, \quad r_{2;ij}^u = U_{i+1:i-1,j}^{m-1}\,, \quad r_{3;ij}^u = U_{i+2:i-2,j}^{m-1} \tag{5.270}$$

$$r_{1;ij}^v = V_{i,j+1:j-1}^m\,, \quad r_{2;ij}^v = V_{i,j+1:j-1}^{m-1}\,, \quad r_{3;ij}^v = V_{i,j+2:j-2}^{m-1} \tag{5.271}$$

7. Camerlengo & O'Brien (1980).

$$\begin{array}{ll} U_{ij}^{m+1} = U_{i+1:i-1,j}^m & \text{if} \quad U_{i+1:i-1,j}^m \geq U_{i+2:i-2,j}^{m-1} \\ U_{ij}^{m+1} = U_{ij}^m & \text{otherwise} \end{array} \tag{5.272}$$

$$\begin{array}{ll} V_{ij}^{m+1} = V_{i,j+1:j-1}^m & \text{if} \quad V_{i,j+1:j-1}^m \geq V_{i,j+2:j-2}^{m-1} \\ V_{ij}^{m+1} = V_{ij}^m & \text{otherwise} \end{array} \tag{5.273}$$

8. Flather (1976) with specified elevation and transport (see equation (4.366)).

$$\begin{aligned} U_{ij}^{m+1} &= U_{ij}^e \mp \frac{1}{2} s_{ij}^u c_{ij}^u (\zeta_{i:i-1,j}^{m+1} - \zeta_{ij}^e) \\ V_{ij}^{m+1} &= V_{ij}^e \mp \frac{1}{2} s_{ij}^v c_{ij}^v (\zeta_{i,j:j-1}^{m+1} - \zeta_{ij}^e) \end{aligned} \tag{5.274}$$

9. Flather (1976) with specified elevation (see equation (4.367)).

$$\begin{aligned} U_{ij}^{m+1} &= U_{ij}^{L;m+1} \mp \frac{1}{2} s_{ij}^u c_{ij}^u (\zeta_{i:i-1,j}^{m+1} - \zeta_{ij}^e) \\ V_{ij}^{m+1} &= V_{ij}^{L;m+1} \mp \frac{1}{2} s_{ij}^v c_{ij}^v (\zeta_{i,j:j-1}^{m+1} - \zeta_{ij}^e) \end{aligned} \tag{5.275}$$

where U^L, V^L are the local solutions obtained from (5.263) and (5.264).

10. Røed & Smedstad (1984) (see equations (4.368)—(4.369)).

    The local solution for ζ is determined from

$$\zeta_{ij}^{L;m+1} = \zeta_{ij}^{L;m} - \frac{\Delta\tau(h^v_{1;i:i-1,j+1}V^m_{i:i-1,j+1} - h^v_{1;i:i-1,j}V^m_{i:i-1,j})}{h^c_{1;i:i-1,j}h^c_{2;i:i-1,j}} \tag{5.276}$$

    at U-nodes, and

$$\zeta_{ij}^{L;m+1} = \zeta_{ij}^{L;m} - \frac{\Delta\tau(h^u_{2;i+1,j:j-1}U^m_{i+1,j:j-1} - h^u_{2;i,j:j-1}U^m_{i,j:j-1})}{h^c_{1;i,j:j-1}h^c_{2;i,j:j-1}} \tag{5.277}$$

    at V-nodes.

    The transports U and V are obtained using

$$\begin{aligned} U_{ij}^{m+1} &= U_{ij}^{L;m+1} \mp c^u_{ij}(\zeta^{m+1}_{i:i-1,j} - \zeta^{L;m+1}_{ij}) && (5.278)\\ V_{ij}^{m+1} &= V_{ij}^{L;m+1} \mp c^v_{ij}(\zeta^{m+1}_{i,j:j-1} - \zeta^{L;m+1}_{ij}) && (5.279) \end{aligned}$$

    where the local solutions $U_{ij}^{L;m+1}$, $U_{ij}^{L;m+1}$ are given by (5.263) and (5.264).

11. Characteristic method with specified elevation and transport.

    Using the notations of Section 4.10.1 the transports are calculated as the average between the incoming (R_i) and outgoing (R_o) characteristic

$$\begin{aligned} U_{ij}^{m+1} &= \frac{1}{2}(R^{m+1;u}_{i;ij} + R^{m+1;u}_{o;ij})\\ V_{ij}^{m+1} &= \frac{1}{2}(R^{m+1;v}_{i;ij} + R^{m+1;v}_{o;ij}) \end{aligned} \tag{5.280}$$

    The incoming characteristic R_i is defined by equations (4.373) using prescribed values for transports and elevations

$$\begin{aligned} R^{m+1;u}_{i;ij} &= U^e_{ij} \pm \frac{1}{2}c^u_{ij}s^u_{ij}\Big(\zeta^e_{ij} + (2 - s^u_{ij})\zeta^{m+1}_{i:i-1,j}\Big)\\ R^{m+1;v}_{i;ij} &= V^e_{ij} \pm \frac{1}{2}c^v_{ij}s^v_{ij}\Big(\zeta^e_{ij} + (2 - s^v_{ij})\zeta^{m+1}_{i,j:j-1}\Big) \end{aligned} \tag{5.281}$$

    The outgoing characteristic R_i is obtained by solving the discretised versions of equations (4.371)–(4.372):

$$\begin{aligned} &(1 + \frac{3}{2}\alpha^u_{ij}r^u_{ij})R^{m+1;u}_{o;ij} = R^{m;u}_{o;ij}\\ &\quad + \alpha^u_{ij}r^u_{ij}(U^{m+1}_{i+1:i-1,j} + \frac{1}{2}R^{m+1;u}_{i;ij} \mp 2c^u_{ij}\zeta^{m+1}_{i:i-1,j}) \end{aligned}$$

$$
\begin{aligned}
&+\frac{\alpha^u_{ij}}{h^c_{2;i:i-1,j}}\Big(\pm(h^v_{1;i:i-1,j+1}V^m_{i:i-1,j+1}-h^v_{1;i:i-1,j}V^m_{i:i-1,j})\\
&+U^{m;c}_{i:i-1,j}(h^u_{2;i+1:i-1,j}-h^u_{2;ij})\Big)\\
&+\Delta\tau\Big(f^u_{ij}(\theta_cV^{m+1;c}_{i:i-1,j}+(1-\theta_c)V^{m;c}_{i:i-1,j})+H^{m+1;u}_{ij}F^{t;m+1}_{1;ij}+\tau^c_{s1;ij}-\tau^{n;u}_{b1;ij}\Big)
\end{aligned}
\tag{5.282}
$$

and

$$
\begin{aligned}
&(1+\frac{3}{2}\alpha^v_{ij}r^v_{ij})R^{m+1;v}_{o;ij}=R^{m;v}_{o;ij}\\
&+\alpha^v_{ij}r^v_{ij}(V^{m+1}_{i,j+1:j-1}+\frac{1}{2}R^{m+1;v}_{i;ij}\mp 2c^v_{ij}\zeta^{m+1}_{i,j:j-1})\\
&+\frac{\alpha^v_{ij}}{h^c_{1;i,j:j-1}}\Big(\pm(h^u_{2;i+1,j:j-1}U^m_{i+1,j:j-1}-h^u_{2;i,j:j-1}U^m_{i,j:j-1})\\
&+V^{m;c}_{i,j:j-1}(h^v_{1;i,j+1:j-1}-h^v_{1;ij})\Big)\\
&-\Delta\tau\Big(f^v_{ij}(\theta_cU^{m+1;c}_{i,j:j-1}+(1-\theta_c)U^{m;c}_{i,j:j-1})-H^{m+1;v}_{ij}F^{t;m+1}_{2;ij}-\tau^c_{s2;ij}+\tau^{n;v}_{b2;ij}\Big)
\end{aligned}
\tag{5.283}
$$

Note that the propagation term is integrated fully implicitly and U^{m+1}_{ij}, V^{m+1}_{ij} have been eliminated in (5.282)–(5.283) by substituting for $R^{m+1;u}_i$ and $R^{m+1;v}_i$ from (5.281).

12. Characteristic method with specified elevation.

    The method is as previous with U^e_{ij}, V^e_{ij} replaced by the local solutions $U^{L;m+1}_{ij}$, $V^{L;m+1}_{ij}$ from (5.263) and (5.264).

13. Characteristic method using zero normal gradient.

    The method is the same as previous except that the incoming characteristics are obtained as solutions of the discretised versions of equations (4.375)–(4.376):

$$
\begin{aligned}
&R^{m+1;u}_{i;ij}=R^{m;u}_{i;ij}\\
&-\frac{\alpha^u_{ij}}{h^c_{2;i:i-1,j}}\Big(\pm(h^v_{1;i:i-1,j+1}V^m_{i:i-1,j+1}-h^v_{1;i:i-1,j}V^m_{i:i-1,j})\\
&+U^{c;m}_{i:i-1,j}(h^u_{2;i+1:i-1,j}-h^u_{2;ij})\Big)\\
&+\Delta\tau\Big(f^u_{ij}(\theta_cV^{m+1;c}_{i:i-1,j}+(1-\theta_c)V^{m;c}_{i:i-1,j})+H^{m+1;u}_{ij}F^{t;m+1}_{1;ij}+\tau^c_{s1;ij}-\tau^{n;u}_{b1;ij}\Big)
\end{aligned}
$$

(5.284)

and

$$R^{m+1;v}_{i;ij} = R^{m;v}_{i;ij} - \frac{\alpha^v_{ij}}{h^c_{1;i,j:j-1}}\Big(\pm(h^u_{2;i+1,j:j-1}U^m_{i+1,j:j-1} - h^u_{2;i,j:j-1}U^m_{i,j:j-1}) + V^{c;m}_{i,j:j-1}(h^v_{1;i,j+1:j-1} - h^v_{1;ij})\Big) - \Delta\tau\Big(f^v_{ij}(\theta_c U^{m+1;c}_{i,j:j-1} + (1-\theta_c)U^{m;c}_{i,j:j-1}) - H^{m+1;v}_{ij}F^{t;m+1}_{2;ij} - \tau^c_{s2;ij} + \tau^{n;v}_{b2;ij}\Big) \quad (5.285)$$

5.3.16.2 open boundary conditions for 2-D advective and diffusive fluxes

Two schemes are available to evaluate the cross-stream advective fluxes in the U-equation at Y-open boundaries or in the V-equation at X-open boundaries

1. The first one uses a zero gradient condition

$$\overline{F}^{uv}_{12;ij} = \overline{F}^{uv}_{12;i,j+1:j-1} \quad \text{or} \quad \overline{F}^{uv}_{21;ij} = \overline{F}^{uv}_{21;i+1:i-1,j} \quad (5.286)$$

   which is the same as before.

2. The flux is determined using the upwind scheme (where possible). This means that

$$\begin{aligned} \overline{F}^{uv}_{12;ij} &= \frac{1}{2}\overline{v}^{uv}_{ij}\Big((1+s_{ij})U_{i,j-1:j} + (1-s_{ij})U_{i,j:j-1}\Big) \quad \text{or} \\ \overline{F}^{uv}_{21;ij} &= \frac{1}{2}\overline{u}^{uv}_{ij}\Big((1+s_{ij})V_{i-1:i,j} + (1-s_{ij})V_{i:i-1,j}\Big) \end{aligned} \quad (5.287)$$

   where $s_{ij} = 1$ in case of an inflow condition and either

   - (i,j-1:j) is a U-open boundary or (i-1:i,j) is a V-open boundary
   - (i-1,j) is a closed (land or coastal) V-node or (i,j-1) is a closed (land or coastal) U-node
   - (i,j) is a closed V-node or (i,j) is a closed U-node.

   In all other cases, $s_{ij} = -1$.

The cross-stream diffusive fluxes in the U-equation are avaluated as follows

- If either (i,j-1:j) is a U-open boundary, or (i-1,j) is a closed (land or coastal) V-node, or (i,j) is a closed V-node, the flux is calculated by equation (5.199) for an internal node.

- Otherwise, if (i,j-1:j) is an interior U-node, a zero gradient condition is used

$$\overline{D}^{uv}_{12;ij} = \overline{D}^{uv}_{12;i,j+1:j-1} \tag{5.288}$$

- Otherwise, the flux is set to zero, i.e.

$$\overline{D}^{uv}_{12;ij} = 0 \tag{5.289}$$

Likewise, at the fluxes in the V-equation are given by

- If either (i-1:i,j) is a V-open boundary, or (i,j-1) is a closed (land or coastal) U-node, or (i,j) is a closed U-node, the flux is calculated by equation (5.201) for an internal node.

- Otherwise, if (i-1:i,j) is an interior V-node, a zero gradient condition is used

$$\overline{D}^{uv}_{21;ij} = \overline{D}^{uv}_{21;i,j+1:j-1} \tag{5.290}$$

- Otherwise, the flux is set to zero, i.e.

$$\overline{D}^{uv}_{21;ij} = 0 \tag{5.291}$$

An optional relaxation scheme has been implemented which reduces the impact of advection within a user-defined distance from the open boundaries. In that case, the advective terms are multiplied by the relaxation factor

$$\alpha_{or} = \min(d/d_{max}, 1) \tag{5.292}$$

where d is the distance to the nearest open boundary. Experiments showed that, with an appropriate choice of the maximum relaxation distance d_{max}, instabilities, due to inaccuracies at the open boundaries, are prevented to propagate into the domain. The scheme has shown to be useful, in particular, to reduce instabilities, observed near ragged open boundaries.

5.3.16.3 boundary conditions at closed lateral boundaries

Following (4.392)–(4.393) one has

$$U_{ij} = 0\,, \quad V_{ij} = 0 \tag{5.293}$$

at coastal boundaries.

Likewise all fluxes for the cross-advective and diffusive terms are set to zero at closed Y- or X-node boundaries:

$$\overline{F}^{uv}_{12;ij} = 0.0\,, \quad \overline{D}^{uv}_{12;ij} = 0.0\,, \quad \overline{F}^{uv}_{21;ij} = 0.0\,, \quad \overline{D}^{uv}_{21;ij} = 0.0 \tag{5.294}$$

where a Y- or X-node grid point is called "closed" if one of the neighbouring V- or U-node points in the X- or Y-direction is a closed velocity node.

5.3.17 Lateral boundary conditions for the 3-D currents

5.3.17.1 open boundary conditions for horizontal 3-D currents

Open boundary conditions for the 3-D mode are discussed in Section 4.10.2.1. The aim is to provide values of u at U- and of v at V-open boundaries for each vertical level. The depth-mean parts of the currents are already determined by the 2-D open boundary conditions which means than only the baroclinic parts δu and δv need to be specified.

The discretised versions of all open available open boundary conditions are listed below using the same numbering system as in Section 4.10.2.1.

0. Zero gradient condition (see equation (4.377)).

$$\delta u^{n+1}_{ijk} = \frac{h^u_{2;i+1:i-1,j} h^{n+1;u}_{3;i+1:i-1,jk}}{h^u_{2;ij} h^{n+1;u}_{3;ijk}} \delta u^{n+1;u}_{i+1:i-1,jk} \tag{5.295}$$

$$\delta v^{n+1}_{ijk} = \frac{h^v_{1;i,j+1:j-1} h^{n+1;v}_{3;i,j+1:j-1,k}}{h^v_{1;ij} h^{n+1;v}_{3;ijk}} \delta v^{n+1;v}_{i,j+1:j-1,k} \tag{5.296}$$

This is the default condition.

1. Specified external profile (see equation (4.378)).

$$\delta u^{n+1}_{ijk} = \delta u^e_{ijk} \tag{5.297}$$

$$\delta v^{n+1}_{ijk} = \delta v^e_{ijk} \tag{5.298}$$

$$\tag{5.299}$$

2. Second order gradient condition (see equation (4.379))

$$\delta u_i = \frac{h^u_{2;i+1:i-1}}{h^u_{2;i}} \frac{h^u_{3;i+1:i-1}}{h^u_{3;i}} \Big(1 + \frac{h^c_{1;i:i-1}}{h^c_{1;i+1:i-2}} \frac{h^c_{2;i:i-1}}{h^c_{2;i+1:i-2}}\Big) \delta u_{i+1:i-1}$$

$$- \quad \frac{h^u_{2;i+2:i-2}}{h^u_{2;i}} \frac{h^u_{3;i+2:i-2}}{h^u_{3;i}} \frac{h^c_{1;i:i-1}}{h^c_{1;i+1:i-2}} \frac{h^c_{2;i:i-1}}{h^c_{2;i+1:i-2}} \delta u_{i+2:i-2} \tag{5.300}$$

$$\delta v_i = \frac{h^v_{1;j+1:j-1}}{h^v_{1;j}} \frac{h^v_{3;j+1:j-1}}{h^v_{3;j}} \Big(1 + \frac{h^c_{2;j:j-1}}{h^c_{2;j+1:j-2}} \frac{h^c_{1;j:j-1}}{h^c_{1;j+1:j-2}}\Big) \delta v_{j+1:j-1}$$

$$- \quad \frac{h^v_{1;j+2:j-2}}{h^v_{1;j}} \frac{h^v_{3;j+2:j-2}}{h^v_{3;j}} \frac{h^c_{2;j:j-1}}{h^c_{2;j+1:j-2}} \frac{h^c_{1;j:j-1}}{h^c_{1;j+1:j-2}} \delta v_{j+2:j-2} \tag{5.301}$$

$$\tag{5.302}$$

3. Local solution

$$\frac{h^{n+1;u}_{3;ik} \delta u^{n+1}_{ik} - h^{n;u}_{3;ik} \delta u^n_{ik}}{h^{n+1;u}_{3;ik}} = f\Big(\theta_c v^{n+1;c}_{ik} - (1-\theta_c) v^{n;c}_{ik}\Big) \tag{5.303}$$

$$+ \quad F^{b;n}_{1;i+1:i-1,k} - \frac{\overline{F}^{b;n}_{1;i+1:i-1,k}}{H^{n+1;u}_{i+1:i-1}} + \frac{D^{uw}_{13;ij,k+1} - D^{uw}_{13;ik}}{h^u_{3;i}} + \frac{\tau_{b1;i} - \tau_{s1;i}}{H^{n+1;u}_i}$$

$$\frac{h^{n+1;v}_{3;jk} \delta v^{n+1;c}_{jk} - h^{n;v}_{3;jk} \delta v^{n;c}_{jk}}{h^{n+1;v}_{3;jk}} = -f\Big(\theta_c u^{n+1;c}_{jk} - (1-\theta_c) u^{n;c}_{jk}\Big) \tag{5.304}$$

$$+ \quad F^{b;n}_{2;j+1:j-1,k} - \frac{\overline{F}^{b;n}_{2;j+1:j-1,k}}{H^{n+1;v}_{j+1:j-1}} + \frac{D^{vw}_{23;j,k+1} - D^{vw}_{23;jk}}{h^v_{3;jk}} + \frac{\tau_{b2;j} - \tau_{s2;j}}{H^{n+1;v}_j}$$

The diffusive fluxes are obtained from (5.210), (5.212) using the condition (5.245) at the surface and the bottom stress formulations given in Section 5.3.15.2.

