

Chapter 4

4	Numerical Model Development	4.1
	Abstract	4.1
	Résumé	4.1
4.1	Governing equations	4.2
4.2	Solution strategy: the iterative method	4.5
4.3	Numerical method: the finite-volume approximation	4.8
4.3.1	Grid arrangement	4.8
4.3.2	Computation of the surface area and of the cell volume	4.9
4.3.3	Cell-face interpolation and gradient computation	4.11
4.3.4	Discretisation of the time derivative terms	4.12
4.3.5	Discretisation of the convective terms	4.13
4.3.6	Discretisation of the diffusive terms	4.14
4.3.7	Convective-diffusive terms: hybrid and power-law schemes	4.16
4.3.8	Source terms	4.19
4.3.9	Assembly of the coefficients.....	4.22
4.4	Pressure-velocity coupling	4.24
4.4.1	SIMPLE algorithm.....	4.24
4.4.2	Pressure correction procedure.....	4.28
4.4.3	Under-relaxation factor and time step	4.29
4.5	Boundary conditions	4.29
4.5.1	Boundary placement	4.29
4.5.2	Inflow boundary	4.30
4.5.3	Outflow boundary	4.32
4.5.4	Wall boundary	4.33
4.5.5	Symmetry boundary	4.41
4.5.6	Surface boundary	4.43
4.6	Solution procedures	4.47
4.6.1	Spatial discretisation	4.47
4.6.2	Matrix solvers	4.49
4.7	Summary	4.53
	References	4.53
	Notations	4.54

4 Numerical Model Development

Abstract

Presented in this chapter is the development of a 3D numerical model intended to simulate flow around a cylinder. The model is based on the Reynolds-averaged Navier-Stokes and continuity equations for incompressible flow, closed with the k - ϵ turbulence model. The working equation of the model is obtained by discretizing the governing equations, written in a general convective-diffusive transport equation, using finite-volume techniques on a structured, collocated, boundary-fitted, hexahedral control-volume grid. The hybrid (*Spalding, 1972*) or power-law (*Patankar, 1980*) upwind-central difference scheme, combined with the deferred correction method (*Ferziger and Peric, 1997*), is employed in the discretisation of the governing equations. The solution of the working equation is achieved by an iterative method according to SIMPLE algorithm (*Patankar and Spalding, 1972*). Along solid boundaries, use is made of the wall function method, while along surface boundaries the pressure defect is used to define the surface position. On other boundaries, namely inlet, outlet, and symmetry boundaries, classical methods are used, such as zero gradients, zero stresses, or known functions.

Résumé

Ce chapitre présente un développement d'un modèle numérique pour simuler l'écoulement tridimensionnel autour d'un cylindre. Le modèle est basée sur la représentation en volumes finis des équations de Reynolds, de continuité et de k - ϵ . Les équations, sous forme d'une équation de transport, sont en suite formulées pour un maillage structuré dont les variables primitives sont définies au centre des volumes de contrôle. Les flux convectif et diffusif sont calculés par les méthodes *hybride* (*Spalding, 1972*) ou *loi de puissance* (*Patankar, 1980*) avec des corrections des termes non orthogonaux (*Ferziger and Peric, 1997*). Le modèle utilise la méthode itérative de SIMPLE pour résoudre les équations de travail ainsi obtenues. Les conditions aux bords le long d'une parois sont imposées par la loi logarithmique. La surface d'eau est déterminée à partir des pressions résiduelles dans les cellules de surface. En autres types des bords, par exemple à l'entrée, à la sortie et aux plans de symétrie, des méthodes standards sont appliquées soit des gradients nuls, sans cisaillement ou des valeurs connues.

4.1 Governing equations

The flow model that is developed in this work is based on the approximate solution of the time-averaged equations of motion and continuity for incompressible flows by using finite-volume method. In the Cartesian coordinate system these equations read:

$$\frac{\partial u}{\partial t} + \frac{\partial uu}{\partial x} + \frac{\partial vu}{\partial y} + \frac{\partial wu}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{xx}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yx}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zx}}{\partial z} + g_x \quad (4.1)$$

$$\frac{\partial v}{\partial t} + \frac{\partial uv}{\partial x} + \frac{\partial vv}{\partial y} + \frac{\partial wv}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{xy}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yy}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zy}}{\partial z} + g_y \quad (4.2)$$

$$\frac{\partial w}{\partial t} + \frac{\partial uw}{\partial x} + \frac{\partial vw}{\partial y} + \frac{\partial ww}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial z} + \frac{1}{\rho} \frac{\partial \tau_{xz}}{\partial x} + \frac{1}{\rho} \frac{\partial \tau_{yz}}{\partial y} + \frac{1}{\rho} \frac{\partial \tau_{zz}}{\partial z} + g_z \quad (4.3)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (4.4)$$

in which x , y , and z are Cartesian co-ordinates in the horizontal, transversal, and vertical, respectively; u , v , and w are the corresponding (time-averaged) velocity components, p is the (time-averaged) pressure, ρ is the mass density of water, g_x , g_y , g_z are the x , y , z components of the gravitational acceleration, and τ_{ij} 's are the j direction components of the shear stress acting on the surface normal to the i direction. These stresses are due to the molecular viscosity and turbulent fluctuation. For flows having sufficiently high Reynolds number, the viscous stresses are much smaller in comparison with those of the turbulence and thus can be neglected. Using Boussinesq's eddy viscosity concept, these stresses are proportional to the velocity gradients according to the following expressions (see *Rodi*, 1984, p. 10):

$$\begin{aligned} \frac{\tau_{xx}}{\rho} &= \nu_t 2 \frac{\partial u}{\partial x} - \frac{2}{3} k, & \frac{\tau_{xy}}{\rho} &= \frac{\tau_{yx}}{\rho} = \nu_t \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right), \\ \frac{\tau_{yy}}{\rho} &= \nu_t 2 \frac{\partial v}{\partial y} - \frac{2}{3} k, & \frac{\tau_{xz}}{\rho} &= \frac{\tau_{zx}}{\rho} = \nu_t \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right), \\ \frac{\tau_{zz}}{\rho} &= \nu_t 2 \frac{\partial w}{\partial z} - \frac{2}{3} k, & \frac{\tau_{yz}}{\rho} &= \frac{\tau_{zy}}{\rho} = \nu_t \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right). \end{aligned} \quad (4.5)$$

in which ν_t is the turbulent or eddy viscosity and k is the turbulent kinetic energy defined as $k = \frac{1}{2} (\overline{u'u'} + \overline{v'v'} + \overline{w'w'})$ where superscripts mean the fluctuating components. Inserting the definitions in Eq. 4.5 into the momentum equations, Eqs. 4.1 to 4.3, one obtains:

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial uu}{\partial x} + \frac{\partial vu}{\partial y} + \frac{\partial wu}{\partial z} &= -\frac{1}{\rho} \frac{\partial p}{\partial x} - \frac{2}{3} \frac{\partial k}{\partial x} \\ &+ \frac{\partial}{\partial x} \nu_t 2 \left(\frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \nu_t \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \nu_t \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) + g_x \end{aligned}$$

$$\begin{aligned} \frac{\partial v}{\partial t} + \frac{\partial uv}{\partial x} + \frac{\partial vv}{\partial y} + \frac{\partial wv}{\partial z} &= -\frac{1}{\rho} \frac{\partial p}{\partial y} - \frac{2}{3} \frac{\partial k}{\partial y} \\ &+ \frac{\partial}{\partial x} \nu_t \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \nu_t 2 \left(\frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial z} \nu_t \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) + g_y \end{aligned}$$

$$\begin{aligned} \frac{\partial w}{\partial t} + \frac{\partial uw}{\partial x} + \frac{\partial vw}{\partial y} + \frac{\partial ww}{\partial z} &= -\frac{1}{\rho} \frac{\partial p}{\partial z} - \frac{2}{3} \frac{\partial k}{\partial z} \\ &+ \frac{\partial}{\partial x} \nu_t \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) + \frac{\partial}{\partial y} \nu_t \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) + \frac{\partial}{\partial z} \nu_t 2 \left(\frac{\partial w}{\partial z} \right) + g_z \end{aligned}$$

Separating the normal and cross second-derivatives and putting the former on the left-hand sides, one gets:

$$\begin{aligned} \frac{\partial u}{\partial t} + \frac{\partial uu}{\partial x} + \frac{\partial vu}{\partial y} + \frac{\partial wu}{\partial z} - \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial x} \right) - \frac{\partial}{\partial y} \left(\nu_t \frac{\partial u}{\partial y} \right) - \frac{\partial}{\partial z} \left(\nu_t \frac{\partial u}{\partial z} \right) &= -\frac{1}{\rho} \frac{\partial p}{\partial x} \\ &+ \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial x} \right) + g_x \end{aligned} \quad (4.6)$$

$$\begin{aligned} \frac{\partial v}{\partial t} + \frac{\partial uv}{\partial x} + \frac{\partial vv}{\partial y} + \frac{\partial wv}{\partial z} - \frac{\partial}{\partial x} \left(\nu_t \frac{\partial v}{\partial x} \right) - \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial y} \right) - \frac{\partial}{\partial z} \left(\nu_t \frac{\partial v}{\partial z} \right) &= -\frac{1}{\rho} \frac{\partial p}{\partial y} \\ &+ \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial y} \right) + g_y \end{aligned} \quad (4.7)$$

$$\begin{aligned} \frac{\partial w}{\partial t} + \frac{\partial uw}{\partial x} + \frac{\partial vw}{\partial y} + \frac{\partial ww}{\partial z} - \frac{\partial}{\partial x} \left(\nu_t \frac{\partial w}{\partial x} \right) - \frac{\partial}{\partial y} \left(\nu_t \frac{\partial w}{\partial y} \right) - \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial z} \right) &= -\frac{1}{\rho} \frac{\partial p}{\partial z} \\ &+ \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial z} \right) + \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial z} \right) + \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial z} \right) + g_z \end{aligned} \quad (4.8)$$

The second to fourth terms on the left-hand side of Eqs. 4.6 to 4.8 represent a convective transport and the next three terms represent a diffusive transport. The terms on the right-hand side are considered as sources and are treated as known quantities when solving the equations for the velocity components u , v , and w . The turbulent kinetic energy gradient, being small compared to the pressure gradient, is neglected.

From the k - ϵ turbulence model (*Launder and Spalding, 1974; Rodi, 1984, p. 27*), the turbulent viscosity, ν_t , is given by:

$$\nu_t = c_\mu \frac{k^2}{\varepsilon} \quad (4.9)$$

where ε is the dissipation of the turbulent kinetic energy. The field distributions of the turbulent kinetic energy and its dissipation are obtained from the following transport equations (*Launder and Spalding, 1974; Rodi, 1984, p. 28*):

$$\frac{\partial k}{\partial t} + \frac{\partial uk}{\partial x} + \frac{\partial vk}{\partial y} + \frac{\partial wk}{\partial z} - \frac{\partial}{\partial x} \left(\frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial x} \right) - \frac{\partial}{\partial y} \left(\frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial y} \right) - \frac{\partial}{\partial z} \left(\frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial z} \right) = G - \varepsilon \quad (4.10)$$

$$\frac{\partial \varepsilon}{\partial t} + \frac{\partial u\varepsilon}{\partial x} + \frac{\partial v\varepsilon}{\partial y} + \frac{\partial w\varepsilon}{\partial z} - \frac{\partial}{\partial x} \left(\frac{\nu_t}{\sigma_\varepsilon} \frac{\partial \varepsilon}{\partial x} \right) - \frac{\partial}{\partial y} \left(\frac{\nu_t}{\sigma_\varepsilon} \frac{\partial \varepsilon}{\partial y} \right) - \frac{\partial}{\partial z} \left(\frac{\nu_t}{\sigma_\varepsilon} \frac{\partial \varepsilon}{\partial z} \right) = \frac{\varepsilon}{k} (c_1 G - c_2 \varepsilon) \quad (4.11)$$

in which G is the production of kinetic-energy given by:

$$G = \nu_t \left\{ 2 \left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \frac{\partial u}{\partial y} + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \frac{\partial u}{\partial z} + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right) \frac{\partial v}{\partial x} + 2 \left(\frac{\partial v}{\partial y} \right)^2 + \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \frac{\partial v}{\partial z} + \left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} \right) \frac{\partial w}{\partial x} + \left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \right) \frac{\partial w}{\partial y} + 2 \left(\frac{\partial w}{\partial z} \right)^2 \right\} \quad (4.12)$$

The model coefficients c_μ , c_1 , c_2 , σ_k , and σ_ε contained in the above transport equations are assumed to be constant and take the values given in Table 4.1 (*Launder and Spalding, 1974; Rodi, 1984, p. 29*).

Table 4.1 Values of coefficients in k- ε model.

c_μ	c_1	c_2	σ_k	σ_ε
0.09	1.44	1.92	1.0	1.3

It is more convenient to cast the continuity equation, Eq. 4.4, the momentum equations, Eqs. 4.6 to 4.8, and the transport equations of k and ε , Eqs. 4.10 and 4.11, into a general transport equation (*Versteeg and Malalasekera, 1995, p. 25*):

$$\frac{\partial \phi_\ell}{\partial t} + \nabla \cdot (\phi_\ell \vec{V}) - \nabla \cdot (\Gamma_\ell \vec{\nabla} \phi_\ell) = R_\ell \quad (4.13)$$

In the above equations, ϕ_ℓ is any dependent scalar variable, \vec{V} is the velocity vector, Γ_ℓ is the diffusion coefficient, and R_ℓ is a column matrix of scalar sources (see its definition in Table 4.2). Integrating this equation over a three-dimensional discrete control volume yields (*Versteeg and Malalasekera, 1995, p. 25*):

$$\frac{\partial}{\partial t} \iiint_{\mathcal{V}} \phi_\ell \, d\mathcal{V} + \iiint_{\mathcal{V}} \nabla \cdot (\phi_\ell \vec{V}) \, d\mathcal{V} - \iiint_{\mathcal{V}} \nabla \cdot (\Gamma_\ell \vec{\nabla} \phi_\ell) \, d\mathcal{V} = \iiint_{\mathcal{V}} R_\ell \, d\mathcal{V} \quad (4.14)$$

The volume integrals of the convective and diffusive terms, the second and third terms on the left-hand side, can be expressed as integral over the closed surface bounding the control volume by applying Gauss divergence theorem (*Versteeg and Malalasekera, 1995, p. 26; Hirsch, 1988, p. 241*):

$$\frac{\partial}{\partial t} \iiint_{\mathcal{V}} \phi_{\ell} d\mathcal{V} + \iint_{\mathcal{S}} \phi_{\ell} \vec{\mathbf{V}} \cdot d\vec{\mathbf{S}} - \iint_{\mathcal{S}} \Gamma_{\ell} \vec{\mathbf{V}} \phi_{\ell} \cdot d\vec{\mathbf{S}} = \iiint_{\mathcal{V}} R_{\ell} d\mathcal{V} \quad (4.15)$$

where $\vec{\mathbf{S}}$ is the surface vector normal outward to the control volume $d\mathcal{V}$.

Table 4.2 Terms in the general transport equation, Eq. 4.13.

$\frac{\partial \phi_{\ell}}{\partial t} + \nabla(\phi_{\ell} \vec{\mathbf{V}}) - \nabla(\Gamma_{\ell} \vec{\mathbf{V}} \phi_{\ell}) = R_{\ell} \quad (4.13)$			
ℓ	ϕ_{ℓ}	Γ_{ℓ}	R_{ℓ}
1	u	ν_t	$-\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial x} \right) + g_x$
2	v	ν_t	$-\frac{1}{\rho} \frac{\partial p}{\partial y} + \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial y} \right) + \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial y} \right) + g_y$
3	w	ν_t	$-\frac{1}{\rho} \frac{\partial p}{\partial z} + \frac{\partial}{\partial x} \left(\nu_t \frac{\partial u}{\partial z} \right) + \frac{\partial}{\partial y} \left(\nu_t \frac{\partial v}{\partial z} \right) + \frac{\partial}{\partial z} \left(\nu_t \frac{\partial w}{\partial z} \right) + g_z$
4	1	0	0
5	k	ν_t / σ_k	G- ϵ
6	ϵ	$\nu_t / \sigma_{\epsilon}$	$c_1 \frac{\epsilon}{k} G - c_2 \frac{\epsilon}{k} \epsilon$

4.2 Solution strategy: the iterative method

The integral form of the general transport equation, Eq. 4.15, is used to obtain the solution for u, v, w, k, and ϵ by substituting these variables to the scalar variable ϕ . The solution of the equation is sought at discrete time steps; calculations are performed at every discrete time steps and repeated until a steady-state solution is obtained. The time derivative term in Eq. 4.15 facilitates the application of the model to transient flow problems; in this case, the solution at each discrete time step must converge. When the problems concern steady case ones —as is the case in the present work— the time derivative serves as an iteration loop. In this case the solution at each time step is considered as an intermediate solution, and the end-solution (the steady-state one) is obtained when all variables ϕ 's converge. Since the end-solution that is sought, it is not necessary to force the intermediate solution to converge at the same degree of convergence as that of the end-solution. The complete computational procedure is depicted in the flowchart shown in Fig. 4.1. The time loop, from the initial until the steady-state solution is depicted as the n-iteration. Every variable in the governing

equations, Eq. 4.15, is linked to each other since they appear in every equation. To obtain the solution of every variable that satisfies all equations at a time step, an iterative procedure is employed; this is called the m -iteration in Fig. 4.1. The basic idea of the procedure is to consecutively solve the equations for each variable in the order of the momentum, the continuity and the k - ϵ equations. The momentum equation is solved successively for the u , v , and w components. The order of the computation is not important. The solution of each variable is sought independently, for example when solving the x -momentum equation for u , the other variables appearing in that equation, the v , w , k and ϵ , are assumed as known. When all velocity components are obtained, the pressure is computed through the continuity equation, which in turn will modify the velocity. New solutions of the momentum equation are then necessary. When the velocity and pressure converge, the k - ϵ equations are solved based on the latest values of the velocity components. The k is solved first and the ϵ follows. When solving for ϵ , the latest value of k is used. The procedure is repeated until every variable satisfies all governing equations.

Eq. 4.15, however, cannot be used to obtain directly the pressure. The pressure appears in the momentum equations, but does not have any equation of its own. The fourth equation, the continuity, does not explicitly link the pressure to the velocity. The solution of the pressure is thus not straightforward; some kind of a ‘trial-and-correction’ procedure is employed. This is indicated as the ℓ -iteration. Firstly, the pressure is estimated and supplied to the momentum equation to get the u , v , and w velocity components. Secondly, the continuity equation is imposed upon those velocities. If the velocities do not satisfy the continuity equation, the velocities and the pressure are then corrected. The corrected pressure is used as the new estimate and the procedure is repeated.

Upon the completion of the ℓ -iteration, the computation continues to the k - ϵ model. Given the velocity obtained from the ℓ -iteration, the k - ϵ equations are solved consecutively, and the eddy-viscosity is subsequently obtained. A check is carried out to all new variables ϕ 's. If each ϕ satisfies all the governing equations, those variables are regarded as the values at the new time step, otherwise the computation goes back to the solution of the momentum equation (the m -iteration).

The surface boundary which determines the computational domain but its position is part of the solution, is handled at the n -iteration and kept constant during the m -iterations. Thus the positioning of the free surface is carried out explicitly. At the end of the m -iteration, the water surface is moved according to the pressure defect at the surface. This in turn will change the computational domain for the new time of the n -iteration.

The overall procedure thus involves three blocks of iteration. The first iteration block is to get solution of the velocity and pressure, i.e. solving the momentum equations and imposing the continuity. The second block solves the momentum, continuity, and k - ϵ equations within a time step. The last block is the time marching iteration to get the steady-state solution.

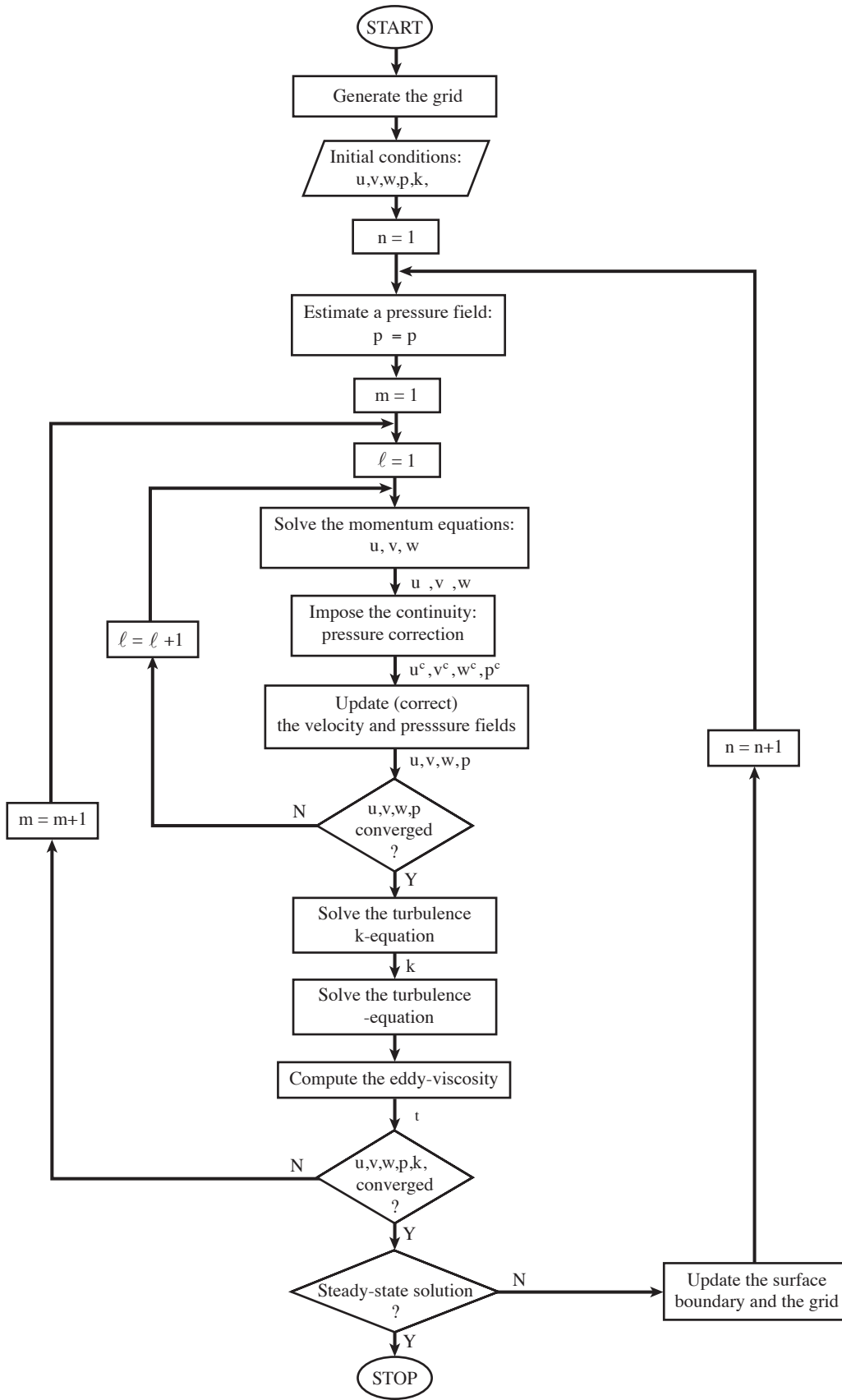


Fig. 4.1 Overall iterative procedure of the solution of Eq. 4.15.

4.3 Numerical method: the finite-volume approximation

4.3.1 Grid arrangement

The approximation to the solution of Eq. 4.15 is sought by finite-volume approach. The computational domain is discretized in a 3D grid having a finite number of control volumes (cells); the integration is then carried out in each cell. A non-orthogonal hexahedron cell is selected in the present model. A typical one is shown in Fig. 4.2. A *cell* is identified by its center, \mathbf{P} , which makes up the *node* where the dependent variable is to be defined. A cell has six neighbors, named according to their respective compass directions, being the **E**ast, **W**est, **N**orth, **E**ast, **T**op, and **B**ottom. The cell faces are identified at the face center and named with lower-case letters, namely the **e**, **w**, **n**, **s**, **t**, and **b**. The Cartesian coordinate system is selected for describing both the geometrical and flow properties, being the z axis defines the bottom-to-top direction.

It is to be noted, however, that the grid in this model is selected such that the cell faces **e**, **w**, **n**, and **s** are parallel to the z axis. This choice is taken to facilitate the handling of the surface boundary. The discretisation of the governing equation, nevertheless, is carried out for general non-orthogonal cells.

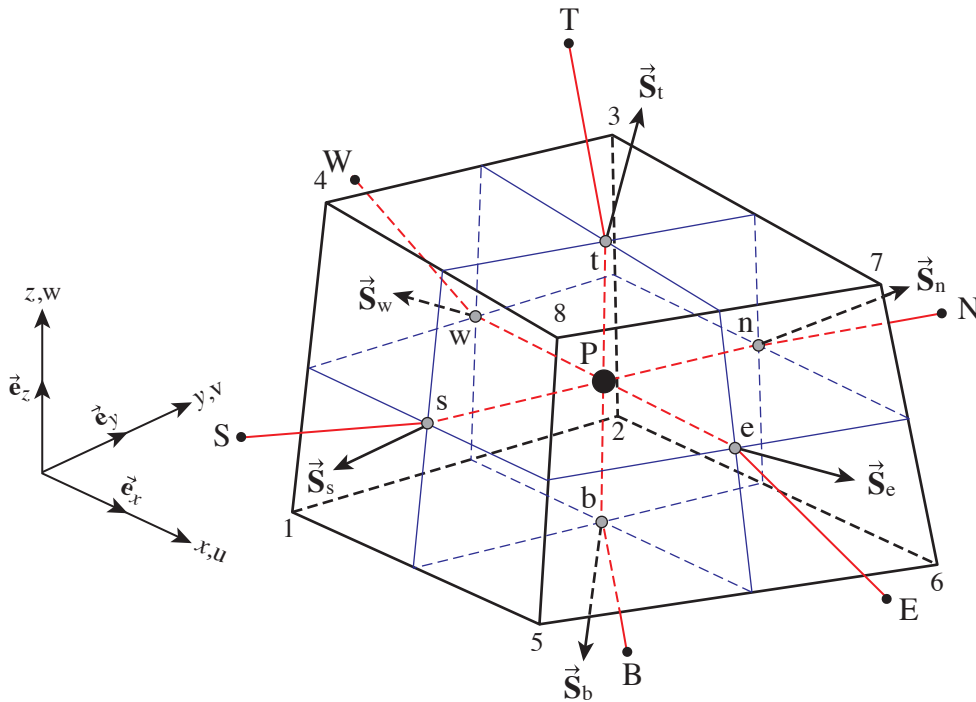


Fig. 4.2 Typical hexahedron control volume.