4. Radiation condition using the baroclinic wave speed (see equation (4.382)).

$$\delta u^{n+1}_{ijk} = (1 - w^u_{ijk}) \delta u^n_{ijk} + w^u_{ijk} \delta u^n_{i+1:i-1,jk} \tag{5.305}$$

$$\delta v^{n+1}_{ijk} = (1 - w^v_{ijk}) \delta v^n_{ijk} + w^v_{ijk} \delta v^n_{i,j+1:j-1,k} \tag{5.306}$$

The weight factors are given by

$$w^u_{ijk} = \pm \frac{R \Delta t}{h^c_{1;i:i-1,j}} \sqrt{g^c_{i:i-1,j} H^{n+1;c}_{i:i-1,j}}$$

$$w^v_{ijk} = \pm \frac{R \Delta t}{h^c_{2;i,j:j-1}} \sqrt{g^c_{i,j:j-1} H^{n+1;c}_{i,j:j-1}} \tag{5.307}$$

where R is the prescribed ratio of the baroclinic to surface gravity wave speed. Default value is 0.03.

5. Orlanski condition (see equation (4.383)).

   In analogy with the 2-D case one has

$$\delta u^{n+1}_{ijk} = \Big(1 - O_R(r^u_{1;ijk}, r^u_{2;ijk}, r^u_{3;ijk}\Big)\delta u^n_{ijk} + O_R(r^u_{1;ijk}, r^u_{2;ijk}, r^u_{3;ijk})\delta u^n_{i+1:i-1,jk} \tag{5.308}$$

$$\delta v^{n+1}_{ijk} = \Big(1 - O_R(r^v_{1;ijk}, r^v_{2;ijk}, r^v_{3;ijk})\Big)\delta v^n_{ijk} + O_R(r^v_{1;ijk}, r^v_{2;ijk}, r^v_{3;ijk})\delta v^n_{i,j+1:j-1,k} \tag{5.309}$$

   where the Orlanski function O_R is defined by (5.269) and

$$r^u_{1;ijk} = \delta u^n_{i+1:i-1,jk}\,, \quad r^u_{2;ijk} = \delta u^{n-1}_{i+1:i-1,jk}\,, \quad r^u_{3;ijk} = \delta u^{n-1}_{i+2:i-2,jk} \tag{5.310}$$

$$r^v_{1;ijk} = \delta v^n_{i,j+1:j-1,k}\,, \quad r^v_{2;ijk} = \delta v^{n-1}_{i,j+1:j-1,k}\,, \quad r^v_{3;ijk} = \delta v^{n-1}_{i,j+2:j-2,k} \tag{5.311}$$

Once the baroclinic and mean components are known, the full 3-D currents are determined by adding the two components

$$u^{n+1}_{ijk} = U^{n+1}_{ij}/H^{n+1;c}_{i+1:i-1,j} + \delta u^{n+1}_{ijk}\,, \quad v^{n+1}_{ijk} = V^{n+1}_{ij}/H^{n+1;c}_{i,j+1:j-1} + \delta v^{n+1}_{ijk} \tag{5.312}$$

5.3.17.2 open boundary conditions for the advective and diffusive fluxes of the 3-D currents

The formulations are identical to the one given in Section 5.3.16.2, except that the fluxes now include an additional k-index.

5.3.17.3 boundary conditions for the 3-D mode at closed lateral boundaries

The formulations are identical to the one given in Section 5.3.16.3, except that the fluxes now include an additional k-index. They are given for completeness.

Following (4.392)–(4.393) one has

$$u_{ijk} = 0\,, \quad v_{ijk} = 0 \tag{5.313}$$

at coastal boundaries.

Likewise all fluxes for the cross-advective and diffusive terms are set to zero at closed Y- or X-node boundaries:

$$F^{uv}_{12;ijk} = 0.0\,, \quad D^{uv}_{12;ijk} = 0.0\,, \quad F^{uv}_{21;ijk} = 0.0\,, \quad D^{uv}_{21;ijk} = 0.0 \tag{5.314}$$

5.3.18 Solution of the discretised equations for momentum

In the case of a 3-D current the discretised transport equations reduce to a system of linear equations of the form

$$\begin{array}{ccccccc} & & B_{ij1}X^{new}_{ij1} & + & C_{ij1}X^{new}_{ij2} & = & D_{ij1}(X^{old}) \\ A_{ijk}X^{new}_{ij,k-1} & + & B_{ijk}X^{new}_{ijk} & + & C_{ijk}X^{new}_{ij,k+1} & = & D_{ijk}(X^{old}) \\ A_{ij,N_z}X^{new}_{ij,N_z-1} & + & B_{ijN_z}X^{new}_{ijN_z} & & & = & D_{ijN_z}(X^{old}) \end{array} \tag{5.315}$$

where X^{old} and X^{new} are the values of the values of the current (u or v) at the "old" and "new" time step. Equations (5.315) form a tridiagonal matrix system in the vertical, which has to be solved at each horizontal grid point (i,j).

Omitting the i and j indices for simplicity, the elements A_k, B_k and C_k can be written as the sum of different components, each representing particular term(s) in the corresponding momentum equation. Explicit expressions are given below for the update of u at the predictor step without operator splitting, as given by equation (5.7) or (5.31) so that $X^{old} = u^n$ and $X^{new} = u^p$. They are easily extended to the case for v-equation.

When operator splitting is used, four of the six steps are explicit integrations in which case the solution is straigtforward. The two steps (5.58), (5.59) involving implicit terms are treated in a similar way.

The discretised 2-D equations for transports are written as

$$B_{ij}X^{new}_{ij} = D_{ij}X^{old}_{ij} \tag{5.316}$$

The composition of the B- and D-matrices is readily obtained from the discretisation formulae in the preceding sections and is not given here. The solution of (5.316) is straightforward.

5.3.18.1 composition of the tridiagonal matrix

1. Time derivative.

   The contribution of the time derivative is given by

   $$A^t_k = 0\,, \quad B^t_k = 1\,, \quad C^t_k = 0\,, \quad D^t_k = u^n_k \tag{5.317}$$

   where $1 \le k \le N_z$.

2. Vertical advection.

The vertical advection term is split up into two contributions arising from the fluxes below and above a k-level. The former are given by

$$\begin{aligned} A_k^{a-} &= -\theta_a c_k^- (\alpha_k + f_k) \\ B_k^{a-} &= -\theta_a c_k^- (\beta_k - f_k) \\ C_k^{a-} &= 0 \\ D_k^{a-} &= (1-\theta_a) c_k^- \Big((\alpha_k + f_k) u_{k-1}^n + (\beta_k - f_k) u_k^n \Big) \end{aligned} \tag{5.318}$$

where $2 \leq k \leq N_z$,

$$c_k^- = \frac{\Delta t \omega_k^{uw}}{2 h_{3;k}^u}, \quad f_k = \Big(1 - \Omega(r_k^{uw})\Big) s_k \tag{5.319}$$

and α_{ijk}, β_{ijk}, s_{ijk}, r_{ijk}^{uw} are defined by (5.172) and (5.173).

The terms arising from the flux above the k-level, are

$$\begin{aligned} A_k^{a+} &= 0 \\ B_k^{a+} &= \theta_a c_k^+ (\alpha_{k+1} + f_{k+1}) \\ C_k^{a+} &= \theta_a c_k^+ (\beta_{k+1} - f_{k+1}) \\ D_k^{a+} &= -(1-\theta_a) c_k^+ \Big((\alpha_{k+1} + f_{k+1}) u_k^n + (\beta_{k+1} - f_{k+1}) u_{k+1}^n \Big) \end{aligned} \tag{5.320}$$

where $1 \leq k \leq N_z - 1$ and

$$c_k^+ = \frac{\Delta t \omega_{k+1}^{uw}}{2 h_{3;k}^u} \tag{5.321}$$

3. Vertical diffusion.

   As for vertical advection the fluxes below and above a k-level are taken separately. The former are given by

$$\begin{aligned} A_k^{d-} &= -\theta_v \frac{\Delta t \nu_{T;k}^{uw}}{h_{3;k}^u h_{3;k}^{uw}} \\ B_k^{d-} &= \theta_v \frac{\Delta t \nu_{T;k}^{uw}}{h_{3;k}^u h_{3;k}^{uw}} \\ C_k^{d-} &= 0 \\ D_k^{d-} &= -(1-\theta_v) \frac{\Delta t \nu_{T;k}^{uw}}{h_{3;k}^u h_{3;k}^{uw}} (u_k^n - u_{k-1}^n) \end{aligned} \tag{5.322}$$

where $2 \leq k \leq N_z$.

The terms taken from the flux above the k-level, are

$$\begin{aligned} A_k^{d+} &= 0 \\ B_k^{d+} &= \theta_v \frac{\Delta t \nu_{T;k+1}^{uw}}{h_{3;k}^u h_{3;k+1}^{uw}} \\ C_k^{d+} &= -\theta_v \frac{\Delta t \nu_{T;k+1}^{uw}}{h_{3;k}^u h_{3;k+1}^{uw}} \\ D_k^{d+} &= (1-\theta_v) \frac{\Delta t \nu_{T;k+1}^{uw}}{h_{3;k}^u h_{3;k+1}^{uw}} (u_{k+1}^n - u_k^n) \end{aligned} \tag{5.323}$$

where $1 \leq k \leq N_z - 1$.

4. Other explicit terms.

   All other terms are explicit. Their contributions can be written as

$$\begin{aligned} A_k^e &= B_k^e = C_k^e = 0 \\ D_k^e &= \Delta t \Big(\mathcal{O}_{1;k} - \tilde{\mathcal{A}}_{h1}(u)_k^n - \tilde{\mathcal{A}}_{h2}(u)_k^n + \mathcal{D}_{mh1}(\tau_{11})_k^{n;u} + \mathcal{D}_{mh2}(\tau_{12})_k^{n;u} \Big) \end{aligned} \tag{5.324}$$

   with $\mathcal{O}_{1;k}$ defined by (5.63).

5. Surface boundary conditions.

   The contributions arising from the surface boundary conditions (5.244) for advection and (5.245) for diffusion are added to the highest level:

$$\begin{aligned} A_{N_z}^{a+} &= B_{N_z}^{a+} = C_{N_z}^{a+} = D_{N_z}^{a+} = 0 \\ A_{N_z}^{d+} &= B_{N_z}^{d+} = C_{N_z}^{d+} = 0 \\ D_{N_z}^{d+} &= \frac{\Delta t \tau_{s1}^u}{h_{3;N_z}^u} \end{aligned} \tag{5.325}$$

6. Bottom boundary conditions.

   The contributions arising from the bottom boundary conditions (5.248) for advection are added to the lowest level:

$$A_1^{a-} = B_1^{a-} = C_1^{a-} = D_1^{a-} = 0 \tag{5.326}$$

   The bottom contributions for vertical diffusion depends on whether the bottom stress is expressed as function of the bottom current (equation

(5.249))

$$A_1^{d-} = C_1^{d-} = 0\,, \quad B_1^{d-} = \theta_v \frac{\Delta t k_b^u}{h_{3;1}^u}\,, \quad D_1^{d-} = -(1-\theta_v)\frac{\Delta t k_b^u u_1^n}{h_{3;1}^u} \tag{5.327}$$

or the depth mean current (equation (5.252))

$$A_1^{d-} = B_1^{d-} = C_1^{d-} = 0\,, \quad D_1^{d-} = -\frac{\Delta t k_b^u \overline{u}^n}{h_{3;1}^u} \tag{5.328}$$

where the friction velocity is calculated using (5.251) or (5.254).

5.3.18.2 solution of tridiagonal systems

Tridiagonal matrix systems of the form (5.315) are solved using the algorithm, described in Press *et al.* (1992):

$$\begin{aligned}
\beta_1 &= B_1\,,\ X_1^\star = D_1/\beta_1 \\
\gamma_k &= C_{k-1}/\beta_{k-1}\,,\ \beta_k = B_k - A_k\gamma_k\,,\ X_k^\star = (D_k - A_k X_{k-1}^\star)/\beta_k \\
&\qquad \text{for } k = 2,\cdots,N_z \\
X_{N_z}^{new} &= X_{N_z}^\star \\
X_k^{new} &= X_k^\star - \gamma_{k+1} X_{k+1}^{new} \text{ for } k = N_z-1,\cdots,1
\end{aligned} \tag{5.329}$$

5.3.19 Elliptic equation for the free surface correction

When the implicit method for the surface slope term in the momentum equations is taken, an elliptic equation is obtained for the free surface correction ζ'. The discretised form of this equation is written in the form (5.37). The matrices A to G are evaluated in two phases:

1. The transports are first taken at interior wet points only and set to zero at solid and open boundaries.

2. The explicit and implicit terms arising by applying the open boundary condition are added.

5.3.19.1 interior terms

Defining

$$p_{ij}^u = \Delta t \mu_{ij}^u g_{ij}^u H_{ij}^{n+1,it;u} \frac{h_{2;ij}^u}{h_{1;ij}^u}\,, \quad p_{ij}^v = \Delta t \mu_{ij}^v g_{ij}^v H_{ij}^{n+1,it;v} \frac{h_{1;ij}^v}{h_{2;ij}^v} \tag{5.330}$$

where μ equals 1 at wet interior nodes and 0 at solid or open boundary velocity nodes, the coefficients of the elliptic matrix equation (5.37) become

$$\begin{aligned} A_{ij} &= -p^u_{ij}\,, \quad B_{ij} = -p^v_{ij}\,, \quad D_{ij} = -p^v_{i,j+1}\,, \quad E_{ij} = -p^u_{i+1,j} \\ C_{ij} &= \frac{h^c_{1;ij}h^c_{2;ij}}{\Delta t} + p^u_{i+1,j} + p^u_{ij} + p^v_{i,j+1} + p^v_{ij} \\ F_{ij} &= \frac{h^c_{1;ij}h^c_{2;ij}}{\Delta t}(\zeta^n_{ij} - \zeta^{n+1,it}) - h^u_{2;i+1,j}\mu^u_{i+1,j}U^p_{i+1,j} + h^u_{2;ij}\mu^u_{ij}U^p_{ij} \\ &\quad -h^v_{1;i,j+1}\mu^v_{i,j+1}V^p_{i,j+1} + h^v_{1;ij}\mu^v_{ij}V^p_{ij} \end{aligned} \tag{5.331}$$

5.3.19.2 open boundary terms

From the discetised forms of the open boundary conditions additional terms are added to the coefficient matrices. They are described below for each type of open boundary condition using the notations introduced in Section 5.3.16.1. The following definitions are made in addition

$$\begin{aligned} m^u_{ij} &= \Delta t g^u_{i+1:i-1,j} H^{n+1,it;u}_{i+1:i-1,j} \frac{h^u_{2;ij}}{h^u_{1;i+1:i-1,j}} \\ m^v_{ij} &= \Delta t g^v_{i,j+1:j-1} H^{n+1,it;v}_{i,j+1:j-1} \frac{h^v_{1;ij}}{h^v_{2;i,j+1:j-1}} \end{aligned} \tag{5.332}$$

$$\beta^u_{ij} = \frac{1}{1 + 1.5\alpha^u_{ij} r^u_{ij}}\,, \quad \beta^v_{ij} = \frac{1}{1 + 1.5\alpha^v_{ij} r^v_{ij}} \tag{5.333}$$

$$\begin{aligned} O^u_{ij} &= \Delta t(f^u_{ij}V^c_{i:i-1} + H^{n+1,it;u}_{ij} F^{t;n+1}_{1;ij} + \tau^c_{s1;ij} - \tau^{n;u}_{b1;ij}) \\ O^v_{ij} &= \Delta t(-f^v_{ij}U^c_{i,j:j-1} + H^{n+1,it;v}_{ij} F^{t;n+1}_{2;ij} + \tau^c_{s2;ij} - \tau^{n;v}_{b2;ij}) \end{aligned} \tag{5.334}$$

0. Clamped

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij}U^n_{ij} \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij}V^n_{ij} \end{aligned} \tag{5.335}$$

1. Zero slope

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij}(U^n_{ij} + O^u_{ij}) \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij}(V^n_{ij} + O^v_{ij}) \end{aligned} \tag{5.336}$$

2. Zero volume flux

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} U^p_{i+1:i-1,j} \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} V^p_{i,j+1:j-1} \end{aligned} \tag{5.337}$$

$$\begin{aligned} C_{i:i-1,j} &= C_{i:i-1,j} - h^u_{2;ij} m^u_{ij} \\ C_{i,j:j-1} &= C_{i,j:j-1} - h^v_{1;ij} m^v_{ij} \end{aligned} \tag{5.338}$$

$$\begin{aligned} \text{West} &: \quad E_{ij} = E_{ij} + h^u_{2;ij} m^u_{ij} \\ \text{East} &: \quad A_{i-1,j} = A_{i-1,j} + h^u_{2;ij} m^u_{ij} \\ \text{South} &: \quad D_{ij} = D_{ij} + h^v_{1;ij} m^v_{ij} \\ \text{North} &: \quad B_{i,j-1} = B_{i,j-1} + h^v_{1;ij} m^v_{ij} \end{aligned} \tag{5.339}$$