The dependent variables, $\phi(u, v, w, p, k, \epsilon)$, are defined at the cell center \mathbf{P} , thus constituting a *cell-centered non-staggered grid*. Non-staggered grid variable arrangements may yield a problem of pressure-velocity decoupling that creates a spurious oscillation in the solution. This problem does not exist with the use of staggered grid. However, staggered grids require separate control volumes for the velocity and other dependent variables that

increase the computer storage requirement. For flows with three-dimensional geometry, the storage space required for additional control volumes is enormous. In addition, the non-orthogonality of the cells gives another complexity since the velocity components are not related to the alignment of the cell face. This makes the non-staggered grid is more suitable for 3D problems. To avoid the problem of pressure-velocity decoupling, use is made of the interpolation method according to Rhie and Chow (*Rhie and Chow, 1983*). This method consists of determining the convective velocities on a non-staggered grid through the use of the discretized momentum equation, thus coupling the pressure field with the velocity field. The standard method for staggered grids, the SIMPLE (*Patankar and Spalding, 1972*), is then used to correct the pressure. The SIMPLE, an acronym for Semi-Implicit Method for Pressure-Linked, has been successfully employed for flow computations in two-dimensional problems (*Kobayashi and Pereira, 1991; Obi et al., 1989; Ferziger and Peric, 1997*) as well as three-dimensional cases (*Olsen and Kjellesvig, 1998; Wu et al., 2000*). The present model adopts a similar method.

4.3.2 Computation of the surface area and of the cell volume

Cell-face area

The surface vector of the cell faces can be evaluated from the vector products of the diagonals. As can be seen in Fig. 4.3, the area of the east face, quadrilateral 5678, is half of that of parallelogram ABCD built on the diagonals 57 and 68 (note the use of the clockwise convention, seen from the cell center, to index the corners). Hence the surface vector is (*Hirsch, 1988, p. 247*):

$$\vec{S}_{5678} = \frac{1}{2} \vec{S}_{ABCD} = \frac{1}{2} \begin{bmatrix} \bar{e}_x & \bar{e}_y & \bar{e}_z \\ \Delta x_{57} & \Delta y_{57} & \Delta z_{57} \\ \Delta x_{68} & \Delta y_{68} & \Delta z_{68} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \bar{e}_x & \bar{e}_y & \bar{e}_z \\ (x_7 - x_5) & (y_7 - y_5) & (z_7 - z_5) \\ (x_8 - x_6) & (y_8 - y_6) & (z_8 - z_6) \end{bmatrix} \quad (4.16)$$

When the cell face is not coplanar, the above expression gives the projection area of two triangles sharing a common side 57 or 68. The unit vector normal to a cell face is computed as follows:

$$\bar{e}_n = \vec{S} / \|\vec{S}\| \quad (4.17)$$

The normal distance from point P to the east face can then be defined as $\delta_n = \vec{L}_{Pe} \cdot (\bar{e}_n)_e$, where \vec{L}_{Pe} is the vector originating from P to face center e (note that $\vec{L}_{eP} = -\vec{L}_{Pe}$).

Cell volume

The cell volume is obtained by dividing the hexahedron into six tetrahedrons sharing one common diagonal 17 and one crest 1. Hence with $\vec{\mathbf{L}}_{17} = \vec{\mathbf{L}}_7 - \vec{\mathbf{L}}_1$, where $\vec{\mathbf{L}}_1$ and $\vec{\mathbf{L}}_7$ are the position vectors of 1 and 7, the cell volume is thus:

$$\begin{aligned} \mathbf{V}_{12345678} &= \mathbf{V}_{1857} + \mathbf{V}_{1567} + \mathbf{V}_{1627} + \mathbf{V}_{1237} + \mathbf{V}_{1347} + \mathbf{V}_{1487} \\ &= \frac{1}{6} \left\{ \vec{\mathbf{L}}_{17} \cdot \left[\left(\vec{\mathbf{L}}_{18} \times \vec{\mathbf{L}}_{15} \right) + \left(\vec{\mathbf{L}}_{15} \times \vec{\mathbf{L}}_{16} \right) + \left(\vec{\mathbf{L}}_{16} \times \vec{\mathbf{L}}_{12} \right) + \right. \right. \\ &\quad \left. \left. \left(\vec{\mathbf{L}}_{12} \times \vec{\mathbf{L}}_{13} \right) + \left(\vec{\mathbf{L}}_{13} \times \vec{\mathbf{L}}_{14} \right) + \left(\vec{\mathbf{L}}_{14} \times \vec{\mathbf{L}}_{18} \right) \right] \right\} \end{aligned} \quad (4.18)$$

in which $\vec{\mathbf{L}}_{mn} = \Delta x_{mn} \vec{\mathbf{e}}_x + \Delta y_{mn} \vec{\mathbf{e}}_y + \Delta z_{mn} \vec{\mathbf{e}}_z = (x_n - x_m) \vec{\mathbf{e}}_x + (y_n - y_m) \vec{\mathbf{e}}_y + (z_n - z_m) \vec{\mathbf{e}}_z$

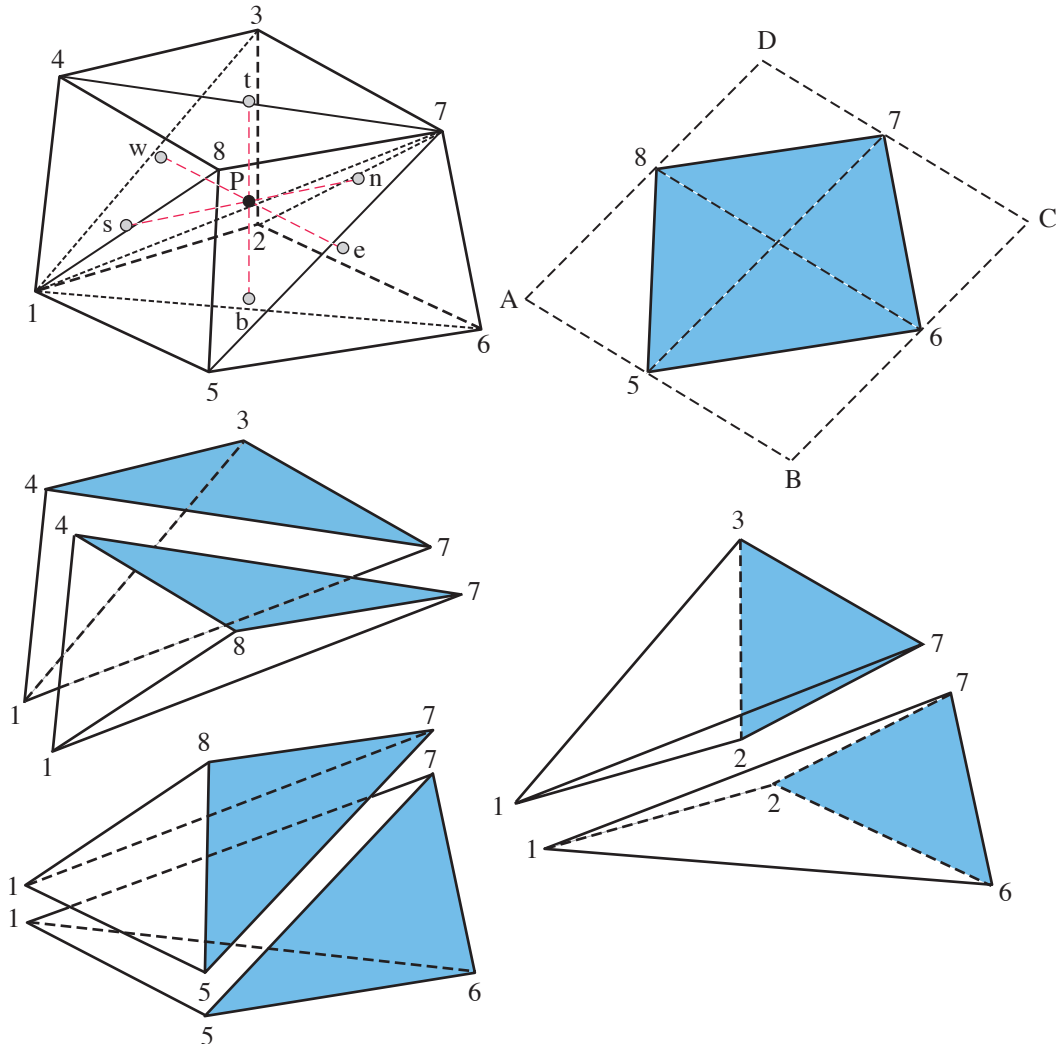


Fig. 4.3 Evaluation of the surface vector and cell volume.

4.3.3 Cell-face interpolation and gradient computation

Cell-face interpolation

Non-staggered grids define all computed variables at the cell centers. When values at the cell face are required, linear interpolation applies. Writing for the east face, the linear interpolation takes the following form:

$$\overline{(\phi)}_e = (1 - \beta_e) \phi_P + \beta_e \phi_E \quad (4.19)$$

with the interpolation factor β_e defined as: $\beta_e = L_{Pe} / L_{PE}$.

This expression is extensively used in the model, except in two cases, namely (a) when evaluating convective terms by using upwind differences (Sect. 4.3.5), and (b) when computing interpolated coefficients and starred velocities in the pressure-correction equation (Sect. 4.4).

Gradients

The Gauss theorem provides the gradient at the cell center. The gradients along the x , y , and z directions read (*Hirsch, 1988, p. 253*):

$$\begin{aligned} \left(\frac{\partial \phi}{\partial x} \right)_P &\approx \frac{1}{\mathcal{V}} \iiint_{\mathcal{V}} \vec{\nabla}(\phi \vec{e}_x) \cdot d\mathcal{V} = \frac{1}{\mathcal{V}} \iint_S \phi \vec{e}_x \cdot d\vec{S} \approx \frac{1}{\mathcal{V}_P} \sum_{cf=ewnstb} (\phi S_x)_{cf} \\ \left(\frac{\partial \phi}{\partial y} \right)_P &\approx \frac{1}{\mathcal{V}} \iiint_{\mathcal{V}} \vec{\nabla}(\phi \vec{e}_y) \cdot d\mathcal{V} = \frac{1}{\mathcal{V}} \iint_S \phi \vec{e}_y \cdot d\vec{S} \approx \frac{1}{\mathcal{V}_P} \sum_{cf=ewnstb} (\phi S_y)_{cf} \\ \left(\frac{\partial \phi}{\partial z} \right)_P &\approx \frac{1}{\mathcal{V}} \iiint_{\mathcal{V}} \vec{\nabla}(\phi \vec{e}_z) \cdot d\mathcal{V} = \frac{1}{\mathcal{V}} \iint_S \phi \vec{e}_z \cdot d\vec{S} \approx \frac{1}{\mathcal{V}_P} \sum_{cf=ewnstb} (\phi S_z)_{cf} \end{aligned} \quad (4.20)$$

The summation extends over the six cell faces, cf: the east, west, north, south, top, and bottom. The dependent variable at the cell face, ϕ_{cf} , is obtained by linear interpolation of the variables at the two intermediate neighboring cell centers (see Eq. 4.19). The same linear interpolation is applied when gradients are needed at the cell face. Using an overbar symbol to denote linear interpolated values, the gradients at the east face read:

$$\begin{aligned} \overline{\left(\frac{\partial \phi}{\partial x} \right)}_e &= (1 - \beta_e) \left(\frac{\partial \phi}{\partial x} \right)_P + \beta_e \left(\frac{\partial \phi}{\partial x} \right)_E \\ \overline{\left(\frac{\partial \phi}{\partial y} \right)}_e &= (1 - \beta_e) \left(\frac{\partial \phi}{\partial y} \right)_P + \beta_e \left(\frac{\partial \phi}{\partial y} \right)_E \\ \overline{\left(\frac{\partial \phi}{\partial z} \right)}_e &= (1 - \beta_e) \left(\frac{\partial \phi}{\partial z} \right)_P + \beta_e \left(\frac{\partial \phi}{\partial z} \right)_E \end{aligned} \quad (4.21)$$

4.3.4 Discretisation of the time derivative terms

The (pseudo-) time derivative term serves as a global iteration that embodies the iterative solution procedure as described in Sect. 4.2. The time iteration can be considered as an iteration level marking the progress of the surface computation since the surface boundary is updated at the end of a time step. The solution of Eq.4.15 at a given time step (n-iteration) designates an intermediate solution. The final solution will be achieved when the iteration converges towards the steady-state solution. For an intermediate solution, a simple first-order finite-difference scheme can be appropriately used to evaluate the time derivative of Eq. 4.15:

$$\frac{\partial}{\partial t} \iiint_{\mathcal{V}} \phi \, d\mathcal{V} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t} \mathcal{V}^n \quad (4.22)$$

The cell volume \mathcal{V} is explicitly defined at time level n (\mathcal{V}^n), since the new geometry of the computational domain is not yet known *a priori*. This applies also to all geometrical parameters such as surface area (S^n) and spatial coordinate and distance (x^n, y^n, z^n, L^n).

In solving Eq. 4.15 for ϕ^{n+1} , iterations have to be carried out to handle the non-linear terms. As shown in Fig. 4.1, there are two iteration loops, the ℓ - and m-iterations, in arriving to ϕ^{n+1} from known values ϕ^n . When these iterations converge, that is $\ell = \infty$ and $m = \infty$, we have $\phi^{n+1} = \phi^{n, \ell \approx \infty, m \approx \infty}$. Eq. 4.22 thus can be approximated as:

$$\frac{\partial}{\partial t} \iiint_{\mathcal{V}} \phi \, d\mathcal{V} \approx \frac{\phi^{n+1} - \phi^n}{\Delta t} \mathcal{V}^n = \frac{\mathcal{V}^{n, \ell=1}}{\Delta t} (\phi^{n, \ell+1} - \phi^{n, \ell=1}) \quad (4.23)$$

The variable index ℓ is used to refer either ℓ - or m-iteration. With this approach, the transport equation, Eq. 4.15, can be rewritten as:

$$\underbrace{\frac{\mathcal{V}^{n, \ell=1}}{\Delta t} (\phi^{n, \ell+1} - \phi^{n, \ell=1})}_{\text{time derivation}} + \underbrace{\left[\iint_S \phi \vec{\mathbf{V}} \cdot d\vec{\mathbf{S}} \right]^{n \rightarrow n+1}}_{\text{convection}} - \underbrace{\left[\iint_S \Gamma \vec{\nabla} \phi \cdot d\vec{\mathbf{S}} \right]^{n \rightarrow n+1}}_{\text{diffusion}} = \underbrace{\left[\iiint_{\mathcal{V}} R \, d\mathcal{V} \right]^{n \rightarrow n+1}}_{\text{source}} \quad (4.24)$$

The pseudo-time index $n \rightarrow n+1$ is introduced to indicate the progress of the iterations n, m, and ℓ , used to evaluate the terms in bracket. Since the geometrical parameters are all evaluated at time level n, the convection-diffusion and the source terms contain explicit terms. The scheme is thus explicit. The pseudo-time step, Δt , is related to the under-relaxation factor used in the iterative procedure of the pressure computation; this will be discussed later in Sect. 4.4.3.

4.3.5 Discretisation of the convective terms

The discrete form of the convective terms in Eq. 4.24 for cell P reads:

$$\left(F^C\right)_P^{n \rightarrow n+1} = \left[\oint_S \phi^{\ell+1}(\vec{V})^n \cdot (d\vec{S})^n \right]_P \approx \sum_{cf=e,wn,sn,b} \left[\phi^{\ell+1}(\vec{V})^n \cdot (\vec{S})^n \right]_{cf} \quad (4.25)$$

The usual convention of the summation index applies, that is the summation runs over the six cell faces: the east, west, north, south, top, and bottom. The evaluation of the convective transport through the east face is elaborated in the following paragraphs and a similar approach applies to the other faces.

$$\begin{aligned} \left(F^C\right)_e^{n \rightarrow n+1} &= \left(\phi \vec{V} \cdot \vec{S} \right)_e^{n \rightarrow n+1} = \left(\vec{V} \cdot \vec{S} \right)_e^{n \rightarrow n+1} \phi_e^{n \rightarrow n+1} \\ &= \left(u_e^{n,\ell} S_{e,x}^n + v_e^{n,\ell} S_{e,y}^n + w_e^{n,\ell} S_{e,z}^n \right) \phi_e^{n,\ell+1} = q_e^\ell \phi_e^{\ell+1} \end{aligned} \quad (4.26)$$

In the above equation, q_e is the discharge (the mass flux per unit mass) *normal* to the east face. For simplicity, the time index ‘n’ is omitted and the notation q_e^ℓ stands for the discharge obtained from $u_e^{n,\ell}$ and S_e^n . A linearisation has been applied to the convective term in Eq. 4.26 by setting ϕ as the only unknown while taking the discharge, q_e , explicitly from the previous iteration.

The unknown variable at the east face, ϕ_e , is estimated by using upwind scheme, that is by taking its value at the upstream control volume which depends on the flow direction (*Versteeg and Malalasekera, 1995, p. 115*):

$$\phi_e^{\ell+1} = \phi_P^{\ell+1} \text{ if } q_e^\ell \geq 0, \quad \phi_e^{\ell+1} = \phi_E^{\ell+1} \text{ if } q_e^\ell < 0 \quad (4.27)$$

The convective flux across the east face, Eq. 4.26, is then:

$$\left(F^C\right)_e^{n \rightarrow n+1} = \max[q_e^\ell, 0] \phi_P^{\ell+1} - \max[-q_e^\ell, 0] \phi_E^{\ell+1} \quad (4.28)$$

Note that the discharge, q_e , is a scalar product of the velocity and the surface vector and it has a positive sign when leaving the cell. Thus the discharge across the west cell face of cell P is equal to the opposite value of that across the east face of cell W. The same is true for the other cell faces. The discharge across the north or top faces of cell P is equal to the opposite value of that across the south or bottom face of cells N or T, respectively. This property has to be kept in the calculation of the convective flux in order to maintain the flux consistency. The convective flux leaving the cell P across the east face is equal to that entering cell W; otherwise a discrepancy occurs between neighboring cells. The convective flux across the west face thus reads:

$$\left(F^C\right)_w^{n \rightarrow n+1} = \max[q_w^\ell, 0] \phi_P^{\ell+1} - \max[-q_w^\ell, 0] \phi_W^{\ell+1} \quad (4.28a)$$

4.3.6 Discretisation of the diffusive terms

The discrete form of the diffusive terms reads in Eq. 4.24 for cell P:

$$\left(F^D\right)^{n \rightarrow n+1} = \oint_S \left[\Gamma \vec{\nabla} \phi \cdot d\vec{S} \right]^{n \rightarrow n+1} \approx \sum_{cf=ewnstb} \left(\Gamma \vec{\nabla} \phi \cdot \vec{S} \right)_{cf}^{n \rightarrow n+1} \quad (4.29)$$

in which the summation extends over the six cell faces and the non-linear terms are linearized as in the evaluation of the convection term. The diffusion across the east face is elaborated and a similar approach applies for the other faces.

In evaluating the diffusive term across the east face, it is convenient to use a *local coordinate system* attached to the east face as shown in Fig. 4.4. Across the east face, Eq. 4.29 reads:

$$\left(F^D\right)_e^{n \rightarrow n+1} = \left(\Gamma \vec{\nabla} \phi \cdot \vec{S} \right)_e^{n \rightarrow n+1} = \Gamma_e^{n,\ell} \left(\frac{\partial \phi}{\partial n} \right)_e^{n,\ell+1} \left(\vec{e}_n \cdot \vec{S} \right)_e^n \quad (4.30)$$

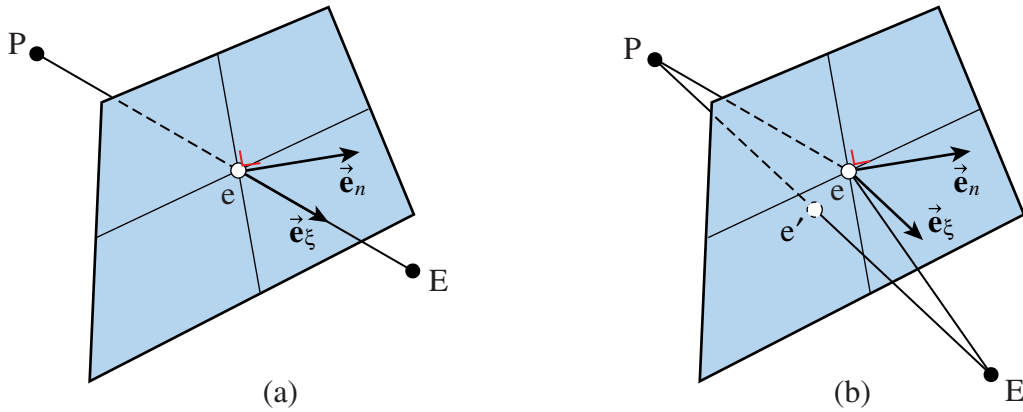


Fig. 4.4 Evaluation of the diffusion terms across the east face

The evaluation of the normal gradient presents some difficulties for its y and z components. Besides the variable at the neighbor cell E, additional ones at NE, SE, S, and N might have to be taken into consideration. This would increase the number of unknowns. To overcome this problem, the so called *deferred-correction approach* (Ferziger and Peric, 1997) is selected in the present model, where only the immediate neighbor cell needs to be considered. In this approach the normal gradient term is evaluated implicitly by a simple approximation and a correction is added. The correction is taken as the difference between the correct and approximate gradients; both are explicitly obtained from the previous iteration. This correction is put in the source terms at the right-hand side. The diffusion term evaluated with this approach reads (Ferziger and Peric, 1997, pp. 218-222):

$$\left(F^D\right)_e^{n \rightarrow n+1} = \Gamma_e^\ell S_e \left(\frac{\partial \phi}{\partial \xi}\right)_e^{\ell+1} + \underbrace{\Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n}\right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi}\right)}_e^\ell \right]}_{\text{correction, explicit}} \quad (4.31)$$

In the above expression, the time index n is omitted for simplicity and a term without any index refers to the initial solution of the time step $n \rightarrow n+1$ (for example S_e is constant during a time step, $S_e = S_e^{n, \ell=1}$). In the local coordinate, ξ is the direction of a straight line joining P and E (see Fig. 4.4). An approximation is used to evaluate the gradient in the implicit term of Eq. 4.31 where a central difference is used (*Ferziger and Peric, 1997; p. 224*).

$$\left(\frac{\partial \phi}{\partial \xi}\right)_e^{\ell+1} = \frac{\phi_E^{\ell+1} - \phi_P^{\ell+1}}{L_{PE}} \quad \text{with} \quad L_{PE} = \|\vec{L}_{PE}\| \quad (4.32)$$

Substituting this relation to the implicit gradient, the diffusive flux reads:

$$\left(F^D\right)_e^{n \rightarrow n+1} = \left(\frac{\Gamma_e^\ell S_e}{L_{PE}}\right) \phi_E^{\ell+1} - \left(\frac{\Gamma_e^\ell S_e}{L_{PE}}\right) \phi_P^{\ell+1} + \underbrace{\Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n}\right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi}\right)}_e^\ell \right]}_{\text{correction, explicit}} \quad (4.33)$$

and for the west face, it reads:

$$\left(F^D\right)_w^{n \rightarrow n+1} = \left(\frac{\Gamma_w^\ell S_w}{L_{PW}}\right) \phi_W^{\ell+1} - \left(\frac{\Gamma_w^\ell S_w}{L_{PW}}\right) \phi_P^{\ell+1} + \underbrace{\Gamma_w^\ell S_w \left[\overline{\left(\frac{\partial \phi}{\partial n}\right)}_w^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi}\right)}_w^\ell \right]}_{\text{correction explicit}} \quad (4.33a)$$

The explicit parts can be easily obtained since the Cartesian components of the gradient are known from the previous computation.

$$\begin{aligned} \overline{\left(\frac{\partial \phi}{\partial n}\right)}_e^\ell &= \overline{\left(\frac{\partial \phi}{\partial x}\right)}_e^\ell \frac{S_{e,x}}{S_e} + \overline{\left(\frac{\partial \phi}{\partial y}\right)}_e^\ell \frac{S_{e,y}}{S_e} + \overline{\left(\frac{\partial \phi}{\partial z}\right)}_e^\ell \frac{S_{e,z}}{S_e} \\ \overline{\left(\frac{\partial \phi}{\partial \xi}\right)}_e^\ell &= \overline{\left(\frac{\partial \phi}{\partial x}\right)}_e^\ell \frac{\Delta x_{PE}}{L_{PE}} + \overline{\left(\frac{\partial \phi}{\partial y}\right)}_e^\ell \frac{\Delta y_{PE}}{L_{PE}} + \overline{\left(\frac{\partial \phi}{\partial z}\right)}_e^\ell \frac{\Delta z_{PE}}{L_{PE}} \end{aligned} \quad (4.34)$$

Applying Eq. 4.31 to the six cell faces gives 7 unknowns to the diffusion transport term for each computational cell P. Errors due to the use of a simple central difference to obtain diffusion across a cell face, Eq. 4.32, are minimized by the correction given in the explicit part of Eq. 4.31. It shall be nevertheless noted that the error will be magnified when the ξ direction of the cell face is far from its n -direction or when the east face center does not coincide with the ξ line (see Fig. 4.4b).