3. Specified elevation

$$\begin{aligned} U^{L;n+1,it}_{ij} &= U^{L;n}_{ij} \mp s^u_{ij} \alpha^u_{ij} c^u_{ij} r^u_{ij} (\zeta^{n+1,it}_{i:i-1,j} - \zeta^e_{ij}) + O^u_{ij} \\ V^{L;n+1,it}_{ij} &= V^{L;n}_{ij} \mp s^v_{ij} \alpha^v_{ij} c^v_{ij} r^v_{ij} (\zeta^{n+1,it}_{i,j:j-1} - \zeta^e_{ij}) + O^v_{ij} \end{aligned} \tag{5.340}$$

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} U^{L;n+1,it}_{ij} \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} V^{L;n+1,it}_{ij} \end{aligned} \tag{5.341}$$

$$\begin{aligned} C_{i:i-1,j} &= C_{i:i-1,j} + h^u_{2;ij} s^u_{ij} \alpha^u_{ij} c^u_{ij} r^u_{ij} \\ C_{i,j:j-1} &= C_{i,j:j-1} + h^v_{1;ij} s^v_{ij} \alpha^v_{ij} c^v_{ij} r^v_{ij} \end{aligned} \tag{5.342}$$

4. Specified transport

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} U^e_{ij} \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} V^e_{ij} \end{aligned} \tag{5.343}$$

5. Radiation condition using shallow water speed

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} \frac{U^n_{ij} + \alpha^u_{ij} U^p_{i+1:i-1,j}}{1 + \alpha^u_{ij}} \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} \frac{V^n_{ij} + \alpha^v_{ij} V^p_{i,j+1:j-1}}{1 + \alpha^v_{ij}} \end{aligned} \tag{5.344}$$

$$\begin{aligned} C_{i:i-1,j} &= C_{i:i-1,j} - \frac{h^u_{2;ij} m^u_{ij} \alpha^u_{ij}}{1+\alpha^u_{ij}} \\ C_{i,j:j-1} &= C_{i,j:j-1} - \frac{h^v_{1;ij} m^v_{ij} \alpha^v_{ij}}{1+\alpha^v_{ij}} \end{aligned} \tag{5.345}$$

$$\begin{aligned} \text{West} \quad &: \quad E_{ij} = E_{ij} + \frac{h^u_{2;ij} m^u_{ij} \alpha^u_{ij}}{1+\alpha^u_{ij}} \\ \text{East} \quad &: \quad A_{i-1,j} = A_{i-1,j} + \frac{h^u_{2;ij} m^u_{ij} \alpha^u_{ij}}{1+\alpha^u_{ij}} \\ \text{South} \quad &: \quad D_{ij} = D_{ij} + \frac{h^v_{1;ij} m^v_{ij} \alpha^v_{ij}}{1+\alpha^v_{ij}} \\ \text{North} \quad &: \quad B_{i,j-1} = B_{i,j-1} + \frac{h^v_{1;ij} m^v_{ij} \alpha^v_{ij}}{1+\alpha^v_{ij}} \end{aligned} \tag{5.346}$$

6. Orlanski (1976) condition

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} \Big(1 - O_R) U^n_{ij} + O_R U^n_{i+1:i-1,j} \Big) \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} \Big(1 - O_R) V^n_{ij} + O_R V^n_{i,j+1:j-1} \Big) \end{aligned} \tag{5.347}$$

where the Orlanski weight function is defined by (5.269)–(5.271) at U- and V-nodes.

7. Camerlengo & O'Brien (1980)

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} U^n_{i+1:i-1,j} \quad \text{if} \quad U^n_{i+1:i-1,j} \geq U^{n-1}_{i+2:i-2,j} \\ F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} U^n_{ij} \quad \text{otherwise} \end{aligned} \tag{5.348}$$

$$\begin{aligned} F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} V^n_{i,j+1:j-1} \quad \text{if} \quad V^n_{i,j+1:j-1} \geq V^{n-1}_{i,j+2:j-2} \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} V^n_{ij} \quad \text{otherwise} \end{aligned} \tag{5.349}$$

8. Flather (1976) with specified elevation and transport

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij} \Big(U^e_{ij} \mp \frac{1}{2} s^u_{ij} c^u_{ij} (\zeta^{n+1,it}_{i:i-1,j} - \zeta^e_{ij}) \Big) \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij} \Big(V^e_{ij} \mp \frac{1}{2} s^v_{ij} c^v_{ij} (\zeta^{n+1,it}_{i,j:j-1} - \zeta^e_{ij}) \Big) \end{aligned} \tag{5.350}$$

$$\begin{aligned} C_{i:i-1,j} &= C_{i:i-1,j} + \frac{1}{2} h^u_{2;ij} s^u_{ij} c^u_{ij} \\ C_{i,j:j-1} &= C_{i,j:j-1} + \frac{1}{2} h^v_{1;ij} s^v_{ij} c^v_{ij} \end{aligned} \tag{5.351}$$

9. Flather (1976) with specified elevation

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij}\Big(U^{L;n+1,it}_{ij} \mp \frac{1}{2}s^u_{ij}c^u_{ij}(\zeta^{n+1,it}_{i:i-1,j} - \zeta^e_{ij})\Big) \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij}\Big(V^{L;n+1,it}_{ij} \mp \frac{1}{2}s^v_{ij}c^v_{ij}(\zeta^{n+1,it}_{i,j:j-1} - \zeta^e_{ij})\Big) \end{aligned} \tag{5.352}$$

where the local solutions ($U^{L;n+1,it}$,$V^{L;n+1,it}$) are determined by (5.340).

$$\begin{aligned} C_{i:i-1,j} &= C_{i:i-1,j} + h^u_{2;ij}s^u_{ij}c^u_{ij}(\frac{1}{2} + \alpha^u_{ij}r^u_{ij}) \\ C_{i,j:j-1} &= C_{i,j:j-1} + h^v_{1;ij}s^v_{ij}c^v_{ij}(\frac{1}{2} + \alpha^v_{ij}r^v_{ij}) \end{aligned} \tag{5.353}$$

10. Røed & Smedstad (1984)

$$\begin{aligned} \zeta^{L;n+1,it}_{ij} &= \zeta^{L;n}_{ij} - \frac{\Delta t(h^v_{1;i:i-1,j+1}V^n_{i:i-1,j+1} - h^v_{1;i:i-1,j}V^n_{i:i-1,j})}{h^c_{1;i:i-1,j}h^c_{2;i:i-1,j}} \\ \zeta^{L;n+1,it}_{ij} &= \zeta^{L;n}_{ij} - \frac{\Delta t(h^u_{2;i+1,j:j-1}U^n_{i+1,j:j-1} - h^u_{2;i,j:j-1}U^n_{i,j:j-1})}{h^c_{1;i,j:j-1}h^c_{2;i,j:j-1}} \end{aligned} \tag{5.354}$$

$$\begin{aligned} F_{i:i-1,j} &= F_{i:i-1,j} \pm h^u_{2;ij}\Big(U^{L;n+1,it}_{ij} - s^u_{ij}c^u_{ij}(\zeta^{n+1,it}_{i:i-1,j} - \zeta^{L;n+1,it}_{ij})\Big) \\ F_{i,j:j-1} &= F_{i,j:j-1} \pm h^v_{1;ij}\Big(V^{L;n+1,it}_{ij} - s^v_{ij}c^v_{ij}(\zeta^{n+1,it}_{i,j:j-1} - \zeta^{L;n+1,it}_{ij})\Big) \end{aligned} \tag{5.355}$$

$$\begin{aligned} C_{i:i-1,j} &= C_{i:i-1,j} + h^u_{2;ij}c^u_{ij}(1 + s^u_{ij}\alpha^u_{ij}r^u_{ij}) \\ C_{i,j:j-1} &= C_{i,j:j-1} + h^v_{1;ij}c^v_{ij}(1 + s^v_{ij}\alpha^v_{ij}r^v_{ij}) \end{aligned} \tag{5.356}$$

11. Characteristic method with specified elevation and transport

$$\begin{aligned} R^{n+1,it;u}_{i;ij} &= U^e_{ij} \pm \frac{1}{2}s^u_{ij}c^u_{ij}\Big(\zeta^e_{ij} + (2 - s^u_{ij})\zeta^{n+1;it}_{i:i-1,j}\Big) \\ R^{n+1,it;v}_{i;ij} &= V^e_{ij} \pm \frac{1}{2}s^v_{ij}c^v_{ij}\Big(\zeta^e_{ij} + (2 - s^v_{ij})\zeta^{n+1,it}_{i,j:j-1}\Big) \end{aligned} \tag{5.357}$$

$$(1 + \frac{3}{2}\alpha^u_{ij}r^u_{ij})R^{n+1,it;u}_{o;ij} = R^{n;u}_{o;ij}$$

$$
\begin{aligned}
&+ \alpha^u_{ij} r^u_{ij} (U^p_{i+1:i-1,j} + \frac{1}{2} R^{n+1,it;u}_{i;ij} \mp 2 c^u_{ij} \zeta^{n+1,it}_{i:i-1,j}) \\
&+ \frac{\alpha^u_{ij}}{h^c_{2;i:i-1,j}} \Big(\pm (h^v_{1;i:i-1,j+1} V^n_{i:i-1,j+1} - h^v_{1;i:i-1,j} V^n_{i:i-1,j}) \\
&+ U^{n;c}_{i:i-1,j} (h^u_{2;i+1:i-1,j} - h^u_{2;ij}) \Big) + O^u_{ij}
\end{aligned} \tag{5.358}
$$

and

$$
\begin{aligned}
(1 + \frac{3}{2} \alpha^v_{ij} r^v_{ij}) R^{n+1,it;v}_{o;ij} &= R^{n;v}_{o;ij} \\
&+ \alpha^v_{ij} r^v_{ij} (V^p_{i,j+1:j-1} + \frac{1}{2} R^{n+1,it;v}_{i;ij} \mp 2 c^v_{ij} \zeta^{n+1,it}_{i,j:j-1}) \\
&+ \frac{\alpha^v_{ij}}{h^c_{1;i,j:j-1}} \Big(\pm (h^u_{2;i+1,j:j-1} U^n_{i+1,j:j-1} - h^u_{2;i,j:j-1} U^n_{i,j:j-1}) \\
&+ V^{n;c}_{i,j:j-1} (h^v_{1;i,j+1:j-1} - h^v_{1;ij}) \Big) + O^v_{ij}
\end{aligned} \tag{5.359}
$$

$$
\begin{aligned}
F_{i:i-1,j} &= F_{i:i-1,j} \pm \frac{1}{2} h^u_{2;ij} \Big(R^{n+1,it;u}_{i;ij} + R^{n+1,it;u}_{o;ij} \Big) \\
F_{i,j:j-1} &= F_{i,j:j-1} \pm \frac{1}{2} h^v_{1;ij} \Big(R^{n+1,it;v}_{i;ij} + R^{n+1,it;v}_{o;ij} \Big)
\end{aligned} \tag{5.360}
$$

$$
\begin{aligned}
C_{i:i-1,j} &= C_{i:i-1,j} - \frac{1}{4} h^u_{2;ij} s^u_{ij} c^u_{ij} (2 - s^u_{ij}) (1 + \frac{1}{2} \alpha^u_{ij} r^u_{ij} \beta^u_{ij}) \\
&\quad + h^u_{2;ij} \alpha^u_{ij} c^u_{ij} r^u_{ij} \beta^u_{ij} - \frac{1}{2} \alpha^u_{ij} r^u_{ij} \beta^u_{ij} m^u_{ij}
\end{aligned} \tag{5.361}
$$

$$
\begin{aligned}
C_{i,j:j-1} &= C_{i,j:j-1} - \frac{1}{4} h^v_{1;ij} s^v_{ij} c^v_{ij} (2 - s^v_{ij}) (1 + \frac{1}{2} \alpha^v_{ij} r^v_{ij} \beta^v_{ij}) \\
&\quad + h^v_{1;ij} \alpha^v_{ij} c^v_{ij} r^v_{ij} \beta^v_{ij} - \frac{1}{2} \alpha^v_{ij} r^v_{ij} \beta^v_{ij} m^v_{ij}
\end{aligned} \tag{5.362}
$$

$$
\begin{aligned}
\text{West} &: \quad E_{ij} = E_{ij} + \frac{1}{2} h^u_{2;ij} \alpha^u_{ij} r^u_{ij} \beta^u_{ij} m^u_{ij} \\
\text{East} &: \quad A_{i-1,j} = A_{i-1,j} + \frac{1}{2} h^u_{2;ij} \alpha^u_{ij} r^u_{ij} \beta^u_{ij} m^u_{ij} \\
\text{South} &: \quad D_{ij} = D_{ij} + \frac{1}{2} h^v_{1;ij} \alpha^v_{ij} r^v_{ij} \beta^v_{ij} m^v_{ij} \\
\text{North} &: \quad B_{i,j-1} = B_{i,j-1} + \frac{1}{2} h^v_{1;ij} \alpha^v_{ij} r^v_{ij} \beta^v_{ij} m^v_{ij}
\end{aligned} \tag{5.363}
$$

12. Characteristic method with specified elevation

    The discretisations are the same as in the previous case with (U^{e}_{ij},V^{e}_{ij}) replaced by $(U^{L;n+1,it}_{ij},V^{L;n+1,it}_{ij})$.

13. Characteristic method using zero normal gradient

    The discretisations are the same as in the previous case with $(R^{n+1,it;u}_{i;ij},R^{n+1,it,v}_{i;ij})$ defined by

$$\begin{aligned} R^{n+1,it;u}_{i;ij} &= R^{n;u}_{i;ij} \\ &- \frac{\alpha^{u}_{ij}}{h^{c}_{2;i:i-1,j}}\Big(\pm(h^{v}_{1;i:i-1,j+1}V^{n}_{i:i-1,j+1} - h^{v}_{1;i:i-1,j}V^{n}_{i:i-1,j}) \\ &+ U^{c;m}_{i:i-1,j}(h^{u}_{2;i+1:i-1,j} - h^{u}_{2;ij})\Big) + O^{u}_{ij} \end{aligned} \tag{5.364}$$

    and

$$\begin{aligned} R^{n+1,it;v}_{i;ij} &= R^{n;v}_{i;ij} \\ &- \frac{\alpha^{v}_{ij}}{h^{c}_{1;i,j:j-1}}\Big(\pm(h^{u}_{2;i+1,j:j-1}U^{n}_{i+1,j:j-1} - h^{u}_{2;i,j:j-1}U^{n}_{i,j:j-1}) \\ &+ V^{c;m}_{i,j:j-1}(h^{v}_{1;i,j+1:j-1} - h^{v}_{1;ij})\Big) + O^{v}_{ij} \end{aligned} \tag{5.365}$$

5.4 Drying/wetting and inundation schemes

5.4.1 Drying and wetting algorithm

The drying and flooding algorithm implemented in COHERENS closely follows the version used in the GETM model (Burchard & Bolding, 2002) and consists of the following steps.

1. The advective, horizontal diffusion, Coriolis, curvature and baroclinic pressure gradient terms in the 2-D and 3-D momentum equations are multiplied by a "drying" factor α. For example, the u-equation becomes

$$\begin{aligned} \frac{1}{h_3}\frac{\partial}{\partial t}\big(h_3 u\big) &+ \alpha\Big[\mathcal{A}_{h1}(u) + \mathcal{A}_{h2}(u) + \mathcal{A}_{v}(u) \\ &+ \frac{v}{h_1 h_2}\Big(u\frac{\partial h_1}{\partial \xi_2} - v\frac{\partial h_2}{\partial \xi_1}\Big) - 2\Omega v\sin\phi\Big] \\ &= -\frac{g}{h_1}\frac{\partial \zeta}{\partial \xi_1} - \frac{1}{\rho_0 h_1}\frac{\partial P_a}{\partial \xi_1} + \alpha F^{b}_{1} + F^{t}_{1} + \mathcal{D}_{mv}(u) \\ &+ \alpha\Big(\mathcal{D}_{mh1}(\tau_{11}) + \mathcal{D}_{mh2}(\tau_{12})\Big) \end{aligned} \tag{5.366}$$

where α decreases from 1 when the total water depth is lower than a critical value and becomes 0 when H reaches a minimum value

$$\begin{aligned} \alpha &= 1 && \text{if} \quad H > d_{crit} \\ \alpha &= \frac{H - d_{min}}{d_{crit} - d_{min}} && \text{if} \quad d_{min} \leq H \leq d_{crit} \\ \alpha &= 0 && \text{if} \quad H < d_{min} \end{aligned} \tag{5.367}$$

In this way the momentum equations reduce to a balance between the time derivative, surface slope and vertical diffusion (bottom friction) terms when $H \rightarrow d_{min}$. The formulation provides a continuous transition from a wet to a dry condition in contrast to schemes, discussed below, using a "mask" function which sets cells to a dry or wet state depending on some drying criterium. A second advantage is that the scheme only involves two tunable parameters. Default values are d_{crit}=0.1 m, d_{min}=0.02 m.

2. The total water depth at C-nodes is bounded below by the minimum water depth to prevent negative water depths

$$H^c_{ij} = \max(H^c_{ij}, d_{min}) \tag{5.368}$$

so that a (spurious) small amount of water remains if a cell becomes dry. Equation (5.368) implies a second condition for the surface level

$$\zeta_{ij} \geq d_{min} - h_{ij} \tag{5.369}$$

This means that if the value of ζ_{ij}, obtained from the continuity equation, falls below the minimum value (5.369), a (small) amount of water is added to the water column in violation of mass conservation. No correction is applied in the current version of COHERENS, but the spuriously added water is stored in the program variable $\delta_e H$ which measures the error in water depth and is increased by the amount $d_{min} - H$ when H drops below d_{min}. By its definition $\delta_e H$ is non-negative and cannot decrease in time.

3. In the absence of a drying mechanism the total water depth at the velocity nodes is calculated as the (weigthed) average of the depths at the surrounding C-nodes. Experiments showed that this method produces large horizontal gradients for the vertical grid spacings for water depths close to the minimum value, and, hence, unrealistically high vertical current magnitudes after substitution in the baroclinic

continuity equation (4.102). To smoothen the solution the depth at the velocity is taken as the minimum of the surrounding C-node values, i.e.

$$\begin{aligned} H^u_{ij} &= \min(H^c_{i-1,j}, H^c_{ij}) \qquad \text{if} \qquad \min(H^c_{i-1,j}, H^c_{ij}) < d_{crit} \\ H^v_{ij} &= \min(H^c_{i,j-1}, H^c_{ij}) \qquad \text{if} \qquad \min(H^c_{i,j-1}, H^c_{ij}) < d_{crit} \end{aligned} \tag{5.370}$$

Disadvantage is an unphysical retardation of the flow through the velocity interfaces.