4.3.7 Convective-diffusive terms: hybrid and power-law schemes

The convective upwind scheme, Eq. 4.27, is simple and easy to implement; it accounts for the flow direction. The scheme, however, is first order accurate and produces considerable error when diffusive transport is important. To avoid that problem, the so-called *hybrid scheme* (Spalding, 1972) and *power-law scheme* (Patankar, 1980) give formulae which combine the convective and diffusive transports in a special way. Depending on the grid Peclet number Pe , being the ratio of the convective and diffusive conductance, $Pe = qL/\Gamma S$ where L is the nodal distance, either the upwind-scheme convection, central-difference diffusion, or combination of the two, is considered to transport any scalar quantity ϕ across a cell face.

The *hybrid scheme* (Spalding, 1972) uses the upwind scheme for large Peclet numbers ($|Pe| \geq 2$) and central difference for small Peclet numbers ($|Pe| < 2$). According to this scheme, the total flux across the east face, $F_e = F_e^C - F_e^D$, is defined as follows:

- for $Pe_e = q_e L_{PE} / \Gamma_e S_e < -2$, only the convective transport is taken into account:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_E^{\ell+1} \quad (4.35)$$

- for $-2 \leq Pe_e = q_e L_{PE} / \Gamma_e S_e < 0$, a part of the diffusive transport is also taken into consideration:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_E^{\ell+1} - (1 + 0.5Pe_e) \left\{ \frac{\Gamma_e^\ell S_e}{L_{PE}} (\phi_E^{\ell+1} - \phi_P^{\ell+1}) + \Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_e^\ell \right] \right\} \quad (4.36)$$

- for $0 \leq Pe_e = q_e L_{PE} / \Gamma_e S_e < 2$, a part of the diffusive transport is also taken into consideration:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_P^{\ell+1} - (1 - 0.5Pe_e) \left\{ \frac{\Gamma_e^\ell S_e}{L_{PE}} (\phi_E^{\ell+1} - \phi_P^{\ell+1}) + \Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_e^\ell \right] \right\} \quad (4.37)$$

- for $Pe_e = q_e L_{PE} / \Gamma_e S_e > 2$, only the convective transport is taken into account:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_P^{\ell+1} \quad (4.38)$$

The *power-law scheme* (Patankar, 1980, p. 90-91) sets the limiting value of Pe where the diffusion no longer affects the transport at $Pe = 10$, instead of $Pe = 2$ used in the hybrid scheme.

- for $Pe_e = q_e L_{PE} / \Gamma_e S_e < -10$:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_E^{\ell+1} \quad (4.39)$$

- for $-10 \leq \text{Pe}_e = q_e L_{\text{PE}} / \Gamma_e S_e < 0$:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_E^{\ell+1} - (1 + 0.1\text{Pe}_e)^5 \left\{ \frac{\Gamma_e^\ell S_e}{L_{\text{PE}}} (\phi_E^{\ell+1} - \phi_P^{\ell+1}) + \Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_e^\ell \right] \right\} \quad (4.40)$$

- for $0 \leq \text{Pe}_e = q_e L_{\text{PE}} / \Gamma_e S_e < 10$:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_P^{\ell+1} - (1 - 0.1\text{Pe}_e)^5 \left\{ \frac{\Gamma_e^\ell S_e}{L_{\text{PE}}} (\phi_E^{\ell+1} - \phi_P^{\ell+1}) + \Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_e^\ell \right] \right\} \quad (4.41)$$

- for $\text{Pe}_e = q_e L_{\text{PE}} / \Gamma_e S_e > 10$:

$$F_e^{n \rightarrow n+1} = q_e^\ell \phi_P^{\ell+1} \quad (4.42)$$

Equations 4.35 to 4.42 can be combined into a compact form as follows (*Patankar, 1980, pp. 94-95*):

$$F_e^{n \rightarrow n+1} = \max[q_e^\ell, 0] \phi_P^{\ell+1} - \max[-q_e^\ell, 0] \phi_E^{\ell+1} - f_e^{\text{D}} \left\{ \frac{\Gamma_e^\ell S_e}{L_{\text{PE}}} (\phi_E^{\ell+1} - \phi_P^{\ell+1}) + \Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_e^\ell \right] \right\} \quad (4.43)$$

where f^{D} is a factor that depends on the absolute value of grid Peclet number; it has a different form for the hybrid and power-law schemes as shown in Table 4.3.

Table 4.3 Hybrid and power-law scheme diffusion factors.

Scheme	$f^{\text{D}} = f(\text{Pe}) = f(qL/\Gamma S)$
Hybrid	$\max[(1 - 0.5 \text{Pe}), 0]$
Power-law	$\max[(1 - 0.1 \text{Pe})^5, 0]$

Arranging the terms in Eq. 4.43, one has:

$$\begin{aligned}
 F_e^{n \rightarrow n+1} = & \underbrace{\left\{ \max[q_e^\ell, 0] + f_e^D \frac{\Gamma_e^\ell S_e}{L_{PE}} \right\} \phi_P^{\ell+1} + \left\{ -\max[-q_e^\ell, 0] - f_e^D \frac{\Gamma_e^\ell S_e}{L_{PE}} \right\} \phi_E^{\ell+1}}_{\text{implicit}} \\
 & - \underbrace{f_e^D \left\{ \Gamma_e^\ell S_e \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_e^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_e^\ell \right]}_{\text{explicit}} \right\}
 \end{aligned} \tag{4.43a}$$

and for the west face, the convective-diffusive flux reads:

$$\begin{aligned}
 F_w^{n \rightarrow n+1} = & \underbrace{\left\{ \max[q_w^\ell, 0] + f_w^D \frac{\Gamma_w^\ell S_w}{L_{PW}} \right\} \phi_P^{\ell+1} + \left\{ -\max[-q_w^\ell, 0] - f_w^D \frac{\Gamma_w^\ell S_w}{L_{PW}} \right\} \phi_W^{\ell+1}}_{\text{implicit}} \\
 & - \underbrace{f_w^D \left\{ \Gamma_w^\ell S_w \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)}_w^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)}_w^\ell \right]}_{\text{explicit}} \right\}
 \end{aligned} \tag{4.43b}$$

Shorter notations is used to write the expression of the convective-diffusive flux, for example Eq. 4.43a maybe rewritten as $F_e^{n \rightarrow n+1} = (a_P^C + a_P^D) \phi_P^{\ell+1} + (a_E^C + a_E^D) \phi_E^{\ell+1} + b_E^D$, in such a way that by summing up the convective-diffusive fluxes across the six faces of cell P, one obtains:

$$\begin{aligned}
 F^{n \rightarrow n+1} = & (a_E^C + a_E^D) \phi_E^{\ell+1} + (a_W^C + a_W^D) \phi_W^{\ell+1} + (a_N^C + a_N^D) \phi_N^{\ell+1} + (a_S^C + a_S^D) \phi_S^{\ell+1} + \\
 & (a_T^C + a_T^D) \phi_T^{\ell+1} + (a_B^C + a_B^D) \phi_B^{\ell+1} + (a_P^C + a_P^D) \phi_P^{\ell+1} + b^D
 \end{aligned} \tag{4.44}$$

where the coefficients indicate contribution of convective-diffusive terms from neighboring cells as presented in Table 4.4. The diffusive-correction terms, being evaluated explicitly, are known from the previous iteration and are included in the independent coefficient b^D .

Table 4.4 Coefficients of the discretized convective-diffusive equations.

Convective terms	Diffusive terms
$a_E^C = -\max[-q_e^\ell, 0]$	$a_E^D = -f^D \Gamma_e^\ell S_e / L_{PE}$
$a_W^C = -\max[-q_w^\ell, 0]$	$a_W^D = -f^D \Gamma_w^\ell S_w / L_{PW}$
$a_N^C = -\max[-q_n^\ell, 0]$	$a_N^D = -f^D \Gamma_n^\ell S_n / L_{PN}$
$a_S^C = -\max[-q_s^\ell, 0]$	$a_S^D = -f^D \Gamma_s^\ell S_s / L_{PS}$
$a_T^C = -\max[-q_t^\ell, 0]$	$a_T^D = -f^D \Gamma_t^\ell S_t / L_{PT}$
$a_B^C = -\max[-q_b^\ell, 0]$	$a_B^D = -f^D \Gamma_b^\ell S_b / L_{PB}$
$a_P^C = \sum_{nb=EWNSTB} -a_{nb}^C + \sum_{cf=ewnstb} q_{cf}^\ell$	$a_P^D = \sum_{nb=EWNSTB} -a_{nb}^D$
—	$b^D = \sum_{cf=ewnstb} -\left(f^D \Gamma^\ell S\right)_{cf} \left[\overline{\left(\frac{\partial \phi}{\partial n}\right)}_{cf}^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi}\right)}_{cf}^\ell \right]$

4.3.8 Source terms

The source terms may consist of scalar quantities, first derivatives, or second derivatives of a scalar quantity. The source, in addition, may also include the time integration term, the diffusion correction, and known variables from the boundary conditions. Sometimes a term initially considered as a source takes advantage to be expressed as a function of the unknown variable at the cell center, such as cells next to a boundary. In that case, the term is linearized which gives (*Versteeg and Malalasekera, 1995, p. 87*):

$$\iiint_{\mathcal{V}} R \, d\mathcal{V} \approx \bar{R} \mathcal{V}_P = b + b_P \phi_P^{\ell+1} \quad (4.45)$$

in which b includes all known quantities (either constants, prescribed, or known from previous iteration) and b_P is the coefficient of the unknown variable at P .

Scalar source terms

The source term containing scalar quantities, b^S , comes from the gravity accelerations, g_x , g_y , g_z , or the turbulent energy production and dissipation, G and ϵ . The scalar source at P represents the average value of those quantities in the cell being considered. It is known and thus is considered as a constant. Therefore, the source terms containing scalar quantities can be easily evaluated according to the following expression:

$$b^S = \iiint_{\mathcal{V}} \phi^\ell d\mathcal{V} = \phi_P^\ell \mathcal{V}_P \quad (4.46)$$

In applying the above relation to the source terms of the k equation coming from the turbulent energy production, G , some approximations are needed. Writing Eq. 4.46 for G , we have:

$$\iiint_{\mathcal{V}} G d\mathcal{V} = G_P \mathcal{V}_P$$

Since the G term contains non-linear gradient terms (see Eq. 4.12), the above relation implies that these terms are evaluated individually (that is by using Eq. 4.20). This means that the integral of these terms are computed in the following fashion (an example is given here for the $(\partial u/\partial x)^2$ term):

$$\iiint_{\mathcal{V}} \left(\frac{\partial u}{\partial x} \right)^2 d\mathcal{V} = \left(\frac{\partial u}{\partial x} \right)_P^2 \mathcal{V}_P = \left(\frac{1}{\mathcal{V}_P} \iiint_{\mathcal{V}} \frac{\partial u}{\partial x} d\mathcal{V} \right)^2 \mathcal{V}_P$$

This approximation, of course, will be inaccurate when the velocity gradient is important. Nevertheless, this method is selected for its easiness to implement.

In the ε equation, the source term is linearized for the term containing ε^2 in the following form:

$$b = c_1 \underbrace{\left(\frac{\varepsilon G}{k} \right)_P^\ell}_{b^s} \mathcal{V}_P - c_2 \underbrace{\left(\frac{\varepsilon}{k} \right)_P^\ell}_{b_p^s} \mathcal{V}_P \varepsilon_P^{\ell+1} \quad (4.47)$$

the last term of which will join the coefficient a_p .

First derivative source terms

The source term containing first derivatives, b^{1D} , is found in the pressure gradient of the momentum equation and in the velocity gradient of the energy production for the k - ε equations. Following the method described in Sect. 4.3.3, the source terms containing first derivatives in the x -, y -, and z -directions are evaluated as follows (the terms within brackets are generally predominant):

$$\begin{aligned} (b^{1D})_x &= \iiint_{\mathcal{V}} \left(\frac{\partial \phi}{\partial x} \right)^\ell d\mathcal{V} = \iiint_{\mathcal{V}} \vec{\nabla}(\phi^\ell \vec{e}_x) d\mathcal{V} \approx \iint_S \phi^\ell \vec{e}_x \cdot d\vec{S} = \sum_{cf=ewnstb} (\phi^\ell S_x)_{cf} \\ &= (\phi_e^\ell S_{e,x} + \phi_w^\ell S_{w,x}) + \phi_n^\ell S_{w,x} + \phi_s^\ell S_{s,x} + \phi_t^\ell S_{t,x} + \phi_b^\ell S_{b,x} \end{aligned} \quad (4.48a)$$

$$\begin{aligned}
 (b^{1D})_y &= \iiint_{\mathbf{x}} \left(\frac{\partial \phi}{\partial y} \right)^\ell d\mathbf{V} = \iiint_{\mathbf{x}} \vec{\nabla}(\phi^\ell \vec{e}_y) d\mathbf{V} \approx \iint_S \phi^\ell \vec{e}_y \cdot d\vec{\mathbf{S}} = \sum_{cf=ewnstb} (\phi^\ell S_y)_{cf} \\
 &= (\phi_n^\ell S_{n,y} + \phi_s^\ell S_{s,y}) + \phi_t^\ell S_{t,y} + \phi_b^\ell S_{b,y} + \phi_e^\ell S_{e,y} + \phi_w^\ell S_{w,y}
 \end{aligned} \tag{4.48b}$$

$$\begin{aligned}
 (b^{1D})_z &= \iiint_{\mathbf{x}} \left(\frac{\partial \phi}{\partial z} \right)^\ell d\mathbf{V} = \iiint_{\mathbf{x}} \vec{\nabla}(\phi^\ell \vec{e}_z) d\mathbf{V} \approx \iint_S \phi^\ell \vec{e}_z \cdot d\vec{\mathbf{S}} = \sum_{cf=ewnstb} (\phi^\ell S_z)_{cf} \\
 &= (\phi_t^\ell S_{t,z} + \phi_b^\ell S_{b,z}) + \phi_e^\ell S_{e,z} + \phi_w^\ell S_{w,z} + \phi_n^\ell S_{n,z} + \phi_s^\ell S_{s,z}
 \end{aligned} \tag{4.48c}$$

Second derivative source terms

The source term containing second derivatives, b^{2D} , is found in the momentum equation. These are due to the non-orthogonal terms of the stresses (see Eqs. 4.6 to 4.8) and the explicit parts of the diffusion terms (see Eq. 4.43). An example is given below for the evaluation of the source terms containing second derivatives in the u-momentum equation.

$$\begin{aligned}
 (b^{2D})_x &= \iiint_{\mathbf{x}} \frac{\partial}{\partial x} \left(v_t \frac{\partial u}{\partial x} \right)^\ell d\mathbf{V} = \iiint_{\mathbf{x}} \vec{\nabla} \left[\left(v_t \frac{\partial u}{\partial x} \right)^\ell \vec{e}_x \right] d\mathbf{V} \\
 &= \iint_S \left(v_t \frac{\partial u}{\partial x} \right)^\ell \vec{e}_x \cdot d\vec{\mathbf{S}} \approx \sum_{cf} \left[\left(v_t \frac{\partial u}{\partial x} \right)^\ell S_x \right]_{cf} \\
 &= (\overline{v_t})_e^\ell \left(\frac{\partial u}{\partial x} \right)_e^\ell S_{e,x} + (\overline{v_t})_w^\ell \left(\frac{\partial u}{\partial x} \right)_w^\ell S_{w,x} + (\overline{v_t})_n^\ell \left(\frac{\partial u}{\partial x} \right)_n^\ell S_{n,x} + \\
 &\quad (\overline{v_t})_s^\ell \left(\frac{\partial u}{\partial x} \right)_s^\ell S_{s,x} + (\overline{v_t})_t^\ell \left(\frac{\partial u}{\partial x} \right)_t^\ell S_{t,x} + (\overline{v_t})_b^\ell \left(\frac{\partial u}{\partial x} \right)_b^\ell S_{b,x}
 \end{aligned} \tag{4.49a}$$

$$\begin{aligned}
 (b^{2D})_y &= \iiint_{\mathbf{x}} \frac{\partial}{\partial y} \left(v_t \frac{\partial v}{\partial x} \right)^\ell d\mathbf{V} = \iiint_{\mathbf{x}} \vec{\nabla} \left[\left(v_t \frac{\partial v}{\partial x} \right)^\ell \vec{e}_y \right] d\mathbf{V} \\
 &= \iint_S \left(v_t \frac{\partial v}{\partial x} \right)^\ell \vec{e}_y \cdot d\vec{\mathbf{S}} \approx \sum_{cf} \left[\left(v_t \frac{\partial v}{\partial x} \right)^\ell S_y \right]_{cf} \\
 &= (\overline{v_t})_n^\ell \left(\frac{\partial v}{\partial x} \right)_n^\ell S_{n,y} + (\overline{v_t})_s^\ell \left(\frac{\partial v}{\partial x} \right)_s^\ell S_{s,y} + (\overline{v_t})_t^\ell \left(\frac{\partial v}{\partial x} \right)_t^\ell S_{t,y} + \\
 &\quad (\overline{v_t})_b^\ell \left(\frac{\partial v}{\partial x} \right)_b^\ell S_{b,y} + (\overline{v_t})_e^\ell \left(\frac{\partial v}{\partial x} \right)_e^\ell S_{e,y} + (\overline{v_t})_w^\ell \left(\frac{\partial v}{\partial x} \right)_w^\ell S_{w,y}
 \end{aligned} \tag{4.49b}$$

$$\begin{aligned}
 (b^{2D})_z &= \iiint_{\mathcal{V}} \frac{\partial}{\partial z} \left(\mathbf{v}_t \frac{\partial w}{\partial x} \right)^\ell d\mathcal{V} = \iiint_{\mathcal{V}} \vec{\nabla} \left[\left(\mathbf{v}_t \frac{\partial w}{\partial x} \right)^\ell \bar{\mathbf{e}}_z \right] d\mathcal{V} \\
 &= \oint_S \left(\mathbf{v}_t \frac{\partial w}{\partial x} \right)^\ell \bar{\mathbf{e}}_z \cdot d\vec{S} \approx \sum_{cf} \left[\left(\mathbf{v}_t \frac{\partial w}{\partial x} \right)^\ell S_z \right]_{cf} \\
 &= \overline{(\mathbf{v}_t)_t}^\ell \overline{\left(\frac{\partial w}{\partial x} \right)_t}^\ell S_{t,z} + \overline{(\mathbf{v}_t)_b}^\ell \overline{\left(\frac{\partial w}{\partial x} \right)_b}^\ell S_{b,z} + \overline{(\mathbf{v}_t)_e}^\ell \overline{\left(\frac{\partial w}{\partial x} \right)_e}^\ell S_{e,z} + \\
 &\quad \overline{(\mathbf{v}_t)_w}^\ell \overline{\left(\frac{\partial w}{\partial x} \right)_w}^\ell S_{w,z} + \overline{(\mathbf{v}_t)_n}^\ell \overline{\left(\frac{\partial w}{\partial x} \right)_n}^\ell S_{n,z} + \overline{(\mathbf{v}_t)_s}^\ell \overline{\left(\frac{\partial w}{\partial x} \right)_s}^\ell S_{s,z}
 \end{aligned} \tag{4.49c}$$

The last four terms on the right-hand side are due to the grid non-orthogonality; they vanish for orthogonal cells. The cell face values of the velocity gradients are obtained by linear interpolation (see Sect. 4.3.3 and Eq. 4.21).

4.3.9 Assembly of the coefficients

After evaluating all terms of the convection, diffusion, and sources over the entire computational domain and rearranging the coefficients, the discretized transport equation produces a series of algebraic equations. For the unknown variable ϕ at the cell center P, $\phi_P(x,y,z,t)$, and at the neighboring cell centers nb, $\phi_{nb}(x,y,z,t)$, the equation reads:

$$a_P \phi_P^{\ell+1} + \sum_{nb} a_{nb} \phi_{nb}^{\ell+1} = b \tag{4.50}$$

The coefficients a_{nb} , a_P , and b in the above equation are listed below:

- coefficients a_{nb} consist of the convective and diffusive terms (see Table 4.4):

$$\begin{aligned}
 a_E &= a_E^C + a_E^D = -\max[-q_e^\ell, 0] - f_e^D \left(\frac{\Gamma S}{L_{PE}} \right)_e^\ell, \\
 a_W &= a_W^C + a_W^D = -\max[-q_w^\ell, 0] - f_w^D \left(\frac{\Gamma S}{L_{PW}} \right)_w^\ell, \\
 a_N &= a_N^C + a_N^D = -\max[-q_n^\ell, 0] - f_n^D \left(\frac{\Gamma S}{L_{PN}} \right)_n^\ell, \\
 a_S &= a_S^C + a_S^D = -\max[-q_s^\ell, 0] - f_s^D \left(\frac{\Gamma S}{L_{PS}} \right)_s^\ell, \\
 a_T &= a_T^C + a_T^D = -\max[-q_t^\ell, 0] - f_t^D \left(\frac{\Gamma S}{L_{PT}} \right)_t^\ell, \quad \text{and}
 \end{aligned}$$

$$a_B = a_B^C + a_B^D = -\max[-q_b^\ell, 0] - f_b^D \left(\frac{\Gamma S}{L_{PB}} \right)_b^\ell.$$

- coefficient a_p is formed from various terms, namely the pseudo-temporal integration, convective-diffusive terms (Table 4.4), and terms coming from the source linearisation (see Eq. 4.47):

$$\begin{aligned} a_p &= a_p^T + a_p^C + a_p^D - b_p \\ &= \frac{\mathbf{V}_p}{\Delta t} + \left(\sum_{nb} -a_{nb}^C + \sum_{cf} q_{cf} \right) + \left(\sum_{nb} -a_{nb}^D \right) - b_p \end{aligned}$$

- source terms, b :

$$b = b^S + (b^{1D})_x + (b^{1D})_y + (b^{1D})_z + (b^{2D})_x + (b^{2D})_y + (b^{2D})_z + b^T - b^D$$

b^S — scalar sources: Eq. 4.46 or 4.47,

b^{1D} — first derivative sources: Eqs. 4.48,

b^{2D} — second derivative sources: Eqs. 4.49,

$b^T = \frac{\mathbf{V}_p}{\Delta t} \phi_p^n$ — pseudo-time derivative source: Eq. 4.22, and

$b^D = \sum_{cf} - (f^D \Gamma^\ell S)_{cf} \left[\overline{\left(\frac{\partial \phi}{\partial n} \right)_{cf}}^\ell - \overline{\left(\frac{\partial \phi}{\partial \xi} \right)_{cf}}^\ell \right]$ — diffusive correction terms: Eq. 4.43.

Note that for the boundary cells, the coefficients may change from the above definitions. This will be described in Sect. 4.5.

Under-relaxation factor

The solution of Eq. 4.50 for any dependent variable ϕ through out the computational domain is achieved by iterative procedure, marching from known values at the iteration level ℓ to new values at the iteration $\ell + 1$. During the process, oscillation may occur. In order to avoid such a problem, an under-relaxation factor is applied to updating the solution from iteration ℓ to $\ell + 1$. Suppose that the solution at a particular iteration level is $\tilde{\phi}_p$, thus:

$$a_p \tilde{\phi}_p + \sum_{nb} a_{nb} \phi_{nb}^{\ell+1} = b \quad (4.50a)$$

Now, instead of taking that solution for the value of $\phi_p^{\ell+1}$, one may take also into the consideration its value at the previous iteration level, ϕ_p^ℓ , arguing that $\phi_p^{\ell+1}$ should not too much different from ϕ_p^ℓ . The under-relaxation factor, ϖ , is then applied according to the following form:

$$\phi_P^{\ell+1} = \varpi \tilde{\phi}_P + (1 - \varpi) \phi_P^\ell, \quad \text{or} \quad \tilde{\phi}_P = \frac{1}{\varpi} \phi_P^{\ell+1} - \frac{1 - \varpi}{\varpi} \phi_P^\ell \quad (4.51)$$

Substituting this relation to the term $\tilde{\phi}_P$ in Eq. 4.50a, one finds:

$$a_P \left(\frac{1}{\varpi} \phi_P^{\ell+1} - \frac{1 - \varpi}{\varpi} \phi_P^\ell \right) + \sum_{nb} a_{nb} \phi_{nb}^{\ell+1} = b$$

which, after some arrangements of the terms, yields:

$$\tilde{a}_P \phi_P^{\ell+1} + \sum_{nb} a_{nb} \phi_{nb}^{\ell+1} = \tilde{b} \quad (4.52)$$

where the coefficients are now:

$$\tilde{a}_P = \frac{a_P}{\varpi} \quad \text{and} \quad \tilde{b} = b + (1 - \varpi) \frac{a_P}{\varpi} \phi_P^\ell = b + (1 - \varpi) \tilde{a}_P \phi_P^\ell$$

4.4 Pressure-velocity coupling

4.4.1 SIMPLE algorithm

When solving the momentum equation for velocity, the pressure is unknown and an estimated value, p^* , is firstly used instead. In general, the velocity that is obtained does not satisfy the continuity equation. A correction to the estimated pressure is added and a new solution is sought for the new velocity. This procedure is repeated until it gives pressure and velocity fields satisfying not only the momentum equation but also the continuity equation. An iterative solution procedure known as SIMPLE (**S**emi-**I**mplicit **M**ethod for **P**ressure-**L**inked **E**quation) method (*Patankar and Spalding, 1972*) is widely used for this velocity-pressure computation. The method requires velocity and discharge at cell faces, which are not immediately available with the use of non-staggered grids in the present model. The interpolation technique of Rhie-and-Chow (*Rhie and Chow, 1983*) solves this problem. The technique gives interpolated velocity at cell faces from the nodal values. The standard SIMPLE algorithm is then used to perform the pressure correction. This section gives some details of the procedure, which follows the derivation given by Patankar (*Patankar and Spalding, 1972; Versteeg and Malalasekera, 1995; Ferziger and Peric, 1997*).