4. To prevent an unrealistic outflow from a drying cell the surface slope is calculated with a modified surface elevation at both sides

$$\begin{aligned} \frac{\partial \zeta}{\partial \xi_1}\Big|_i &= \max\Big(\zeta_i, \min(d_{min}, H_i - h_{i-1})\Big) \\ &- \max\Big(\zeta_{i-1}, \min(d_{min}, H_{i-1} - h_i)\Big) \\ \frac{\partial \zeta}{\partial \xi_2}\Big|_j &= \max\Big(\zeta_j, \min(d_{min}, H_j - h_{j-1})\Big) \\ &- \max\Big(\zeta_{j-1}, \min(d_{min}, H_{j-1} - h_j)\Big) \end{aligned} \tag{5.371}$$

5. Equations (5.18)–(5.19) show that the bottom drag coefficient and hence the bottom friction increases exponentially at small water depths. This will slow down the water flow at water depths close to the minimum value. Note that this effect is only present when the bottom drag coefficient is calculated as function of a roughness length.

5.4.2 Inundation schemes

The drying/wetting scheme described in Section 5.4.1 has been extended in Version 2.3 of COHERENS by the implementation of so-called "mask functions". The objective is to simulate inundation processes where coastal boundaries are moving dynamically or to simulate the flow over obstacles. Therefore, grid cells of the computational domain will become "dry" or "wet" depending on the value of the total water depth. In this way users have the additional option to simulate inundation apart from the original drying/wetting scheme already present in COHERENS.

Inundation schemes are focused on simulating dynamic processes. "Drying and wetting" refers to an existent functionality used to define "wet" and "dry" areas in the computational domain. COHERENS uses this functionality

to define the coastal boundaries, this definition is performed at the initialisation stage and only once. In this way a distinction is made of three type of grid cells

- Land cells which are always dry and where no calculation is performed. They are defined by the user as cells where the mean water depth has a flagged value.
- Cells which are (temporarily) dry (non-active) but can be inundated by the rising water. Calculations in these cells are disabled until they are inundated from a neighbouring cell. The total water depth remains limited from below by d_{min} so that at least a small amount of water is present. In this way, dynamic coastal boundaries can be simulated. Note that land topography is represented by negative bathymetric values (see below) so that there exists no lower limit for the mean water depth except for values equal to the data flag.
- Active wet cells where all calculations are enabled.

The "mask functions" are defined as criteria for "masking" grid cells according to their condition (dry or wet). When the criterium evaluates as .TRUE. at a particular grid cell, the mask function will "mask in" the cell. Hence, they will be considered for the solution of the hydrodynamic equations. On the other hand, if grid cells become dry, the mask criterium will "mask out" such grid cells and updates of quantitites defined at these cells will be suspended. Dry cells are also excluded from the interpolation of model variables on the model grid (see Section 10.2.2). The process is repeated at the start of each predictor time step. Once a cell is set to a "dry" status, the adjacent velocity nodes are blocked and the all currents at these nodes are set to zero to prevent further drying of the grid cell. The criteria are applied at the start of each predictor time step. If the criterium at a dry cell evaluates as .FALSE. at a later time, the cell is reactivated again and water is allowed to enter through the side faces.

Eleven mask functions are defined and can be used in combined form. They can be divided in four groups. The first group compares the water depths of a cell and its neighbours with a threshold value d_{th} and is composed of the following six criteria:

$$\max(H_{ij}, H_{i-1,j}, H_{i+1,j}, H_{i,j-1}, H_{i,j+1}) < d_{th} \quad (5.372)$$

$$\min(H_{ij}, H_{i-1,j}, H_{i+1,j}, H_{i,j-1}, H_{i,j+1}) < d_{th} \quad (5.373)$$

$$\text{mean}(H_{ij}, H_{i-1,j}, H_{i+1,j}, H_{i,j-1}, H_{i,j+1}) < d_{th} \quad (5.374)$$

$$\max(H_{i-1,j}, H_{i+1,j}, H_{i,j-1}, H_{i,j+1}) < d_{th} \quad (5.375)$$
$$\min(H_{i-1,j}, H_{i+1,j}, H_{i,j-1}, H_{i,j+1}) < d_{th} \quad (5.376)$$
$$\text{mean}(H_{i-1,j}, H_{i+1,j}, H_{i,j-1}, H_{i,j+1}) < d_{td} \quad (5.377)$$

where "mean" denotes an averaged value (excluding land cells which are permanently dry).

A second group of criteria verifies the "status" of the neighbouring cells. The status is defined by the function $\mathcal{N}$ which evaluates to 0 at dry and 1 at sea cells[7]. The following criteria, used to prevent the formation of isolated dry or wet cells, have been implemented:

$$\max(\mathcal{N}_{i-1,j}, \mathcal{N}_{i+1,j}, \mathcal{N}_{i,j-1}, \mathcal{N}_{i,j+1}) = 0 \quad (5.378)$$
$$\min(\mathcal{N}_{i-1,j}, \mathcal{N}_{i+1,j}, \mathcal{N}_{i,j-1}, \mathcal{N}_{i,j+1}) = 0 \quad (5.379)$$

The third group is a variant of the previous one and checks, in addition, whether the total water depth of the grid cell is lower than the threshold value:

$$\max(\mathcal{N}_{i-1,j}, \mathcal{N}_{i+1,j}, \mathcal{N}_{i,j-1}, \mathcal{N}_{i,j+1}) = 0 \quad \text{and} \quad H_{ij} < d_{th} \quad (5.380)$$
$$\min(\mathcal{N}_{i-1,j}, \mathcal{N}_{i+1,j}, \mathcal{N}_{i,j-1}, \mathcal{N}_{i,j+1}) = 0 \quad \text{and} \quad H_{ij} < d_{th} \quad (5.381)$$

The last scheme is intended for channel flows and overflowing dykes. The criterium uses the total and mean water depths at the grid cell and its neighbours

$$\min(h_{i-1,j} - H_{i-1,j}, h_{i+1,j} - H_{i+1,j}) > h_{ij} \quad (5.382)$$

or

$$\min(h_{i,j-1} - H_{i,j-1}, h_{i,j+1} - H_{i,j+1}) > h_{ij} \quad (5.383)$$

depending on whether the along-channel direction is along the X- or Y-axis.

The above criteria can be in applied in combination. This means that, if several criteria have been activated by the user, the cell becomes dry if at least one of them turns .TRUE. The cell becomes wet again if all of them evaluate to .FALSE.

The following remarks are given for the user:

- The inundation algorithms can be used for tidal flats, i.e. areas below mean sea level becomes dry/wet during low/high tide, as well as for land areas above mean sea level. In the latter case, a topography of the land, located above the mean sea level, has to be supplied. Land topography is represented in the code by cells with a negative mean

[7]Represented by the model variable nodeatc (see Section 10.1.2.4).

water depth, sea areas by positive mean water depths. A data flag, given by the model parameter depmean_flag, has to assigned to the locations, considered as permanent land.

- In the 3-D case, the mask cannot be changed during a 3-D (baroclinic) time step. This means that the mask criteria are tested after the last corrector step and before the next predictor time step and not at each barotropic time step. When a purely 2-D grid has been selected, the criteria are obviously applied at each (barotropic) time step.
- In the current implementation, open boundaries cannot be blocked so that they should be located where no drying process can take place.
- The threshold depth d_{th}, used to determine whether a cell is dry or wet, should be defined to a value sufficiently larger than the minimum depth d_{min}, but at the same time lower than the critical depth d_{crit}.

5.5 Scalar transport equations

5.5.1 General aspects of discretisation

General features of the discretisation are

- With exception of turbulence variables (see Section 5.6) scalar quantities are located at C-nodes.
- Horizontal advection and diffusion terms are discretised explicitly in time.
- In analogy with the momentum equations vertical advection is taken semi-implicitly while vertical diffusion is treated fully implicitly. The equations for vertical advection and diffusion are presented here in a more general form covering both the explicit, implicit and semi-implicit cases.
- As recommended by Ruddick (1995), the vertical spacing h_3 is eliminated from the time derivative (except in the absence of advection) by adding corrector terms to the right hand side of the transport equation.
- Source terms are discretised explicitly. Contrary to the momentum and turbulence transport equations the sink terms are evaluated explicitly. This has no impact on temperature and salinity, but has been introduced for future implementation of biological concentrations where conservation plays an important role.

- When the momentum equations are solved using mode-splitting, the advecting current (u_f,v_f) used for horizontal advection is composed of the baroclinic current at the "predictor" step plus a filtered depth-independent component obtained by averaging over the more rapidly varying 2-D mode (see equations (5.26)–(5.27). Otherwise, $u_f = u^{n+1}$, $v_f = v^{n+1}$.

- The transport equation is integrated in time with or without the operator splitting method depending on the type of advection scheme. Note that the program allows to use of different advection schemes in the momentum and scalar transport modules.

In analogy with momentum the time discretisation of a scalar transport equation depends on the type of advection scheme selected in the program. Several schemes are available. The type is selected with the model switch iopt_adv_scal which, in analogy with the switch iopt_adv_3d for momentum, has the following meaning

0: horizontal and vertical advection disabled

1: upwind scheme for horizontal and vertical advection

2: Lax-Wendroff scheme for horizontal, central scheme for vertical advection[8]

3: TVD (Total Variation Diminishing) scheme using the superbee limiter as a weighting function between the upwind scheme and either the Lax-Wendroff scheme in the horizontal or the central scheme in the vertical

4: as the previous case now using the monotonic limiter.

5.5.2 Alternative formulation of the transport equation

The general form of a scalar transport equation is given by (4.76). Before discussing its discretisation, the following modifications are made

1. The advective velocities u, v are replaced by the "filtered" currents u_f, v_f given as the sum of the "filtered" depth-mean current and the baroclinic part of the "predicted" current so that the numerical time-integration guarantees the conservation of the scalar quantity (Deleersnijder, 1993).

[8]The "pure" Lax-Wendroff scheme has only been implemented for illustrative purposes and should be avoided in realistic simulations.

2. Making use of the continuity equation (4.60), the time derivative term is written as

$$\begin{aligned}\frac{1}{h_3}\frac{\partial}{\partial t}(h_3\psi) &= \frac{\partial\psi}{\partial t}+\frac{\psi}{h_3}\frac{\partial h_3}{\partial t}\\ &= \frac{\partial\psi}{\partial t}-\mathcal{C}^f_{s1}(\psi)-\mathcal{C}^f_{s2}(\psi)-\mathcal{C}_{s3}(\psi)\end{aligned} \tag{5.384}$$

where the "corrector" terms are defined by

$$\mathcal{C}^f_{s1}(\psi) = \frac{\psi}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}(h_2h_3u_f) \tag{5.385}$$

$$\mathcal{C}^f_{s2}(\psi) = \frac{\psi}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}(h_1h_3v_f) \tag{5.386}$$

$$\mathcal{C}_{s3}(\psi) = \frac{\psi}{h_3}\frac{\partial\omega}{\partial s} \tag{5.387}$$

3. For reasons of conservation the source and sink are both discretised explicitly. The operation is not without risk since the method may produce negative concentrations (Burchard *et al.*, 2003). To simplify the notations the following operator is defined

$$\mathcal{T}(\psi) = \mathcal{P}(\psi)-\mathcal{S}(\psi) \tag{5.388}$$

An alternative method is the Patankar scheme, discussed in Section 5.6 below, which is monotone but does not guarantee conservation.

The new form of the scalar transport equation then becomes

$$\begin{aligned}\frac{\partial\psi}{\partial t} &+ \mathcal{A}^f_{h1}(\psi)+\mathcal{A}^f_{h2}(\psi)+\mathcal{A}_v(\psi)-\mathcal{C}^f_{s1}(\psi)-\mathcal{C}^f_{s2}(\psi)-\mathcal{C}_{s3}(\psi)\\ &= \mathcal{T}(\psi)+\mathcal{D}_{sv}(\psi)+\mathcal{D}_{sh1}(\psi)+\mathcal{D}_{sh2}(\psi)\end{aligned} \tag{5.389}$$

where $\mathcal{A}^f_{hi}$ are the horizontal advective operators using the filtered currents

$$\mathcal{A}^f_{h1}(\psi) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}(h_2h_3u_f\psi) \tag{5.390}$$

$$\mathcal{A}^f_{h2}(\psi) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}(h_1h_3v_f\psi) \tag{5.391}$$

$$\tag{5.392}$$

5.5.3 Time discretisation

Three cases can be distinguished for the time integration. They are discussed in the subsections below.

5.5.3.1 integration without advection

In the absence of physical advection (iopt_adv_scal=0) the transport equation is integrated in time using

$$\frac{h_3^{n+1}\psi^{n+1} - h_3^n\psi^n}{h_3^{n+1}\Delta t} = \theta_v\mathcal{D}_{sv}(\psi^{n+1}) + (1-\theta_v)\mathcal{D}_{sv}(\psi^n) + \mathcal{T}(\psi^n) + \mathcal{D}_{sh1}(\psi^n) + \mathcal{D}_{sh2}(\psi^n) \tag{5.393}$$

5.5.3.2 integration with advection but without operator splitting

If iopt_adv_scal=1 or 2, the transport equation (5.389) is integrated in time using

$$\begin{aligned}\frac{\psi^{n+1} - \psi^n}{\Delta t} &= -\mathcal{A}_{h1}^f(\psi^n) + \mathcal{C}_{s1}^f(\psi^n) - \mathcal{A}_{h2}^f(\psi^n) + \mathcal{C}_{s2}^f(\psi^n) \\ &\quad - \theta_a\mathcal{A}_v(\psi^{n+1}) - (1-\theta_a)\mathcal{A}_v(\psi^n) + \mathcal{C}_{s3}(\psi^n) + \theta_v\mathcal{D}_{sv}(\psi^{n+1}) \\ &\quad + (1-\theta_v)\mathcal{D}_{sv}(\psi^n) + \mathcal{T}(\psi^n) + \mathcal{D}_{sh1}(\psi^n) + \mathcal{D}_{sh2}(\psi^n)\end{aligned} \tag{5.394}$$

5.5.3.3 integration with operator splitting

If iopt_adv_scal=3, integration is performed along the following steps:

- Part A

$$\frac{\psi_A^{n+1/3} - \psi^n}{\Delta t} = -\mathcal{A}_{h1}^f(\psi^n) + \mathcal{C}_{s1}^f(\psi^n) + \mathcal{D}_{sh1}(\psi^n) \tag{5.395}$$

$$\frac{\psi_A^{n+2/3} - \psi_A^{n+1/3}}{\Delta t} = -\mathcal{A}_{h2}^f(\psi_A^{n+1/3}) + \mathcal{C}_{s2}^f(\psi^n) + \mathcal{D}_{sh2}(\psi_A^{n+1/3}) \tag{5.396}$$

$$\begin{aligned}\frac{\psi_A^{n+1} - \psi_A^{n+2/3}}{\Delta t} &= -\theta_a\mathcal{A}_v(\psi_A^{n+1}) - (1-\theta_a)\mathcal{A}_v(\psi_A^{n+2/3}) + \mathcal{C}_{s3}(\psi^n) \\ &\quad + \theta_v\mathcal{D}_{sv}(\psi_A^{n+1}) + (1-\theta_v)\mathcal{D}_{sv}(\psi_A^{n+2/3}) + \mathcal{T}(\psi^n)\end{aligned} \tag{5.397}$$

- Part B

$$\begin{aligned}\frac{\psi_B^{n+1/3} - \psi^n}{\Delta t} &= -\theta_a\mathcal{A}_v(\psi_B^{n+1/3}) - (1-\theta_a)\mathcal{A}_v(\psi^n) + \mathcal{C}_{s3}(\psi^n) \\ &\quad + \theta_v\mathcal{D}_v(\psi_B^{n+1/3}) + (1-\theta_v)\mathcal{D}_{sv}(\psi^n) + \mathcal{T}(\psi^n)\end{aligned}$$

(5.398)

$$\frac{\psi_B^{n+2/3} - \psi_B^{n+1/3}}{\Delta t} = -\mathcal{A}_{h2}^f(\psi_B^{n+1/3}) + \mathcal{C}_{s2}^f(\psi^n) + \mathcal{D}_{sh2}(\psi_B^{n+1/3}) \quad (5.399)$$

$$\frac{\psi_B^{n+1} - \psi_B^{n+2/3}}{\Delta t} = -\mathcal{A}_{h1}(\psi_B^{n+2/3}) + \mathcal{C}_{s1}^f(\psi^n) + \mathcal{D}_{sh1}(\psi_B^{n+2/3}) \quad (5.400)$$

- Updated value

$$\psi^{n+1} = \frac{1}{2}(\psi_A^{n+1} + \psi_B^{n+1}) \quad (5.401)$$

For the reasons discussed in Section 5.3.3.1 vertical advection is discretised semi-implicitly and vertical diffusion implicitly. The default values, taken for the implicity factors, are then given by $\theta_a = 0.501$, $\theta_v = 1$. The use of the TVD scheme with the operator splitting method increases the CPU time but has the capacity to preserve horizontal and vertical gradients in frontal areas. The user therefore needs to make a balance between CPU time and accuracy when selecting an appropriate scheme.