In the iteration $\ell \rightarrow \ell + 1$, the discretized momentum equation, Eq. 4.52 with $\phi = u, v, w$, can be rewritten as:

$$\tilde{a}_P u_{i,P}^{\ell+1} + \sum_{nb} a_{nb} u_{i,nb}^{\ell+1} = \tilde{b} - \frac{1}{\rho} \nabla_P \left(\frac{\partial p^{\ell+1}}{\partial x_i} \right)_P \quad (4.53)$$

where the symbols u_i and x_i are used to denote the Cartesian components of the velocity and direction, $u_i = u, v, w$ and $x_i = x, y, z$, respectively. Note that the pressure gradient in the above expression has intentionally been extracted from the source term, \tilde{b} , for a reason that will be evidenced later (\tilde{b} in Eq. 4.53 is thus not exactly the same as that in Eq. 4.52).

The coefficients \tilde{a}_p, a_{nb} , and the source terms, \tilde{b} , are functions of the known variables either at the precedent iteration, ℓ , or time step, n . For practical solutions of Eq. 4.53, since there are only 3 equations for 4 unknowns, the pressure p is temporarily fixed at its initial value. The following system of equations is solved in the first stage:

$$\tilde{a}_p u_{i,P}^* + \sum_{nb} a_{nb} u_{i,nb}^* = \tilde{b} - \frac{\mathbf{V}_p}{\rho} \left(\frac{\partial p^*}{\partial x_i} \right)_p \quad (4.54a)$$

$$p^* = p^\ell \quad (4.54b)$$

The estimated pressure, p^* , and the velocities obtained from this pressure, u^*, v^*, w^* , are of course to be corrected:

$$u_i^{\ell+1} = u_i^* + u_i^c \quad (4.55a)$$

$$p^{\ell+1} = p^* + p^c \quad (4.55b)$$

where the corrections, u_i^c and p^c , will result from the momentum equations combined with the continuity equation. The corrections are such that the velocity will satisfy the continuity and momentum equations when the iteration converges. It can therefore be said that Eqs. 4.54a,b become Eq. 4.53 when $\ell \rightarrow \infty$. In that case, having a sufficiently large number of ℓ -iterations, we have $(u_i^*)^{\ell \rightarrow \infty} = u_i^{\ell+1}$; and this is so for the coefficients and source terms. We can therefore obtain the relation between the pressure and velocity corrections by subtraction of Eqs. 4.54a,b from Eq. 4.53:

$$\tilde{a}_p u_{i,P}^c + \sum_{nb} a_{nb} u_{i,nb}^c = - \frac{\mathbf{V}_p}{\rho} \left(\frac{\partial p^c}{\partial x_i} \right)_p \quad (4.56)$$

Eq. 4.56 is a relation in which the corrections tend towards zero. A simplifying approximation can then be introduced by neglecting *a priori* the correction terms of the neighboring cells (*Patankar and Spalding, 1972*). The velocity correction thus reduces to:

$$u_{i,P}^c = - \frac{1}{\rho} \frac{\mathbf{V}_p}{\tilde{a}_p} \left(\frac{\partial p^c}{\partial x_i} \right)_p \quad (4.57)$$

Since the coefficient \tilde{a}_p is derived in such a way that it is the same for all velocity components, i.e. $\tilde{a}_p^u = \tilde{a}_p^v = \tilde{a}_p^w = \tilde{a}_p$, the above relation is valid for *any velocity*

component at any point and thus also for the normal velocity component at a cell face. Writing for the east face, one has:

$$u_{n,e}^c = -\frac{1}{\rho} \left(\frac{\mathbf{V}}{\tilde{a}} \right)_e \left(\frac{\partial p^c}{\partial n} \right)_e \quad (4.58)$$

The coefficient at the cell face, $(\mathbf{V}/\tilde{a})_e$, is defined as the average value of those of the neighboring cell centers P and E:

$$\left(\frac{\mathbf{V}}{\tilde{a}} \right)_e = \frac{1}{2} \left[\left(\frac{\mathbf{V}}{\tilde{a}_P} \right)_P + \left(\frac{\mathbf{V}}{\tilde{a}_E} \right)_E \right] \quad (4.59)$$

Note that $(\tilde{a}_P)_E$ represents the coefficient \tilde{a}_P of Eq. 4.52 written for the cell E, that is not the coefficient a_E of Eq. 4.52 written for the cell P. Note also that the interpolation in Eq. 4.59 does not match the linear interpolation in Eq. 4.19 since this latter is not relevant for the volumes. Indeed the volume related to a cell for the east face is composed of the half volume of cell P and the half volume of cell E. Using the deferred-correction method as in the discretisation of the diffusive terms (see Sect. 4.3.6, Eq. 4.31) to compute the normal pressure gradient yields:

$$u_{n,e}^c = \underbrace{-\frac{1}{\rho} \left(\frac{\mathbf{V}}{\tilde{a}} \right)_e \left[\frac{p_E^c - p_P^c}{L_{PE}} \right]}_{u_{n,e}^{\text{impl}} : \text{implicit}} - \underbrace{\frac{1}{\rho} \left(\frac{\mathbf{V}}{\tilde{a}} \right)_e \left[\overline{\left(\frac{\partial p^c}{\partial n} \right)}_e - \overline{\left(\frac{\partial p^c}{\partial \xi} \right)}_e \right]}_{u_{n,e}^{\text{no}} : \text{explicit}}^{\text{old}} \quad (4.60)$$

in which the terms in the second square bracket are due to non-orthogonality of the cell and are evaluated explicitly. From the starred-velocity and the velocity correction, the normal velocity component can be computed:

$$u_{n,e} = u_{n,e}^* + u_{n,e}^c = u_{n,e}^* + u_{n,e}^{\text{impl}} + u_{n,e}^{\text{no}} \quad (4.61)$$

Here the velocity correction due to the cell non-orthogonality is written separately. Whilst the velocity correction is obtainable from Eq. 4.60, the starred-velocity unfortunately is not directly available at the cell face. Interpolating the starred velocity at the neighboring cell centers to get the cell face value would result in the decoupling of the velocity from the pressure that causes an oscillation of the solution. A remedy to this problem is to use the so-called Rhie-and-Chow interpolation technique (*Rhie and Chow*, 1983) which replaces the interpolated pressure gradient at a cell face with the one computed from the pressure at the immediate neighboring cell centers. As can be seen in the relation below, writing the equivalent of Eq. 4.54a at the east face by simply interpolating these starred velocities would result in velocities that have no direct relation with the pressure difference between P and E:

$$\overline{(\mathbf{u}_n^*)}_e = \overline{\left(\frac{\tilde{\mathbf{b}} - \sum a_{nb} \mathbf{u}_{n,nb}^*}{\tilde{a}_p} \right)}_e - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_e \overline{\left(\frac{\partial p^*}{\partial n} \right)}_e \quad (4.62)$$

In order to relate the velocity at the east face back to the pressure difference between P and E, a correction is given to this interpolated velocity:

$$\mathbf{u}_{n,e}^* = \left[\overline{\left(\frac{\tilde{\mathbf{b}} - \sum a_{nb} \mathbf{u}_{n,nb}^*}{\tilde{a}_p} \right)}_e - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_e \overline{\left(\frac{\partial p^*}{\partial n} \right)}_e \right] + \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_e \left[\overline{\left(\frac{\partial p^*}{\partial n} \right)}_e - \frac{p_E^* - p_P^*}{L_{PE}} \right] \quad (4.63)$$

This expression can be seen as the velocities at the cell centers P and E interpolated to the east face and corrected by a factor due to the interpolation:

$$\mathbf{u}_{n,e}^* = \overline{(\mathbf{u}_n^*)}_e + \mathbf{u}_{n,e}^{RC} \quad (4.64)$$

where the overbar term is obtained from linear interpolation of the velocities in cell centers P and E. Using this expression to substitute the starred velocity in Eq. 4.61 gives:

$$\mathbf{u}_{n,e} = \overline{(\mathbf{u}_n^*)}_e + \mathbf{u}_{n,e}^{RC} + \mathbf{u}_{n,e}^{impl} + \mathbf{u}_{n,e}^{no} \quad (4.65)$$

Next, we need to define the equation of the pressure correction, which is carried out by using the continuity equation. The discretized continuity equation can be obtained by writing the governing equation, Eq. 4.15, with $\phi = 1$, $\Gamma = 0$, and $R = 0$ that gives:

$$\oint_S \vec{\mathbf{V}} \cdot d\vec{\mathbf{S}} \approx \sum_{cf=ewnstb} \left(\vec{\mathbf{V}} \cdot d\vec{\mathbf{S}} \right)_{cf} = \sum_{cf=ewnstb} q_{cf} = \sum_{cf=ewnstb} (\mathbf{u}_{n,cf} S_{cf}) = 0 \quad (4.66)$$

Using Eq. 4.65 with the development of the implicit term in Eq. 4.60, and inserting the result to the above continuity equation, one gets the pressure correction equation of the form:

$$\begin{aligned} & -\frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_e \frac{S_e}{L_{PE}} (p_E^c - p_P^c) - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_w \frac{S_w}{L_{PW}} (p_W^c - p_P^c) - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_n \frac{S_n}{L_{PN}} (p_N^c - p_P^c) \\ & - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_s \frac{S_s}{L_{PS}} (p_S^c - p_P^c) - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_t \frac{S_t}{L_{PT}} (p_T^c - p_P^c) - \frac{1}{\rho} \overline{\left(\frac{\mathbf{V}}{\tilde{a}} \right)}_b \frac{S_b}{L_{PB}} (p_B^c - p_P^c) \\ & + \sum_{cf=ewnstb} \left[\overline{(q^*)}_{cf} + q_{cf}^{RC} + q_{cf}^{no} \right] = 0 \end{aligned} \quad (4.67)$$

where:

$$\overline{(q^*)}_{cf} = \overline{(\mathbf{u}_{cf}^*)} S_{cf,x} + \overline{(\mathbf{v}_{cf}^*)} S_{cf,y} + \overline{(\mathbf{w}_{cf}^*)} S_{cf,z} \quad \text{interpolated discharge}$$

$$q_{cf}^{RC} = \frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_{cf} \left[\overline{\left(\frac{\partial p^*}{\partial n} \right)}_{cf} - \frac{p_{nb}^* - p_P^*}{L_{nbP}} \right] S_{cf} \quad \text{correction due to cell - face interpolation}$$

$$q_{cf}^{no} = -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_{cf} \left[\overline{\left(\frac{\partial p^c}{\partial n} \right)}_{cf} - \overline{\left(\frac{\partial p^c}{\partial \xi} \right)}_{cf} \right]^{old} S_{cf} \quad \text{correction due to non - orthogonal terms}$$

After arranging the terms, one has:

$$a_P^p p_P^c + \sum_{nb} (a_{nb}^p p_{nb}^c) = b^p \quad (4.68)$$

where:

$$\begin{aligned} a_E^p &= -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_e \frac{S_e}{L_{PE}}, & a_W^p &= -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_w \frac{S_w}{L_{PW}}, & a_N^p &= -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_n \frac{S_n}{L_{PN}}, \\ a_S^p &= -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_s \frac{S_s}{L_{PS}}, & a_T^p &= -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_t \frac{S_t}{L_{PT}}, & a_B^p &= -\frac{1}{\rho} \left(\frac{\Psi}{\tilde{a}} \right)_w \frac{S_b}{L_{PB}}, \\ a_P^p &= -\sum_{nb} a_{nb}^p, & b^p &= -\sum_{cf} \left[\overline{(q^*)}_{cf} + q_{cf}^{RC} + q_{cf}^{no} \right]. \end{aligned}$$

The pressure gradients encountered in the source terms are computed according to the following relations:

$$\begin{aligned} \overline{\left(\frac{\partial p^*}{\partial n} \right)}_e &= \overline{\left(\frac{\partial p^*}{\partial x} \right)}_e \left(\frac{S_{e,x}}{S_e} \right) + \overline{\left(\frac{\partial p^*}{\partial y} \right)}_e \left(\frac{S_{e,y}}{S_e} \right) + \overline{\left(\frac{\partial p^*}{\partial z} \right)}_e \left(\frac{S_{e,z}}{S_e} \right) \\ \overline{\left(\frac{\partial p^c}{\partial n} \right)}_e &= \overline{\left(\frac{\partial p^c}{\partial x} \right)}_e \left(\frac{S_{e,x}}{S_e} \right) + \overline{\left(\frac{\partial p^c}{\partial y} \right)}_e \left(\frac{S_{e,y}}{S_e} \right) + \overline{\left(\frac{\partial p^c}{\partial z} \right)}_e \left(\frac{S_{e,z}}{S_e} \right) \\ \overline{\left(\frac{\partial p^c}{\partial \xi} \right)}_e &= \overline{\left(\frac{\partial p^c}{\partial x} \right)}_e \left(\frac{\Delta x_{PE}}{L_{PE}} \right) + \overline{\left(\frac{\partial p^c}{\partial y} \right)}_e \left(\frac{\Delta y_{PE}}{L_{PE}} \right) + \overline{\left(\frac{\partial p^c}{\partial z} \right)}_e \left(\frac{\Delta z_{PE}}{L_{PE}} \right) \end{aligned} \quad (4.69)$$

in which the interpolation of the pressure gradients along the Cartesian coordinates is done using Eq. 4.21.