5.5.4 Discretisation of advection

The advective terms in the scalar transport equations are written as the divergence of the fluxes F_1, F_2, F_3 defined in Table 5.5:

$$\mathcal{A}_{h1}^f(\psi) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2 h_3 u_f \psi\Big) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_1}\Big(h_2 h_3 F_1\Big) \quad (5.402)$$

$$\mathcal{A}_{h2}^f(\psi) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1 h_3 v_f \psi\Big) = \frac{1}{h_1 h_2 h_3}\frac{\partial}{\partial \xi_2}\Big(h_1 h_3 F_2\Big) \quad (5.403)$$

$$\mathcal{A}_v(\psi) = \frac{1}{h_3}\frac{\partial}{\partial s}(\omega\psi) = \frac{1}{h_3}\frac{\partial F_3}{\partial s} \quad (5.404)$$

5.5.4.1 advection in the X-direction

The advective term in the X-direction is obtained by differencing the flux F_1^u at the C-node

$$\mathcal{A}_{h1}^f(\psi)_{ijk}^c = \frac{h_{2;i+1,j}^u h_{3;i+1,jk}^u F_{1;i+1,jk}^u - h_{2;ij}^u h_{3;ijk}^u F_{1;ijk}^u}{h_{1;ij}^c h_{2;ij}^c h_{3;ijk}^c} \quad (5.405)$$

The flux is calculated from

$$F_{1;ij}^u = \Big(1 - \Omega(r_{ijk}^u)\Big) F_{up;ijk}^u + \Omega(r_{ijk}^u) F_{lw;ijk}^u \quad (5.406)$$

where $F^u_{up;ijk}$ and $F^u_{lw;ijk}$ are the upwind and Lax-Wendroff fluxes at the U-node:

$$F^u_{up;ijk} = \frac{1}{2}u_{f;ijk}\Big((\alpha_{ij}+s_{ijk})\psi_{i-1,jk}+(\beta_{ij}-s_{ijk})\psi_{ijk}\Big) \quad (5.407)$$

$$F^u_{lw;ijk} = \frac{1}{2}u_{f;ijk}\Big((\alpha_{ij}+c_{ijk})\psi_{i-1,jk}+(\beta_{ij}-c_{ijk})\psi_{ijk}\Big) \quad (5.408)$$

where s_{ijk} and c_{ijk} are the sign and CFL number of the advecting current

$$s_{ijk} = \mathrm{Sign}(u_{f;ijk})\,, \quad c_{ijk} = \frac{u_{f;ijk}\Delta t}{h^u_{1;ij}} \quad (5.409)$$

$$\alpha_{ij} = \frac{h^c_{1;ij}}{h^u_{1;ij}}\,, \quad \beta_{ij} = \frac{h^c_{1;i-1,j}}{h^u_{1;ij}} \quad (5.410)$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_scal. The argument r of the weight function is defined by

$$r^u_{ijk} = \frac{(\alpha_{ij}+s_{ijk})\Delta F^u_{i-1,jk}+(\beta_{ij}-s_{ijk})\Delta F^u_{i+1,jk}}{2\Delta F^u_{ijk}}$$

$$\Delta F^u_{ijk} = F^u_{lw;ijk}-F^u_{up;ijk} \quad (5.411)$$

5.5.4.2 advection in the Y-direction

The advective term in the Y-direction is obtained by differencing the flux F^v_1 at the C-node

$$\mathcal{A}^f_{h2}(\psi)^c_{ijk} = \frac{h^v_{1;i,j+1}h^v_{3;i,j+1,k}F^v_{2;i,j+1,k}-h^v_{1;ij}h^v_{3;ijk}F^v_{2;ijk}}{h^c_{1;ij}h^c_{2;ij}h^c_{3;ijk}} \quad (5.412)$$

The flux is calculated from

$$F^v_{1;ij} = \Big(1-\Omega(r^v_{ijk})\Big)F^v_{up;ijk}+\Omega(r^v_{ijk})F^v_{lw;ijk} \quad (5.413)$$

where $F^v_{up;ijk}$ and $F^v_{lw;ijk}$ are the upwind and Lax-Wendroff fluxes at the V-node:

$$F^v_{up;ijk} = \frac{1}{2}v_{f;ijk}\Big((\alpha_{ijk}+s_{ijk})\psi_{i,j-1,k}+(\beta_{ijk}-s_{ijk})\psi_{ijk}\Big) \quad (5.414)$$

$$F^v_{lw;ijk} = \frac{1}{2}v_{f;ijk}\Big((\alpha_{ijk}+c_{ijk})\psi_{i,j-1,k}+(\beta_{ijk}-c_{ijk})\psi_{ijk}\Big) \quad (5.415)$$

where s_{ijk} and c_{ijk} are the sign and CFL number of the advecting current

$$s_{ijk} = \text{Sign}(v_{f;ijk}) \,, \quad c_{ijk} = \frac{v_{f;ijk}\Delta t}{h^v_{2;ij}} \tag{5.416}$$

$$\alpha_{ij} = \frac{h^c_{2;ij}}{h^v_{2;ij}} \,, \quad \beta_{ij} = \frac{h^c_{2;i,j-1}}{h^v_{2;ij}} \tag{5.417}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_scal. The argument r of the weight function is defined by

$$\begin{aligned} r^v_{ijk} &= \frac{(\alpha_{ij} + s_{ijk})\Delta F^v_{i,j-1,k} + (\beta_{ij} - s_{ijk})\Delta F^v_{i,j+1,k}}{2\Delta F^v_{ijk}} \\ \Delta F^v_{ijk} &= F^v_{lw;ijk} - F^v_{up;ijk} \end{aligned} \tag{5.418}$$

5.5.4.3 advection in the vertical direction

The vertical advective term is obtained by differencing the flux F^w_3 at the C-node

$$\mathcal{A}_v(\psi)^c_{ijk} = \frac{F^w_{3;ij,k+1} - F^w_{3;ijk}}{h^c_{3;ijk}} \tag{5.419}$$

The flux is calculated from

$$F^w_{3;ijk} = \Big(1 - \Omega(r^w_{ijk})\Big) F^w_{up;ijk} + \Omega(r^w_{ijk}) F^w_{ce;ijk} \tag{5.420}$$

where $F^w_{up;ijk}$ and $F^w_{ce;ijk}$ are the upwind and central fluxes at the W-node:

$$F^w_{up;ijk} = \frac{1}{2}\omega^w_{ijk}\Big((\alpha_{ijk} + s_{ijk})\psi_{ij,k-1} + (\beta_{ijk} - s_{ijk})\psi_{ijk}\Big) \tag{5.421}$$

$$F^w_{ce;ijk} = \frac{1}{2}\omega^w_{ijk}\Big(\alpha_{ijk}\psi_{ij,k-1} + \beta_{ijk}\psi_{ijk}\Big) \tag{5.422}$$

where

$$s_{ijk} = \text{Sign}(\omega^w_{ijk}) \,, \quad \alpha_{ijk} = \frac{h^c_{3;ijk}}{h^w_{3;ijk}} \,, \quad \beta_{ijk} = \frac{h^c_{3;ij,k-1}}{h^w_{3;ijk}} \tag{5.423}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_scal. The argument r of the weight function is defined by

$$\begin{aligned} r^w_{ijk} &= \frac{(\alpha_{ijk} + s_{ijk})\Delta F^w_{ij,k-1} + (\beta_{ijk} - s_{ijk})\Delta F^w_{ij,k+1}}{2\Delta F^w_{ijk}} \\ \Delta F^w_{ijk} &= F^w_{ce;ijk} - F^w_{up;ijk} \end{aligned} \tag{5.424}$$

5.5.4.4 corrector terms

The corrector terms, defined by (5.385)–(5.387), are discretised using

$$\mathcal{C}^f_{s1}(\psi)^c = \psi_{ijk}\frac{h^u_{2;i+1,j}h^u_{3;i+1,jk}u_{f;i+1,jk} - h^u_{2;ij}h^u_{3;ijk}u_{f;ijk}}{h^c_{1;ij}h^c_{2;ij}h^c_{3;ijk}} \quad (5.425)$$

$$\mathcal{C}^f_{s2}(\psi)^c = \psi_{ijk}\frac{h^v_{1;i,j+1}h^v_{3;i,j+1,k}v_{f;i,j+1,k} - h^v_{1;ij}h^v_{3;ijk}v_{f;ijk}}{h^c_{1;ij}h^c_{2;ijk}h^c_{3;ijk}} \quad (5.426)$$

$$\mathcal{C}_{s3}(\psi)^c = \psi_{ijk}\frac{\omega_{ij,k+1} - \omega_{ijk}}{h^c_{3;ijk}} \quad (5.427)$$

5.5.5 Discretisation of diffusion

The diffusive terms in the scalar transport equations are written as the divergence of the fluxes D_1, D_2, D_3 defined in Table 5.5:

$$\mathcal{D}_{sh1}(\psi) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}\Big(\lambda_H\frac{h_2h_3}{h_1}\frac{\partial\psi}{\partial\xi_1}\Big) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_1}\Big(h_2h_3D_1\Big) \quad (5.428)$$

$$\mathcal{D}_{sh2}(\psi) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}\Big(\lambda_H\frac{h_1h_3}{h_2}\frac{\partial\psi}{\partial\xi_2}\Big) = \frac{1}{h_1h_2h_3}\frac{\partial}{\partial\xi_2}\Big(h_1h_3D_2\Big) \quad (5.429)$$

$$\mathcal{D}_{sv}(\psi) = \frac{1}{h_3}\frac{\partial}{\partial s}\Big(\frac{\lambda^\psi_T}{h_3}\frac{\partial\psi}{\partial s}\Big) = \frac{1}{h_3}\frac{\partial D_3}{\partial s} \quad (5.430)$$

5.5.5.1 diffusion in the X-direction

The diffusion term in the X-direction is obtained by differencing the flux D^u_1 at the C-node

$$\mathcal{D}_{sh1}(\psi)^c_{ijk} = \frac{h^u_{2;i+1,j}h^u_{3;i+1,jk}D^u_{1;i+1,jk} - h^u_{2;ij}h^u_{3;ijk}D^u_{1;ijk}}{h^c_{1;ij}h^c_{2;ij}h^c_{3;ijk}} \quad (5.431)$$

The flux is given by

$$D^u_{1;ijk} = \frac{\lambda^u_H(\psi_{ijk} - \psi_{i-1,jk})}{h^u_{1;ij}} \quad (5.432)$$

5.5.5.2 diffusion in the Y-direction

The diffusion term in the Y-direction is obtained by differencing the flux D^v_2 at the C-node

$$\mathcal{D}_{sh2}(\psi)^c_{ijk} = \frac{h^v_{1;i,j+1}h^v_{3;i,j+1,k}D^v_{2;i,j+1,k} - h^v_{1;ij}h^v_{3;ijk}D^v_{2;ijk}}{h^c_{1;ij}h^c_{2;ij}h^c_{3;ijk}} \quad (5.433)$$

The flux is given by

$$D^v_{2;ijk} = \frac{\lambda^v_H(\psi_{ijk} - \psi_{i,j-1,k})}{h^v_{2;ij}} \tag{5.434}$$

5.5.5.3 diffusion in the vertical direction

The vertical diffusion term is obtained by differencing the flux D^w_3 at the C-node

$$\mathcal{D}_{sv}(\psi_{ijk})^c = \frac{D^w_{3;ij,k+1} - D^w_{3;ijk}}{h^c_{3;ijk}} \tag{5.435}$$

The flux is given by

$$D^w_{3;ijk} = \frac{\lambda^{\psi;w}_T(\psi_{ijk} - \psi_{ij,k-1})}{h^w_{3;ijk}} \tag{5.436}$$

5.5.6 Diffusion coefficients for scalars

5.5.6.1 horizontal diffusion coefficients

The discretised values of the horizontal diffusion coefficient at the U- and V-nodes are obtained by applying (4.81) and interpolating $D^c_{T;ijk}$ and $D^{uv}_{S;ijk}$, given by (5.213)–(5.214), to the U- and V-nodes

$$\lambda^u_{H;ijk} = C_s h^u_{1;ij} h^u_{2;ij} \sqrt{\left(D^u_{T;ijk}\right)^2 + \left(D^u_{S;ijk}\right)^2} \tag{5.437}$$

$$\lambda^v_{H;ijk} = C_s h^v_{1;ij} h^v_{2;ij} \sqrt{\left(D^v_{T;ijk}\right)^2 + \left(D^v_{S;ijk}\right)^2} \tag{5.438}$$

5.5.6.2 vertical diffusion coefficient

The vertical diffusion coefficient for scalars λ_T is obtained from one of the available turbulence schemes, described in Section 4.4. Values are first stored at the W-nodes and interpolated afterwards at the U- and V-nodes for the calculation of the vertical diffusion fluxes in the momentum equations. The evaluation of λ_T only involves algebraic expressions so that the discretisation procedure is straightforward.

For further comments see Section 5.3.12.2.

5.5.7 Boundary conditions

5.5.7.1 surface boundary conditions

The program allows four type of boundary conditions

1. Neumann condition with a prescribed (downward) surface flux $F^{\psi;w}_{s;ij}$

$$D^{w}_{3;ijN_z} = F^{\psi;w}_{s;ij} \tag{5.439}$$

2. Neumann condition using a surface transfer velocity

$$D^{w}_{3;ijN_z} = C^{\psi}_{s;ij}\Big(\psi^{w}_{s;ij} - (1-\theta_v)\psi_{ijN_z} - \theta_v\psi^{n+1}_{ijN_z}\Big) \tag{5.440}$$

where $C^{\psi}_{s;ij}$ is the transfer velocity and $\psi^{w}_{s;ij}$ a prescribed external value.

3. Dirichlet condition with a prescribed external value $\psi^{c}_{s;ij}$ at the first C-node below the surface

$$\psi^{n+1}_{ijN_z} = \psi^{c}_{s;ij} \tag{5.441}$$

4. Dirichlet condition with a prescribed external value $\psi^{w}_{s;ij}$ at the surface itself. In that case the value at the first node below the surface is determined by interpolation

$$\psi^{n+1}_{ijN_z} = \frac{2h^{w}_{3;ijN_z}\psi^{w}_{s;ij} + h^{c}_{3;ijN_z}\psi^{n+1}_{ij,n_z-1}}{2h^{w}_{3;ijN_z} + h^{c}_{3;ij,N_z}} \tag{5.442}$$

which can be more conveniently rewritten in "tridiagonal" form

$$-\frac{h^{c}_{3;ijN_z}}{2h^{w}_{3;ijN_z} + h^{c}_{3;ijN_z}}\psi^{n+1}_{ij,N_z-1} + \psi^{n+1}_{ijN_z} = \frac{2h^{w}_{3;ijN_z}}{2h^{w}_{3;ijN_z} + h^{c}_{3;ijN_z}}\psi^{w}_{s;ij} \tag{5.443}$$

Note that the second Neumann condition is semi-implicit whereas both Dirichlet conditions use a fully implicit formulation.

5.5.7.2 bottom boundary conditions

The bottom boundary conditions are similar to the ones at the surface.

1. Neumann condition with a prescribed (upward) bottom flux $F^{\psi;w}_{b;ij}$

$$D^{w}_{3;ij1} = F^{\psi;w}_{b;ij} \tag{5.444}$$

2. Neumann condition using a bottom transfer velocity

$$D^{w}_{3;ij1} = C^{\psi}_{b;ij}\Big((1-\theta_v)\psi_{ij1} + \theta_v\psi^{n+1}_{ij1} - \psi^{w}_{b;ij}\Big) \tag{5.445}$$

where $C^{\psi}_{b;ij}$ is the transfer velocity and $\psi^{w}_{b;ij}$ a prescribed external value.

3. Dirichlet condition with a prescribed external value $\psi^c_{b;ij}$ at the first C-node above the bottom

$$\psi^{n+1}_{ij1} = \psi^c_{b;ij} \tag{5.446}$$

4. Dirichlet condition with a prescribed external value $\psi^w_{b;ij}$ at the bottom itself. In that case the value at the first node above the sea bed is determined by interpolation

$$\psi^{n+1}_{ij1} = \frac{2h^w_{3;ij2}\psi^w_{b;ij} + h^c_{3;ij1}\psi^{n+1}_{ij2}}{2h^w_{3;ij2} + h^c_{3;ij1}} \tag{5.447}$$

which can be more conveniently rewritten in "tridiagonal" form

$$\psi^{n+1}_{ij1} - \frac{h^c_{3;ij1}}{2h^w_{3;ij2} + h^c_{3;ij1}}\psi^{n+1}_{ij2} = \frac{2h^w_{3;ij2}}{2h^w_{3;ij2} + h^c_{3;ij1}}\psi^w_{b;ij} \tag{5.448}$$

Note that the second Neumann condition is semi-implicit whereas both Dirichlet conditions use a fully implicit formulation.

5.5.7.3 lateral boundary conditions

At the open boundaries the flux normal to the boundary is determined by the upwind scheme. This means that

$$F^u_{1;ijk} = \frac{1}{2}u_{f;ijk}\Big((1 \pm s_{ijk})\psi^e_{ijk} + (1 \mp s_{ijk})\psi_{i:i-1,jk}\Big) \tag{5.449}$$

at U-boundaries and

$$F^v_{2;ijk} = \frac{1}{2}v_{f;ijk}\Big((1 \pm s_{ijk})\psi^e_{ijk} + (1 \mp s_{ijk})\psi_{i,j:j-1,k}\Big) \tag{5.450}$$

at V-boundaries, where the upper (lower) sign applies at western/southern (eastern/northern) boundaries, the flow sign s_{ijk} is defined by (5.409) or (5.416) and ψ^e_{ijk} denotes an external profile of ψ at one-half grid distance outside the open boundary. The open boundary problem then consists in determining the external profile ψ^e. The following four methods are available

1. Zero gradient condition.

$$\psi^e_{ijk} = \psi_{i:i-1,jk} \quad \text{or} \quad F^u_{1;ijk} = u_{f;ijk}\psi_{i:i-1,jk} \tag{5.451}$$
$$\psi^e_{ijk} = \psi_{i,j:j-1,k} \quad \text{or} \quad F^v_{2;ijk} = v_{f;ijk}\psi_{i,j:j-1,k} \tag{5.452}$$

2. The external profile is prescribed.

3. Radiation condition using the baroclinic wave speed.

$$\psi^{e;u;n+1}_{ijk} = (1 - w^u_{ijk})\psi^{e;u;n}_{ijk} + w^u_{ijk}\psi^n_{i:i-1,jk} \tag{5.453}$$
$$\psi^{e;v;n+1}_{ijk} = (1 - w^v_{ijk})\psi^{e;v;n}_{ijk} + w^v_{ijk}\psi^n_{i,j:j-1,k} \tag{5.454}$$

The weight factors are given by

$$\begin{aligned} w^u_{ijk} &= \pm\frac{R\Delta t}{h^u_{1;ij}}\sqrt{g^u_{ij}H^{n+1;c}_{i:i-1,j}} \\ w^v_{ijk} &= \pm\frac{R\Delta t}{h^v_{2;ij}}\sqrt{g^v_{ij}H^{n+1;c}_{i,j:j-1}} \end{aligned} \tag{5.455}$$

where R is the prescribed ratio of the baroclinic to surface gravity wave speed. Default value is 0.03.

4. Orlanski condition (see equation (4.383)).

$$\begin{aligned} \psi^{e;u;n+1}_{ijk} &= \Big(1 - O_R(r^u_{1;ijk}, r^u_{2;ijk}, r^u_{3;ijk})\Big)\psi^{e;u;n}_{ijk} \\ &\quad + O_R(r^u_{1;ijk}, r^u_{2;ijk}, r^u_{3;ijk})\psi^n_{i:i-1,jk} \end{aligned} \tag{5.456}$$
$$\begin{aligned} \psi^{e;v;n+1}_{ijk} &= \Big(1 - O_R(r^v_{1;ijk}, r^v_{2;ijk}, r^v_{3;ijk})\Big)\psi^{e;v;n}_{ijk} \\ &\quad + O_R(r^v_{1;ijk}, r^v_{2;ijk}, r^v_{3;ijk})\psi^n_{i,j:j-1,k} \end{aligned} \tag{5.457}$$

where the Orlanski function O_R is defined by (5.269) and

$$r^u_{1;ijk} = \psi^n_{i:i-1,jk}\,, \quad r^u_{2;ijk} = \psi^{n-1}_{i:i-1,jk}\,, \quad r^u_{3;ijk} = \psi^{n-1}_{i+1:i-2,jk} \tag{5.458}$$

$$r^v_{1;ijk} = \psi^n_{i,j:j-1,k}\,, \quad r^v_{2;ijk} = \psi^{n-1}_{i,j:j-1,k}\,, \quad r^v_{3;ijk} = \psi^{n-1}_{i,j+1:j-2,k} \tag{5.459}$$

Advective fluxes normal to a closed (coastal) open boundary are set to zero.

5.5.8 Solution of the discretised equations for scalars

As for momentum, the discretised equations can be written in tridiagonal form, as given by (5.315). Expressions for the matrix components are given below for the case that no operator splitting is used. They are easily extended to the case with operator splitting.

When a Dirichlet boundary condition is used at the surface (bottom), the surface (bottom) value of ψ is determined by the boundary condition itself.

This means that k below varies between k_{min} and k_{max} where k_{min} equals 1 for a Neumann (flux) condition and 2 for a Dirichlet condition at the bottom. Likewise, k_{max} equals N_z for a Neumann and $N_z - 1$ for a Dirichlet condition at the surface.

For simplicity, the i and j indices are omitted.

1. Time derivative.

   The contribution of the time derivative is given by

$$A_k^t = 0\,, \quad B_k^t = 1\,, \quad C_k^t = 0\,, \quad D_k^t = \psi_k^n \tag{5.460}$$

   where $k_{min} \le k \le k_{max}$.

2. Vertical advection.

   The vertical advection term is split up into two contributions arising from the fluxes below and above a k-level. The former are given by

$$\begin{aligned} A_k^{a-} &= -\theta_a c_k^- (\alpha_k + f_k) \\ B_k^{a-} &= -\theta_a c_k^- (\beta_k - f_k) \\ C_k^{a-} &= 0 \\ D_k^{a-} &= (1-\theta_a) c_k^- \Big((\alpha_k + f_k)\psi_{k-1}^n + (\beta_k - f_k)\psi_k^n \Big) \end{aligned} \tag{5.461}$$

   where $2 \le k \le k_{max}$,

$$c_k^- = \frac{\Delta t \omega_k^w}{2h_{3;k}^c}\,, \quad f_k = \Big(1 - \Omega(r_k^w)\Big) s_k \tag{5.462}$$

   and α_{ijk}, β_{ijk}, s_{ijk}, r_{ijk}^w are defined by (5.423) and (5.424).

   The terms arising from the flux above the k-level, are

$$\begin{aligned} A_k^{a+} &= 0 \\ B_k^{a+} &= \theta_a c_k^+ (\alpha_{k+1} + f_{k+1}) \\ C_k^{a+} &= \theta_a c_k^+ (\beta_{k+1} - f_{k+1}) \\ D_k^{a+} &= -(1-\theta_a) c_k^+ \Big((\alpha_{k+1} + f_{k+1})\psi_k^n + (\beta_{k+1} - f_{k+1})\psi_{k+1}^n \Big) \end{aligned} \tag{5.463}$$

   where $k_{min} \le k \le N_z - 1$ and

$$c_k^+ = \frac{\Delta t \omega_{k+1}^w}{2h_{3;k}^c} \tag{5.464}$$

3. Vertical diffusion.

   As for vertical advection the fluxes below and above a k-level are taken separately. The former are given by

$$\begin{aligned} A_k^{d-} &= -\theta_v \frac{\Delta t \lambda_{T;k}^{\psi;w}}{h_{3;k}^c h_{3;k}^w} \\ B_k^{d-} &= \theta_v \frac{\Delta t \lambda_{T;k}^{\psi;w}}{h_{3;k}^c h_{3;k}^w} \\ C_k^{d-} &= 0 \\ D_k^{d-} &= -(1-\theta_v) \frac{\Delta t \lambda_{T;k}^{\psi;w}}{h_{3;k}^c h_{3;k}^w} (\psi_k^n - \psi_{k-1}^n) \end{aligned} \tag{5.465}$$

   where $2 \leq k \leq k_{max}$.

   The terms taken from the flux above the k-level, are

$$\begin{aligned} A_k^{d+} &= 0 \\ B_k^{d+} &= \theta_v \frac{\Delta t \lambda_{T;k+1}^{\psi;w}}{h_{3;k}^c h_{3;k+1}^w} \\ C_k^{d+} &= -\theta_v \frac{\Delta t \lambda_{T;k+1}^{\psi;w}}{h_{3;k}^c h_{3;k+1}^w} \\ D_k^{d+} &= (1-\theta_v) \frac{\Delta t \lambda_{T;k+1}^{\psi;w}}{h_{3;k}^c h_{3;k+1}^w} (\psi_{k+1}^n - \psi_k^n) \end{aligned} \tag{5.466}$$

   where $k_{min} \leq k \leq N_z - 1$.

4. Other explicit terms.

   All other terms are explicit. Their contributions can be written as

$$\begin{aligned} A_k^e &= B_k^e = C_k^e = 0 \\ D_k^e &= \Delta t \Big(\mathcal{T}_k^n - \tilde{\mathcal{A}}_{h1}^f(\psi)_k^n - \tilde{\mathcal{A}}_{h2}^f(\psi)_k^n + \mathcal{C}_{s1}^f(\psi)_k^n \\ &\quad + \mathcal{C}_{s2}^f(\psi)_k^n + \mathcal{C}_{s3}(\psi)_k^n + \mathcal{D}_{sh1}(\psi)_k^n + \mathcal{D}_{sh2}(\psi)_k^n \Big) \end{aligned} \tag{5.467}$$

   where $k_{min} \leq k \leq k_{max}$.

5. Surface boundary conditions.

   Contributions from the surface boundary conditions depends on the type of condition as described in Section 5.5.7.1.

- Neumann condition with a prescribed surface flux $F_s^{\psi;w}$.

$$A^s_{N_z} = B^s_{N_z} = C^s_{N_z} = 0\,, \quad D^s_{N_z} = \frac{\Delta t F_s^{\psi;w}}{h^c_{3;N_z}} \tag{5.468}$$

- Neumann condition using a surface transfer velocity.

$$A^s_{N_z} = C^s_{N_z} = 0\,, \quad B^s_{N_z} = \theta_v \frac{\Delta t C_s^{\psi}}{h^c_{3;N_z}}\,, \quad D^s_{N_z} = \frac{\Delta t C_s^{\psi}}{h^c_{3;N_z}}\Big(\psi_s^w - (1-\theta_v)\psi^n_{N_z}\Big) \tag{5.469}$$

- Dirichlet condition with a prescribed external value ψ_s^c at the first node below the surface.

$$A^s_{N_z} = C^s_{N_z} = 0\,, \quad B^s_{N_z} = 1\,, \quad D^s_{N_z} = \psi_s^c \tag{5.470}$$

- Dirichlet condition with a prescribed external value ψ_s^w at the surface itself.

$$\begin{aligned} A^s_{N_z} &= -\frac{h^c_{3;N_z}}{2h^w_{3;N_z} + h^c_{3;N_z}} \\ B^s_{N_z} &= 1\,, \quad C^s_{N_z} = 0 \\ D^s_{N_z} &= \frac{2h^w_{3;N_z}\psi_s^w}{2h^w_{3;N_z} + h^c_{3;N_z}} \end{aligned} \tag{5.471}$$

6. Bottom boundary conditions.

   Contributions from the bottom boundary conditions depends on the type of condition as described in Section 5.5.7.2.

- Neumann condition with a prescribed bottom flux $F_b^{\psi;w}$.

$$A^b_1 = B^b_1 = C^b_1 = 0\,, \quad D^b_1 = \frac{\Delta t F_b^{\psi;w}}{h^c_{3;1}} \tag{5.472}$$

- Neumann condition using a bottom transfer velocity.

$$A^b_1 = C^b_1 = 0\,, \quad B^b_1 = \theta_v \frac{\Delta t C_b^{\psi}}{h^c_{3;1}}\,, \quad D^b_1 = \frac{\Delta t C_b^{\psi}}{h^c_{3;1}}\Big(\psi_b^w - (1-\theta_v)\psi^n_1\Big) \tag{5.473}$$

- Dirichlet condition with a prescribed external value ψ_b^c at the first node above the sea bed.

$$A^b_1 = C^b_1 = 0\,, \quad B^b_1 = 1\,, \quad D^b_1 = \psi_b^c \tag{5.474}$$

- Dirichlet condition with a prescribed external value ψ_b^w at the bottom itself.

$$\begin{aligned} A_1^b &= 0\,, \quad B_1^b = 1 \\ C_1^b &= -\frac{h_{3;1}^c}{2h_{3;2}^w + h_{3;1}^c} \\ D_1^b &= \frac{2h_{3;2}^w \psi_b^w}{2h_{3;2}^w + h_{3;1}^c} \end{aligned} \tag{5.475}$$

5.6 Turbulence transport equations

This section deals with the numerical solution of the transport equations (4.204) for turbulence energy k, (4.205) for the dissipation rate ε and (4.209) for the turbulent energy times mixing length kl. The discretisation algorithms are highly similar to the ones used for scalar quantities. Main differences are

- Turbulence variables are determined at W-nodes.
- Production terms are taken explicitly at the old time step t^n, whereas the sink terms are discretised in time using the "quasi-implicit" approach, proposed by Patankar (1980), or

$$\mathcal{P}(\psi) = \mathcal{P}(\psi^n)\,, \quad \mathcal{S}(\psi) = \mathcal{P}(\psi^n)\frac{\psi^{n+1}}{\psi^n} \tag{5.476}$$

- Since the turbulent equations are solved before the momentum equations, the horizontal advective terms are discretised using the non-filtered current (u,v).
- The discretisation algorithms for advection and diffusion are the same as for C-node scalar quantities, except that all quantities are displaced in the vertical. This means that 3-D variables (fluxes, advective velocities, diffusion coefficients, ...), previously evaluated at (C,U,V,W,UW, VW)-nodes are now taken at respectively (W,UW,VW,C,U,V)-nodes.

The turbulence transport equations are generally written as

$$\begin{aligned} \frac{\partial \psi}{\partial t} &+ \mathcal{A}_{h1}(\psi) + \mathcal{A}_{h2}(\psi) + \mathcal{A}_v(\psi) - \mathcal{C}_{s1}(\psi) - \mathcal{C}_{s2}(\psi) - \mathcal{C}_{s3}(\psi) \\ &= \mathcal{P}(\psi) - \mathcal{S}(\psi) + \mathcal{D}_{sv}(\psi) + \mathcal{D}_{sh1}(\psi) + \mathcal{D}_{sh2}(\psi) \end{aligned} \tag{5.477}$$

where the corrector and horizontal advective operators $\mathcal{C}_{si}$ and $\mathcal{A}_{hi}$ are given by (5.385)–(5.387) and (5.390)–(5.391) with (u_f,v_f) replaced by (u,v).

Despite the similarities with the previous section, the discretisation methods are described in detail below, to avoid any confusion.

5.6.1 Time discretisation

Three cases can be distinguished for the time integration. They are discussed in the subsections below.

5.6.1.1 integration without advection

In the absence of physical advection (iopt_adv_scal=0) the transport equation is integrated in time using

$$\begin{aligned}\frac{h_3^{n+1;w}\psi^{n+1;w} - h_3^{n;w}\psi^{n;w}}{h_3^{n+1;w}\Delta t} &= \theta_v\mathcal{D}_{sv}(\psi^{n+1;w}) + (1-\theta_v)\mathcal{D}_{sv}(\psi^{n;w}) + \mathcal{P}(\psi^{n;w}) \\ &\quad - \mathcal{S}(\psi^{n;w})\frac{\psi^{n+1;w}}{\psi^{n;w}} + \mathcal{D}_{sh1}(\psi^{n;w}) + \mathcal{D}_{sh2}(\psi^{n;w})\end{aligned} \tag{5.478}$$

5.6.1.2 integration with advection but without operator splitting

If iopt_adv_turb=1 or 2, the transport equation (5.477) is integrated in time using

$$\begin{aligned}\frac{\psi^{n+1;w} - \psi^{n;w}}{\Delta t} &= -\mathcal{A}_{h1}(\psi^{n;w}) + \mathcal{C}_{s1}(\psi^{n;w}) - \mathcal{A}_{h2}(\psi^{n;w}) + \mathcal{C}_{s2}(\psi^{n;w}) \\ &\quad - \theta_a\mathcal{A}_v(\psi^{n+1;w}) - (1-\theta_a)\mathcal{A}_v(\psi^{n;w}) + \mathcal{C}_{s3}(\psi^{n;w}) \\ &\quad + \theta_v\mathcal{D}_{sv}(\psi^{n+1;w}) + (1-\theta_v)\mathcal{D}_{sv}(\psi^{n;w}) + \mathcal{P}(\psi^{n;w}) \\ &\quad - \mathcal{S}(\psi^{n;w})\frac{\psi^{n+1;w}}{\psi^{n;w}} + \mathcal{D}_{sh1}(\psi^{n;w}) + \mathcal{D}_{sh2}(\psi^{n;w})\end{aligned} \tag{5.479}$$

5.6.1.3 integration with operator splitting

If iopt_adv_turb=3, integration is performed along the following steps:

- Part A

$$\frac{\psi_A^{n+1/3;w} - \psi^{n;w}}{\Delta t} = -\mathcal{A}_{h1}(\psi^{n;w}) + \mathcal{C}_{s1}(\psi^{n;w}) + \mathcal{D}_{sh1}(\psi^{n;w}) \tag{5.480}$$

$$\frac{\psi_A^{n+2/3;w} - \psi_A^{n+1/3;w}}{\Delta t} = -\mathcal{A}_{h2}(\psi_A^{n+1/3;w}) + \mathcal{C}_{s2}(\psi^{n:w}) + \mathcal{D}_{sh2}(\psi_A^{n+1/3;w}) \tag{5.481}$$

$$\begin{aligned}\frac{\psi_A^{n+1;w} - \psi_A^{n+2/3;w}}{\Delta t} = & -\theta_a \mathcal{A}_v(\psi_A^{n+1;w}) - (1-\theta_a)\mathcal{A}_v(\psi_A^{n+2/3;w}) \\ & + \mathcal{C}_{s3}(\psi^{n;w}) + \theta_v \mathcal{D}_{sv}(\psi_A^{n+1;w}) \\ & + (1-\theta_v)\mathcal{D}_{sv}(\psi_A^{n+2/3;w}) + \mathcal{P}(\psi^{n;w}) \\ & - \mathcal{S}(\psi^{n+2/3;w})\frac{\psi^{n+1;w}}{\psi^{n+2/3;w}}\end{aligned} \tag{5.482}$$

- Part B

$$\begin{aligned}\frac{\psi_B^{n+1/3;w} - \psi^{n;w}}{\Delta t} = & -\theta_a \mathcal{A}_v(\psi_B^{n+1/3;w}) - (1-\theta_a)\mathcal{A}_v(\psi^{n;w}) \\ & + \mathcal{C}_{s3}(\psi^{n;w}) + \theta_v \mathcal{D}_{sv}(\psi_B^{n+1/3;w}) \\ & + (1-\theta_v)\mathcal{D}_{sv}(\psi^{n;w}) + \mathcal{P}(\psi^{n;w}) \\ & - \mathcal{S}(\psi^{n;w})\frac{\psi^{n+1/3;w}}{\psi^{n;w}}\end{aligned} \tag{5.483}$$

$$\frac{\psi_B^{n+2/3;w} - \psi_B^{n+1/3;w}}{\Delta t} = -\mathcal{A}_{h2}(\psi_B^{n+1/3;w}) + \mathcal{C}_{s2}(\psi^{n;w}) + \mathcal{D}_{sh2}(\psi_B^{n+1/3;w}) \tag{5.484}$$

$$\frac{\psi_B^{n+1;w} - \psi_B^{n+2/3;w}}{\Delta t} = -\mathcal{A}_{h1}(\psi_B^{n+2/3;w}) + \mathcal{C}_{s1}(\psi^{n;w}) + \mathcal{D}_{sh1}(\psi_B^{n+2/3;w}) \tag{5.485}$$

- Updated value

$$\psi^{n+1;w} = \frac{1}{2}(\psi_A^{n+1;w} + \psi_B^{n+1;w}) \tag{5.486}$$

As before, the implicity factors are given by $\theta_a = 0.501$, $\theta_v = 1$.

5.6.2 Discretisation of advection

5.6.2.1 advection in the X-direction

The advective term in the X-direction is obtained by differencing the flux F_1^{uw} at the W-node

$$\mathcal{A}_{h1}(\psi)_{ijk}^w = \frac{h_{2;i+1,j}^u h_{3;i+1,jk}^{uw} F_{1;i+1,jk}^{uw} - h_{2;ij}^u h_{3;ijk}^{uw} F_{1;ijk}^{uw}}{h_{1;ij}^c h_{2;ij}^c h_{3;ijk}^w} \tag{5.487}$$

The flux is calculated from

$$F^{uw}_{1;ij} = \Big(1 - \Omega(r^{uw}_{ijk})\Big) F^{uw}_{up;ijk} + \Omega(r^{uw}_{ijk}) F^{uw}_{lw;ijk} \tag{5.488}$$

where $F^{uw}_{up;ijk}$ and $F^{uw}_{lw;ijk}$ are the upwind and Lax-Wendroff fluxes at the UW-node:

$$F^{uw}_{up;ijk} = \frac{1}{2} u^{uw}_{ijk} \Big((\alpha_{ij} + s_{ijk}) \psi^{w}_{i-1,jk} + (\beta_{ij} - s_{ijk}) \psi^{w}_{ijk} \Big) \tag{5.489}$$

$$F^{uw}_{lw;ijk} = \frac{1}{2} u^{uw}_{ijk} \Big((\alpha_{ij} + c_{ijk}) \psi^{w}_{i-1,jk} + (\beta_{ij} - c_{ijk}) \psi^{w}_{ijk} \Big) \tag{5.490}$$

where s_{ijk} and c_{ijk} are the sign and CFL number of the advecting current

$$s_{ijk} = \mathrm{Sign}(u^{uw}_{ijk}) \,, \quad c_{ijk} = \frac{u^{uw}_{ijk} \Delta t}{h^{u}_{1;ij}} \tag{5.491}$$

$$\alpha_{ij} = \frac{h^{c}_{1;ij}}{h^{u}_{1;ij}} \,, \quad \beta_{ij} = \frac{h^{c}_{1;i-1,j}}{h^{u}_{1;ij}} \tag{5.492}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_turb. The argument r of the weight function is defined by

$$\begin{aligned} r^{u}_{ijk} &= \frac{(\alpha_{ij} + s_{ijk}) \Delta F^{uw}_{i-1,jk} + (\beta_{ij} - s_{ijk}) \Delta F^{uw}_{i+1,jk}}{2 \Delta F^{uw}_{ijk}} \\ \Delta F^{uw}_{ijk} &= F^{uw}_{lw;ijk} - F^{uw}_{up;ijk} \end{aligned} \tag{5.493}$$

5.6.2.2 advection in the Y-direction

The advective term in the Y-direction is obtained by differencing the flux F^{vw}_{1} at the W-node

$$\mathcal{A}_{h2}(\psi)^{w}_{ijk} = \frac{h^{v}_{1;i,j+1} h^{vw}_{3;i,j+1,k} F^{vw}_{2;i,j+1,k} - h^{v}_{1;ij} h^{vw}_{3;ijk} F^{vw}_{2;ijk}}{h^{c}_{1;ij} h^{c}_{2;ij} h^{w}_{3;ijk}} \tag{5.494}$$

The flux is calculated from

$$F^{vw}_{1;ij} = \Big(1 - \Omega(r^{vw}_{ijk})\Big) F^{vw}_{up;ijk} + \Omega(r^{vw}_{ijk}) F^{vw}_{lw;ijk} \tag{5.495}$$

where $F^{vw}_{up;ijk}$ and $F^{vw}_{lw;ijk}$ are the upwind and Lax-Wendroff fluxes at the VW-node:

$$F^{vw}_{up;ijk} = \frac{1}{2} v^{vw}_{ijk} \Big((\alpha_{ij} + s_{ijk}) \psi^{w}_{i,j-1,k} + (\beta_{ij} - s_{ijk}) \psi^{w}_{ijk} \Big) \tag{5.496}$$

$$F^{vw}_{lw;ijk} = \frac{1}{2} v^{vw}_{ijk} \Big((\alpha_{ij} + c_{ijk}) \psi^{w}_{i,j-1,k} + (\beta_{ij} - c_{ijk}) \psi^{w}_{ijk} \Big) \tag{5.497}$$

where s_{ijk} and c_{ijk} are the sign and CFL number of the advecting current

$$s_{ijk} = \mathrm{Sign}(v^{vw}_{ijk}) \,, \quad c_{ijk} = \frac{v^{vw}_{ijk} \Delta t}{h^{v}_{2;ij}} \tag{5.498}$$

$$\alpha_{ij} = \frac{h^{c}_{2;ij}}{h^{v}_{2;ij}} \,, \quad \beta_{ij} = \frac{h^{c}_{2;i,j-1}}{h^{v}_{2;ij}} \tag{5.499}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_turb. The argument r of the weight function is defined by

$$\begin{aligned} r^{vw}_{ijk} &= \frac{(\alpha_{ij} + s_{ijk}) \Delta F^{vw}_{i,j-1,k} + (\beta_{ij} - s_{ijk}) \Delta F^{vw}_{i,j+1,k}}{2 \Delta F^{vw}_{ijk}} \\ \Delta F^{vw}_{ijk} &= F^{vw}_{lw;ijk} - F^{vw}_{up;ijk} \end{aligned} \tag{5.500}$$

5.6.2.3 advection in the vertical direction

The vertical advective term is obtained by differencing the flux F^{c}_{3} at the W-node

$$\mathcal{A}_v(\psi)^{w}_{ijk} = \frac{F^{c}_{3;ijk} - F^{c}_{3;ij,k-1}}{h^{w}_{3;ijk}} \tag{5.501}$$

The flux is calculated from

$$F^{c}_{3;ijk} = \big(1 - \Omega(r^{c}_{ijk}) \big) \, F^{c}_{up;ijk} + \Omega(r^{c}_{ijk}) F^{c}_{ce;ijk} \tag{5.502}$$

where $F^{c}_{up;ijk}$ and $F^{c}_{ce;ijk}$ are the upwind and central fluxes at the C-node:

$$F^{c}_{up;ijk} = \frac{1}{2} \omega^{c}_{ijk} \Big((1 + s_{ijk}) \psi^{w}_{ijk} + (1 - s_{ijk}) \psi^{w}_{ij,k+1} \Big) \tag{5.503}$$

$$F^{c}_{ce;ijk} = \frac{1}{2} \omega^{c}_{ijk} (\psi^{w}_{ij,k} + \psi^{w}_{ij,k+1}) \tag{5.504}$$

where

$$s_{ijk} = \mathrm{Sign}(\omega^{c}_{ijk}) \tag{5.505}$$

The form of the weighting function is given by (5.50)–(5.53), depending on the type of advection scheme, selected by the switch iopt_adv_scal. The argument r of the weight function is defined by

$$\begin{aligned} r^{w}_{ijk} &= \frac{(1 + s_{ijk}) \Delta F^{c}_{ij,k-1} + (1 - s_{ijk}) \Delta F^{c}_{ij,k+1}}{2 \Delta F^{c}_{ijk}} \\ \Delta F^{c}_{ijk} &= F^{c}_{ce;ijk} - F^{c}_{up;ijk} \end{aligned} \tag{5.506}$$

5.6.2.4 corrector terms

The corrector terms are discretised using

$$\mathcal{C}_{s1}(\psi)^w = \psi^w_{ijk}\frac{h^u_{2;i+1,j}h^{uw}_{3;i+1,jk}u^{uw}_{i+1,jk} - h^u_{2;ij}h^{uw}_{3;ijk}u^{uw}_{ijk}}{h^c_{1;ij}h^c_{2;ij}h^w_{3;ijk}} \tag{5.507}$$

$$\mathcal{C}_{s2}(\psi)^w = \psi^w_{ijk}\frac{h^v_{1;i,j+1}h^{vw}_{3;i,j+1,k}v^{vw}_{i,j+1,k} - h^v_{1;ij}h^{vw}_{3;ijk}v^{vw}_{ijk}}{h^c_{1;ij}h^c_{2;ij}h^w_{3;ijk}} \tag{5.508}$$

$$\mathcal{C}_{s3}(\psi)^w = \psi^w_{ijk}\frac{\omega^c_{ij,k+1} - \omega^c_{ijk}}{h^w_{3;ijk}} \tag{5.509}$$

5.6.3 Discretisation of diffusion

5.6.3.1 diffusion in the X-direction

The diffusion term in the X-direction is obtained by differencing the flux D^{uw}_1 at the W-node

$$\mathcal{D}_{sh1}(\psi)^w_{ijk} = \frac{h^u_{2;i+1,j}h^{uw}_{3;i+1,jk}D^{uw}_{1;i+1,jk} - h^u_{2;ij}h^{uw}_{3;ijk}D^{uw}_{1;ijk}}{h^c_{1;ij}h^c_{2;ij}h^w_{3;ijk}} \tag{5.510}$$

The flux is given by

$$D^{uw}_{1;ijk} = \frac{\lambda^{uw}_H(\psi^w_{ijk} - \psi^w_{i-1,jk})}{h^u_{1;ij}} \tag{5.511}$$

5.6.3.2 diffusion in the Y-direction

The diffusion term in the Y-direction is obtained by differencing the flux D^w_2 at the W-node

$$\mathcal{D}_{sh2}(\psi)^w_{ijk} = \frac{h^v_{1;i,j+1}h^{vw}_{3;i,j+1,k}D^{vw}_{2;i,j+1,k} - h^v_{1;ij}h^{vw}_{3;ijk}D^{vw}_{2;ijk}}{h^c_{1;ij}h^c_{2;ij}h^w_{3;ijk}} \tag{5.512}$$

The flux is given by

$$D^{vw}_{2;ijk} = \frac{\lambda^{vw}_H(\psi^w_{ijk} - \psi^w_{i,j-1,k})}{h^v_{2;ij}} \tag{5.513}$$

5.6.3.3 diffusion in the vertical direction

The vertical diffusion term is obtained by differencing the flux D^c_3 at the W-node

$$\mathcal{D}_{sv}(\psi_{ijk})^w = \frac{D^c_{3;ijk} - D^c_{3;ij,k-1}}{h^w_{3;ijk}} \tag{5.514}$$

The flux is given by

$$D^c_{3;ijk} = \frac{\lambda^{\psi;c}_T(\psi^w_{ij,k+1} - \psi^w_{ijk})}{h^c_{3;ijk}} \tag{5.515}$$

5.6.4 Diffusion coefficients for turbulence variables

5.6.4.1 horizontal diffusion coefficients

The horizontal turbulent diffusion coefficients are the same as the one used for scalar transport. They are obtained by vertical interpolation of λ^u_H and λ^v_H, given by (5.437–(5.438) to respectively the UW and VW-nodes.

5.6.4.2 vertical diffusion coefficients

The vertical turbulent diffusion coefficients, used in the ε-equation (4.205) and kl-equation (4.209), are proportional to ν_k which is the one used in the k-equation (4.204). Different formulations for the parameterisation of ν_k are available and discussed in Section 4.4.3.3. No specific discretisation procedures are required since all expressions are purely algebraic.

The following remarks are to be given

- Values are first obtained at the W-nodes and then interpolated at the C-nodes.
- No value is calculated at the surface and the bottom.
- Since ν_k is calculated prior to the solution of the turbulent transport equations, its value is obtained using values of k, ε and l at the old time t^n.

5.6.5 Production and sink terms

All turbulence transport equations contain, besides the diffusion terms, three terms on their right hand side. The first is the shear production term, the second is the buoyancy term which is a production or sink term if $N^2 < 0$ or $N^2 > 0$ and the third is a dissipation (sink) term. Defining

$$N^2 = \max(N^2, 0) + \min(N^2, 0) = N^2_+ - N^2_- \tag{5.516}$$

one has

$$\mathcal{P}(k) \;=\; \nu_T M^2 + \lambda_T N^2_-$$

$$\begin{aligned}\mathcal{P}(\varepsilon) &= c_{1\varepsilon}\frac{\epsilon}{k}\Big(\nu_T M^2 + c_{3\varepsilon}\lambda_T N_-^2\Big)\\ \mathcal{P}(kl) &= \frac{1}{2}l\Big(E_1\nu_T M^2 + E_3\lambda_T N_-^2\Big)\end{aligned} \tag{5.517}$$

and

$$\begin{aligned}\mathcal{S}(k) &= \lambda_T N_+^2 + \varepsilon\\ \mathcal{S}(\varepsilon) &= c_{1\varepsilon}c_{3\varepsilon}\lambda_T N_+^2 + c_{2\varepsilon}\frac{\varepsilon^2}{k}\\ \mathcal{S}(kl) &= \frac{1}{2}\Big(lE_3\lambda_T N_+^2 + \epsilon_0 k^{3/2}\tilde{W}\Big)\end{aligned} \tag{5.518}$$

The discretisation of M^2, $N_\pm^2$, ν_T and λ_T is discussed in Section 5.3.12.2. Production terms are taken explicit in time using values of all quantities at time t^n. Sink terms are discretised quasi-implictly using (5.476):

$$\begin{aligned}\mathcal{S}(k) &= (\lambda_T N_+^2 + \varepsilon^n)\frac{k^{n+1}}{k^n}\\ \mathcal{S}(\varepsilon) &= c_{1\varepsilon}c_{3\varepsilon}\lambda_T N_+^2\frac{\varepsilon^{n+1}}{\varepsilon^n} + c_{2\varepsilon}\frac{\varepsilon^n\varepsilon^{n+1}}{k^n}\\ \mathcal{S}(kl) &= \frac{1}{2}\Big(E_3\frac{\lambda_T N_+^2}{k^n} + \epsilon_0\frac{(kl)^{n+1}\sqrt{k^n}\tilde{W}}{l^n}\Big)\end{aligned} \tag{5.519}$$

5.6.6 Boundary conditions

5.6.6.1 surface boundary conditions

In analogy with the scalar case two Neumann and two Dirichlet type of surface boundary conditions are available in the program.

1. Neumann condition with a prescribed flux $F_{s;N_z+1}^{\psi;w}$ at the surface

$$D_{3;ij,N_z+1}^{w} = F_{s;ij,N_z+1}^{\psi;w} \tag{5.520}$$

The flux at the first C-node below the surface is then determined by interpolating the surface value and the calculated flux at the second C-node below the surface

$$\begin{aligned}D_{3;ijN_z}^{c} &= \frac{2h_{3;ijN_z}^{w}F_{s;ij,N_z+1}^{\psi;w} + h_{3;ijN_z}^{c}D_{3;ij,N_z-1}^{c}}{2h_{3;ijN_z}^{w} + h_{3;ijN_z}^{c}}\\ &= \frac{2h_{3;ijN_z}^{w}F_{s;ij,N_z+1}^{\psi;w}}{2h_{3;ijN_z}^{w} + h_{3;ijN_z}^{c}} + \frac{h_{3;ijN_z}^{c}\lambda_{T;ij,N_z-1}^{\psi;c}(\psi_{ijN_z}^{w} - \psi_{ij,N_z-1}^{w})}{h_{3;ij,N_z-1}^{c}(2h_{3;ijN_z}^{w} + h_{3;ijN_z}^{c})}\end{aligned}$$

Table 5.6: Discretisation schemes for each of the available surface and bottom boundary conditions for turbulent variables.

variable	equation	discretisation scheme
k	(4.281)	Dirichlet at the surface
ε	(4.281)	Dirichlet at the first W-node below the surface
l	(4.281)	Dirichlet at the first W-node below the surface
k	(4.283)	prescribed flux at the surface
ε	(4.284)	prescribed flux at the first C-node below the surface
k	(4.351)	Dirichlet at the bottom
ε	(4.351)	Dirichlet at the first W-node above the sea bed
l	(4.351)	Dirichlet at the first W-node above the sea bed
k	(4.352)	prescribed flux at the bottom
ε	(4.353)	prescribed flux at the first C-node above the sea bed

(5.521)

with

$$\psi^w_{ijN_z} - \psi^w_{ij,N_z-1} = \theta_v(\psi^{n+1;w}_{ijN_z} - \psi^{n+1;w}_{ij,N_z-1}) + (1-\theta_v)(\psi^{n;w}_{ijN_z} - \psi^{n;w}_{ij,N_z-1}) \tag{5.522}$$

2. Neumann using a prescribed flux $F^{\psi;c}_{s;ijN_z}$ at the first C-node below the surface

$$D^c_{3;ijN_z} = F^{\psi;c}_{s;ijN_z} \tag{5.523}$$

3. Dirichlet condition with a prescribed value $\psi^w_{s;ij,N_z+1}$ at the surface

$$\psi_{ij,N_z+1} = \psi^w_{s;ij,N_z+1} \tag{5.524}$$

4. Dirichlet condition with a prescribed value $\psi^w_{s;ijN_z}$ at the first W-node below the surface

$$\psi_{ijN_z} = \psi^w_{s;ijN_z} \tag{5.525}$$

Several formulations of surface boundary conditions have been introduced in Section 4.7.5. The discretisation scheme for each formulation is indicated in Table 5.6.

It is remarked that all turbulent diffusion coefficients are calculated using those values of k, ε and l which are located within the water column and not at the surface itself so that (5.523) and (5.525) can be considered as realistic conditions.

5.6.6.2 bottom boundary conditions

In analogy with the scalar case two Neumann and two Dirichlet type of bottom boundary conditions are available in the program.

1. Neumann condition with a prescribed flux $F^{\psi;w}_{b;ij1}$ at the bottom

$$D^{w}_{3;ij1} = F^{\psi;w}_{b;ij1} \tag{5.526}$$

The flux at the first C-node above the sea bed is then determined by interpolating the bottom value and the calculated flux at the second C-node above the sea bed

$$\begin{aligned} D^{c}_{3;ij1} &= \frac{2h^{w}_{3;ij2}F^{\psi;w}_{b;ij} + h^{c}_{3;ij1}D^{c}_{3;ij2}}{2h^{w}_{3;ij2} + h^{c}_{3;ij1}} \\ &= \frac{2h^{w}_{3;ij2}F^{\psi;w}_{b;ij}}{2h^{w}_{3;ij2} + h^{c}_{3;ij1}} + \frac{h^{c}_{3;ij1}\lambda^{\psi;c}_{T;ij2}(\psi^{w}_{ij3} - \psi^{w}_{ij2})}{h^{c}_{3;ij2}(2h^{w}_{3;ij2} + h^{c}_{3;ij1})} \end{aligned} \tag{5.527}$$

with

$$\psi^{w}_{ij3} - \psi^{w}_{ij2} = \theta_v(\psi^{n+1;w}_{ij3} - \psi^{n+1;w}_{ij2}) + (1-\theta_v)(\psi^{n;w}_{ij3} - \psi^{n;w}_{ij2}) \tag{5.528}$$

2. Neumann using a prescribed flux $F^{\psi;c}_{b;ij1}$ at the first C-node above the bottom

$$D^{c}_{3;ij1} = F^{\psi;c}_{b;ij1} \tag{5.529}$$

3. Dirichlet condition with a prescribed value $\psi^{w}_{b;ij1}$ at the bottom

$$\psi_{ij1} = \psi^{w}_{b;ij1} \tag{5.530}$$

4. Dirichlet condition with a prescribed value $\psi^{w}_{b;ij2}$ at the first W-node above the bottom

$$\psi_{ij2} = \psi^{w}_{b;ij2} \tag{5.531}$$

Several formulations of bottom boundary conditions have been introduced in Section 4.9.4. The discretisation scheme for each formulation is indicated in Table 5.6.

5.6.6.3 lateral boundary conditions

The fluxes normal to an open boundary are calculated using the upwind scheme. Applying the zero gradient condition one obtains

$$\psi^{e;w}_{ijk} = \psi^{w}_{i:i-1,jk} \quad \text{or} \quad F^{uw}_{1;ijk} = u^{uw}_{ijk}\psi^{w}_{i:i-1,jk} \tag{5.532}$$

$$\psi^{e;w}_{ijk} = \psi^{w}_{i,j:j-1,k} \quad \text{or} \quad F^{vw}_{2;ijk} = v^{vw}_{ijk}\psi^{w}_{i,j:j-1k} \tag{5.533}$$

Advective fluxes normal to a closed (coastal) open boundary are set to zero.

5.6.7 Solution of the discretised equations for turbulent transport variables

As for momentum, the discretised equations can be written in the tridiagonal form (5.315). Expressions for the matrix components are given below for the case that no operator splitting is used. They are easily extended to the case with operator splitting.

When a Neumann boundary condition is taken, no calculation is performed at the surface or bottom itself. In case of a Dirichlet condition, the surface (bottom) value of ψ is determined by the boundary condition itself. This means that the vertical index k varies between k_{min} and k_{max}. The lower limit k_{min} equals 3 for a Dirichlet condition at the first W-node above the bottom and 2 otherwise. Likewise k_{max} equals $N_z - 1$ for a Dirichlet condition at the first W-node below the surface and N_z otherwise.

For simplicity, the i and j indices are omitted.

1. Time derivative.

   The contribution of the time derivative is given by

   $$A^t_k = 0\,, \quad B^t_k = 1\,, \quad C^t_k = 0\,, \quad D^t_k = \psi^{n;w}_k \tag{5.534}$$

   where $k_{min} \le k \le k_{max}$.

2. Vertical advection.

   The vertical advection term is split up into two contributions arising from the fluxes below and above a k-level. The former are given by

   $$\begin{aligned} A^{a-}_k &= -\theta_a c^-_k (1 + f_{k-1}) \\ B^{a-}_k &= -\theta_a c^-_k (1 - f_{k-1}) \\ C^{a-}_k &= 0 \end{aligned}$$

$$D_k^{a-} = (1-\theta_a)c_k^- \Big((1+f_{k-1})\psi_{k-1}^{n;w} + (1-f_{k-1})\psi_k^{n;w}\Big) \quad (5.535)$$

where $k_{min} \le k \le k_{max}$,

$$c_k^- = \frac{\Delta t \omega_{k-1}^c}{2h_{3;k}^w}, \quad f_k = \Big(1 - \Omega(r_k^c)\Big)s_k \quad (5.536)$$

and s_{ijk}, r_{ijk}^w are defined by (5.505) and (5.506).

The terms arising from the flux above the k-level, are

$$\begin{aligned} A_k^{a+} &= 0 \\ B_k^{a+} &= \theta_a c_k^+ (1+f_k) \\ C_k^{a+} &= \theta_a c_k^+ (1-f_k) \\ D_k^{a+} &= -(1-\theta_a)c_k^+ \Big((1+f_k)\psi_k^{n;w} + (1-f_k)\psi_{k+1}^{n;w}\Big) \end{aligned} \quad (5.537)$$

where $k_{min} \le k \le k_{max}$ and

$$c_k^+ = \frac{\Delta t \omega_k^c}{2h_{3;k}^w} \quad (5.538)$$

3. Vertical diffusion.

As for vertical advection the fluxes below and above a k-level are taken separately. The former are given by

$$\begin{aligned} A_k^{d-} &= -\theta_v \frac{\Delta t \lambda_{T;k-1}^{\psi;c}}{h_{3;k-1}^c h_{3;k}^w} \\ B_k^{d-} &= \theta_v \frac{\Delta t \lambda_{T;k-1}^{\psi;c}}{h_{3;k-1}^c h_{3;k}^w} \\ C_k^{d-} &= 0 \\ D_k^{d-} &= -(1-\theta_v)\frac{\Delta t \lambda_{T;k-1}^{\psi;c}}{h_{3;k-1}^c h_{3;k}^w}(\psi_k^{n;w} - \psi_{k-1}^{n;w}) \end{aligned} \quad (5.539)$$

where $k_{lo} \le k \le k_{max}$ and k_{lo} equals 2 for a Dirichlet condition at the bottom and 3 otherwise.

The terms taken from the flux above the k-level, are

$$A_k^{d+} = 0$$

$$\begin{aligned} B_k^{d+} &= \theta_v \frac{\Delta t \lambda_{T;k}^{\psi;c}}{h_{3;k}^c h_{3;k}^w} \\ C_k^{d+} &= -\theta_v \frac{\Delta t \lambda_{T;k}^{\psi;c}}{h_{3;k}^c h_{3;k}^w} \\ D_k^{d+} &= (1-\theta_v) \frac{\Delta t \lambda_{T;k}^{\psi;c}}{h_{3;k}^c h_{3;k}^w} (\psi_{k+1}^{n;w} - \psi_k^{n;w}) \end{aligned} \tag{5.540}$$

where $k_{min} \leq k \leq k_{up}$ and k_{up} equals N_z for a Dirichlet condition at the surface and N_z-1 otherwise.

4. Sink terms.

$$A_k^S = C_k^S = D_k^S = 0\,, \quad B_k^S = \frac{\mathcal{S}(\psi^{n;w})_k}{\psi_k^{n;w}} \tag{5.541}$$

where $k_{min} \leq k \leq k_{max}$.

5. Other explicit terms.

All other terms are explicit. Their contributions can be written as

$$\begin{aligned} A_k^e &= B_k^e = C_k^e = 0 \\ D_k^e &= \Delta t \Big(\mathcal{P}_k^n - \tilde{\mathcal{A}}_{h1}(\psi^{n;w})_k^w - \tilde{\mathcal{A}}_{h2}(\psi^{n;w})_k^w + \mathcal{C}_{s1}(\psi^{n;w})_k^w \\ &\quad + \mathcal{C}_{s2}(\psi^{n;w})_k^w + \mathcal{C}_{s3}(\psi^{n;w})_k^w + \mathcal{D}_{sh1}(\psi^{n;w})_k^w + \mathcal{D}_{sh2}(\psi^{n;w})_k^w \Big) \end{aligned} \tag{5.542}$$

where $k_{min} \leq k \leq k_{max}$.

6. Surface boundary conditions.

Contributions from the surface boundary conditions depends on the type of condition as described in Section 5.6.6.1.

- Neumann condition with a prescribed surface flux $F_{s;N_z+1}^{\psi;w}$.

$$\begin{aligned} A_{N_z}^s &= \frac{\theta_v \Delta t h_{3;N_z}^c \lambda_{T;N_z-1}^{\psi;c}}{h_{3;N_z}^w (2h_{3;N_z}^w + h_{3;N_z}^c)) h_{3;N_z-1}^c} \\ B_{N_z}^s &= -\frac{\theta_v \Delta t h_{3;N_z}^c \lambda_{T;N_z-1}^{\psi;c}}{h_{3;N_z}^w (2h_{3;N_z}^w + h_{3;N_z}^c)) h_{3;N_z-1}^c} \\ C_{N_z}^s &= 0 \end{aligned}$$

$$D^s_{N_z} = \frac{\Delta t}{2h^w_{3;N_z} + h^c_{3;N_z}} \Big(2F^{\psi;w}_{s;N_z+1} + (1-\theta_v) \frac{h^c_{3;N_z} \lambda^{\psi;c}_{T;N_z-1} (\psi^{n;w}_{N_z} - \psi^{n;w}_{N_z-1})}{h^w_{3;N_z} h^c_{3;N_z-1}} \Big) \quad (5.543)$$

- Neumann using a prescribed flux $F^{\psi;c}_{s;N_z}$ at the first C-node below the surface

$$A^s_{N_z} = B^s_{N_z} = C^s_{N_z} = 0\,, \quad D^s_{N_z} = \frac{\Delta t F^{\psi;c}_{s;N_z}}{h^w_{3;N_z}} \quad (5.544)$$

- Dirichlet condition with a prescribed value $\psi^w_{s;N_z+1}$ at the surface

$$A^s_{N_z+1} = C^s_{N_z+1} = 0\,, \quad B^s_{N_z+1} = 1\,, \quad D^s_{N_z+1} = \psi^w_{s;N_z+1} \quad (5.545)$$

- Dirichlet condition with a prescribed value $\psi^w_{s;N_z}$ at the first W-node below the surface

$$A^s_{N_z} = C^s_{N_z} = 0\,, \quad B^s_{N_z} = 1\,, \quad D^s_{N_z} = \psi^w_{s;N_z} \quad (5.546)$$

7. Bottom boundary conditions.

   Contributions from the bottom boundary conditions depends on the type of condition as described in Section 5.6.6.2.

   - Neumann condition with a prescribed bottom flux $F^{\psi;w}_{b;1}$ at the bottom

$$\begin{aligned} A^b_2 &= 0 \\ B^b_2 &= -\frac{\theta_v \Delta t h^c_{3;1} \lambda^{\psi;c}_{T;2}}{h^w_{3;2}(2h^w_{3;2} + h^c_{3;1}))h^c_{3;2}} \\ C^b_2 &= \frac{\theta_v \Delta t h^c_{3;1} \lambda^{\psi;c}_{T;2}}{h^w_{3;2}(2h^w_{3;2} + h^c_{3;1}))h^c_{3;2}} \\ D^b_2 &= -\frac{\Delta t}{2h^w_{3;2} + h^c_{3;1}} \Big(2F^{\psi;w}_{b;1} + (1-\theta_v) \frac{h^c_{3;1} \lambda^{\psi;c}_{T;2} (\psi^{n;w}_3 - \psi^{n;w}_2)}{h^w_{3;2} h^c_{3;2}} \Big) \end{aligned} \quad (5.547)$$

   - Neumann using a prescribed flux $F^{\psi;c}_{b;1}$ at the first C-node above the bottom

$$A^b_2 = B^b_2 = C^s_2 = 0\,, \quad D^b_2 = -\frac{\Delta t F^{\psi;c}_{b;1}}{h^w_{3;2}} \quad (5.548)$$

- Dirichlet condition with a prescribed value $\psi^w_{b;1}$ at the bottom

$$A^b_1 = C^b_1 = 0\,, \quad B^b_1 = 1\,, \quad D^b_1 = \psi^w_{b;1} \tag{5.549}$$

- Dirichlet condition with a prescribed value $\psi^w_{b;2}$ at the first W-node above the bottom

$$A^b_2 = C^b_2 = 0\,, \quad B^b_2 = 1\,, \quad D^b_2 = \psi^w_{b;2} \tag{5.550}$$

5.7 Discretisations on reduced grids

5.7.1 Discretised 1-D mode equations

1. To make the code compatible with the 3-D case which uses an Arakawa C-grid, the model grid on which the equations are discretised, does not not reduce to a single point but consists of 3 rows and columns (i.e. nc=nr=3) of which the last column and the last row consist of dummy land points (see Figure 5.1). This produces a computational overhead since the same calculation is performed at each of the four wet C-nodes and the two internal U and V velocity nodes.

2. Momentum equations

   - The 1-D versions (4.109)–4.110) are integrated in time without operator and mode splitting using the formulations given in Sections 5.3.1.1 and 5.3.1.3. Firstly, "predicted" values are calculated

$$\frac{\tilde{u}^p - u^n}{\Delta t} = fv^n + \theta_v \mathcal{D}_{mv}(\tilde{u}^p) + (1-\theta_v)\mathcal{D}_{mv}(u^n) - g\frac{\partial \zeta^{n+1}}{\partial x} + F_1^{t;n+1} \tag{5.551}$$

$$\frac{\tilde{v}^p - v^n}{\Delta t} = -fu^n + \theta_v \mathcal{D}_{mv}(\tilde{v}^p) + (1-\theta_v)\mathcal{D}_{mv}(v^n) - g\frac{\partial \zeta^{n+1}}{\partial y} + F_2^{t;n+1} \tag{5.552}$$

   An implicit correction is added for the Coriolis force giving (u^p, v^p) by equations (5.10). "Corrected" values are obtained by

$$u^{n+1} = \frac{h^n_{3;k}}{h^{n+1}_{3;k}} u^p\,, \quad v^{n+1} = \frac{h^n_{3;k}}{h^{n+1}_{3;k}} v^p \tag{5.553}$$

   where the surface slopes and the elevations used to calculate the vertical grid spacing $h^{n+1}_{3;k} = (h + \zeta^{n+1})\Delta\sigma_k$ are prescribed externally by expressions of the form (4.285).

- The vertical diffusion terms and coefficients are discretised using the formulations given in Sections 5.3.11 and 5.3.12.2.
- Once the currents are updated, the depth-mean currents $\overline{u}$ and $\overline{v}$ are evaluated (for user output only).

3. Scalar equations

- The transport equation for a scalar ψ is integrated in time without operator splitting using the discretisation (5.393).
- The vertical diffusion term and coefficient are discretised as described in Sections 5.5.5.3 and 5.5.6.2.
- Neumann and Dirichlet onditions can be applied as discussed in Sections 5.5.7.1–5.5.7.2.

4. The turbulence equations are solved as in the 3-D case without advection and operator splitting.

5.7.2 Discretised depth-integrated equations

1. Momentum equations

- The surface elevation and depth-integrated currents are updated by solving the 2-D momentum equations using the same discretisation procedures given in Sections 5.3 without the depth-integrated baroclinic terms but with the same barotropic time step $\Delta\tau$.
- To make the code compatible with the 3-D case all "3-D" currents are set to their depth-mean value

$$\begin{aligned} u_f &= \overline{u} = U/H = u \\ v_f &= \overline{v} = V/H = v \end{aligned} \tag{5.554}$$

2. Scalar equations

- Scalar transport is discretised in exactly the same way as in the 3-D case with the larger 3-D time step Δt.
- The vertical diffusion term is retained and discretised using (5.436) except that the upper and lower fluxes are located at the respectively the surface and the bottom and therefore obtained by the surface and bottom boundary conditions which must obviously be of the Neumann type.

3. No turbulence transport equations need to be solved.

5.8 Solution procedure

The general solution procedure can be summarised as follows

1. Initial time $t = 0$.
   - 1.1 Obtain initial conditions for U, V, ζ, u, v, ω, T, S, k, l or ε, and quantities which are updated in time at open boundaries.
   - 1.2 Initialise ρ, β_T, β_S from the equation of state.
   - 1.3 Evaluate the astronomical force at the initial time.
   - 1.4 Initialise meteorological data.
   - 1.5 Initialise open boundary data for the 2-D mode, 3-D mode, temperature, salinity.
   - 1.6 Initialise surface and bottom stress.
2. Predictor step at $t = t^p = t^n + \Delta\tau$ with n=0, ..., $N_{tot} - 1$.
   - 2.1 Update meteorological data (if needed).
   - 2.2 Update ρ, β_T, β_S from the equation of state.
   - 2.3 Update all vertical diffusion coefficients. In case of a RANS model, k, l or ε are first updated at time t^{n+1}.
   - 2.4 Evaluate the components of the baroclinic pressure gradient.
   - 2.5 Evaluate the horizontal diffusion coefficients at different nodes.
   - 2.6 Obtain u^p, v^p by solving the 3-D momentum equations.
3. Barotropic time steps $t = t^n + m\Delta\tau$ with m=1, ..., M_t.
   - 3.1 Update meteorological data (if needed).
   - 3.2 Solve 2-D continuity equation for ζ.
   - 3.3 Update open boundary data for the 2-D mode (if needed).
   - 3.4 Update astronomical force.
   - 3.5 Update U, V by solving the 2-D momentum equations.
   - 3.6 Update the time-averaged transports U_f, V_f.
4. Corrector step at $t = t^{n+1} = t^n + \Delta t$ with n=1, ..., N_{tot}.
   - 4.1 3-D mode
     - 4.1.1 Update open boundary data (if needed).

4.1.2 Apply open boundary conditions.
4.1.3 Apply filter correction to obtain u^{n+1}, v^{n+1}.
4.1.4 Evaluate filtered currents u_f, v_f.
4.1.5 Solve 3-D baroclinic continuity equation for ω.
4.1.6 Update physical vertical current.
4.1.7 Update bottom and surface stress (if needed).
4.2 Update temperature at time t^{n+1}.
4.2.1 Update open boundary data (if needed).
4.2.2 Apply open boundary conditions.
4.2.3 Evaluate solar irradiance.
4.2.4 Evaluate surface (non-solar) heat fluxes.
4.2.5 Solve temperature equation.
4.3 Update salinity at time t^{n+1}.
4.3.1 Update open boundary data (if needed).
4.3.2 Apply open boundary conditions.
4.3.3 Solve salinity equation.

Note that

- Some of the previous steps are only conditionally performed, depending on the setting of model switches. For example, the temperature equation is only updated when iopt_temp=2, the astronomical tidal force is only included if iopt_astro_tide=1, ...
- Update of surface or open boundary forcing data depends on the settings of the tlims attribute, discussed in Section 14.7.2 of the User Manual.