4.4.2 Pressure correction procedure

The source term, b^p , has pressure correction terms contained in the discharge due to non-orthogonality of the cells, q_e^{no} . These terms are evaluated explicitly by a double-step pressure correction procedure as follows:

- Solve Eq. 4.68 for p^c by neglecting the non-orthogonal terms, $q_{cf}^{no} = 0$, and correct the velocities and pressure according to Eqs. 4.55a,b.

- Solve again Eq.4.4 with the non-orthogonal terms now available from the first step and correct once again the velocities and pressure.

4.4.3 Under-relaxation factor and time step

To avoid instability of the computation, it is a common practice to put an under-relaxation factor to the pressure correction, $0 \leq \omega^p \leq 1$, in updating the pressure:

$$p = p^* + \omega^p p^c \quad (4.70)$$

As mentioned in Sect. 4.2, the time step plays also as an under-relaxation factor for steady flow cases. This type of application, that is using the transient equations to solve steady flows, is generally known as a pseudo-transient computation. In order to achieve the effects of under-relaxed iterative steady-state computations from a given initial field by means of a pseudo-transient computation starting from the same initial field, the time-step size is taken such that (Fletcher, 1997, p. 365):

$$\omega^p = \frac{1}{1 + E_{\Delta t}} \quad \text{with} \quad E_{\Delta t} = \frac{\tilde{a}_p}{\mathbf{V}_p} \Delta t \quad (4.71)$$

4.5 Boundary conditions

4.5.1 Boundary placement

The boundary conditions that can be considered in the model are inflow, outflow, wall, (water) surface, and symmetry boundaries. The spatial discretisation of the computational domain is done in such a way that the boundaries coincide with the cell face (see Fig. 4.5). The cell neighboring the boundary has special characteristics that modify the definition of cell set forth in Sect. 4.3.1; it has more than one node and less than six neighbors. Three types of cell and node are introduced (see Fig. 4.5):

- *Interior cell* (the white cell) is a computational cell where the dependent variable, ϕ , is unknown and is to be computed at the *interior node* (the solid circle); an interior cell has only one node, the interior node.
- *Boundary cell* (the gray cell) is a boundary neighboring cell whose one or more of its faces coincide with a boundary. A boundary cell has one *interior node* (the solid circle) at the center of the cell and one *boundary node* (the gray circle) at the center of each face that coincides with the boundary. The known boundary values of all variables ϕ are to be defined at the boundary node, either given or extrapolated from the interior nodes.
- *Dummy cell* (hatched cell) and *dummy node* (white node) are used to denote the domain which is excluded from the computation, for example blocked-regions, cylinders, and corners. These dummy cells and nodes are necessary in order to maintain a continuous ordering of the cell and node indexes.

Except of some special cases for the k and ϵ equations, the effect of the boundaries to the computation for the interior node of a boundary cell is additive. In the discretized equation of u , v , w , and p^c , the contribution of each boundary node is added to the source term, b , of the interior node of the boundary cell, and the coefficient related to this boundary node is eventually set to zero. The k and ϵ for the interior nodes of boundary cells having wall or free-surface boundaries, however, are defined by a given expression. In the following sections are presented the method of computation for the boundary cells.

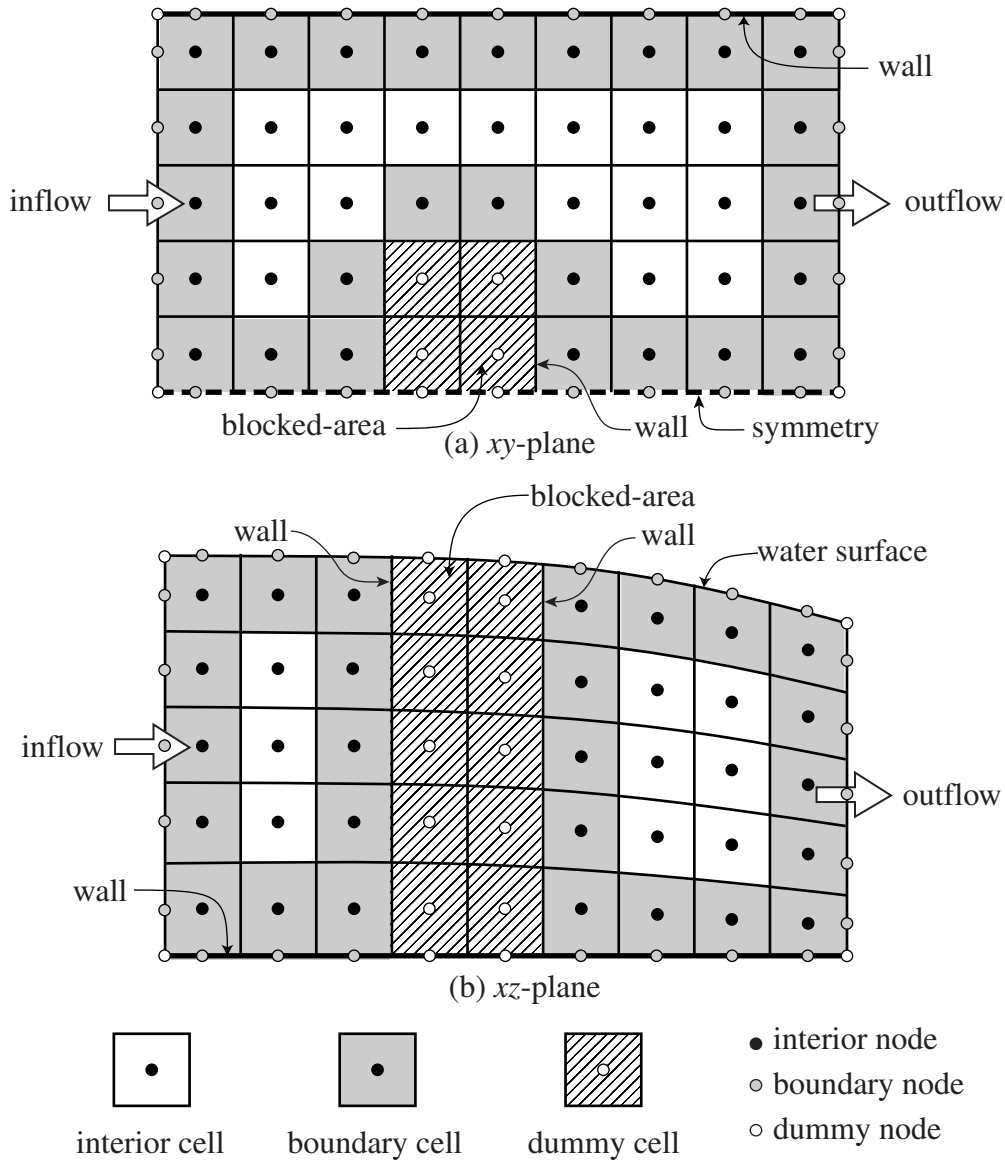


Fig. 4.5 Boundary conditions implemented in the model.

4.5.2 Inflow boundary

Suppose that the inflow boundary lies at the west face, w , of the boundary cell P (see Fig. 4.6). The inflow boundary values across the face w are imposed as the boundary

condition, whose values are defined at the boundary node W located at the same place as w (see Sect. 4.5.1):

$$\phi_W = \phi_{in} \quad (4.72)$$

The pressure is assumed to vary linearly between W, P and E:

$$\frac{p_W - p_P}{L_{PW}} = \frac{p_E - p_P}{L_{PE}} \Rightarrow p_W = (1 + \beta_e) p_P - \beta_e p_E \quad (4.73)$$

The above relation holds as well for p^* and p^c values.

The boundary node denoted by W (instead of w) upon which the inflow boundary values are defined allows the discretized equations formerly established for interior nodes to be applied to the boundary node W without the need to change the notation.

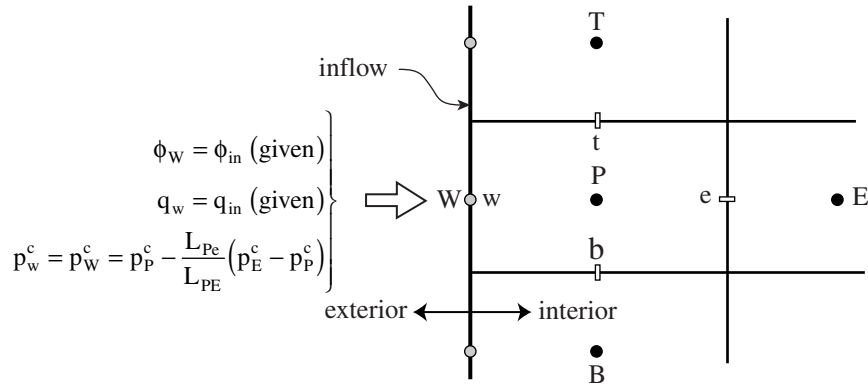


Fig. 4.6 Inflow boundary.

Momentum and k-ε equations

In forming the coefficients in Eq. 4.52, the following steps apply:

- all variables at the inlet are given: $\phi_W = \phi_{in}$
- evaluate the convective-diffusive terms as for normal interior cells:
 $a_W^C, a_W^D, (a_P^C)_W, (a_P^D)_W, (b^D)_W, (b^{1D})_W, (b^{2D})_W$
- bring the contribution of node W to the source term: $b \leftarrow b - (b^D)_W - (a_W^C + a_W^D) \phi_W$
- set the coefficient at node W to zero: $a_W = 0$

Pressure and velocity corrections

The contribution of the discharge across the west face, q_w , to the continuity equation, Eq. 4.66, is replaced by the imposed discharge, q_{in} . In forming the coefficients in Eq. 4.68, the following steps apply:

- set the coefficient at node W to zero: $a_W^p = 0$
- set the contribution of the inflowing discharge to the source term: $(b^p)_w = -q_{in}$

For the velocity correction, Eq. 4.57, the pressure correction gradient, $\partial p^c / \partial x_i$, is computed by the finite-volume technique, Eq. 4.20, which requires the value of $(p^c)_w = (p^c)_W$. This latter is obtained by Eq. 4.73.

4.5.3 Outflow boundary

Suppose that the outflow lies at the east face of the boundary cell P (see Fig. 4.7). Across the outflow face, the convective flux is computed according to the upwinding principle of Eq. 4.28:

$$(F^C)_e = q_e^\ell \phi_P^{\ell+1} = q_{out}^\ell \phi_P^{\ell+1}$$

while the diffusion flux is set to zero:

$$(F^D)_e = 0$$

leading to a simplified form of Eq. 4.43:

$$F_e = q_{out}^\ell \phi_P^{\ell+1}$$

where q_{out}^ℓ is either imposed or computed from the upwinding of P value at the former iteration step:

$$q_{out}^\ell = \vec{V}_{out}^\ell \cdot \vec{S}_e = \vec{V}_P^\ell \cdot \vec{S}_e$$

The same upwinding process finds the other variables at the boundary node E.

$$\phi_E = \phi_e = \phi_P$$

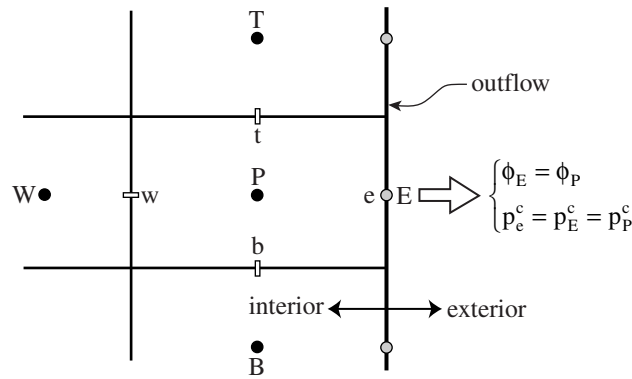


Fig. 4.7 Outflow boundary.

Momentum and k - ε equations

- set all coefficients related to node E in Eq. 4.52 to zero:

$$a_E^C = a_E^D = (a_P^C)_E = (a_P^D)_E = (b^D)_E = (b^{1D})_E = (b^{2D})_E = 0$$
- extrapolate all variables at P to E: $\phi_E = \phi_P$
- the outflowing discharge results: $q_{out}^\ell = \vec{V}_e^\ell \cdot \vec{S}_e = \vec{V}_P^\ell \cdot \vec{S}_e$

Pressure and velocity corrections

The contribution of the discharge across the east face, q_e , to the continuity equation, Eq. 4.66, is replaced by the outflowing discharge, q_{out} . In forming the coefficients in Eq. 4.68, the following steps are done:

- set the coefficient at node E to zero: $a_E^p = 0$
- set the contribution of the inflowing discharge to the source term: $(b^p)_e = -q_{out}$

For the velocity correction, Eq. 4.57, the pressure correction gradient, $\partial p^c / \partial x_i$, is computed by the finite-volume technique, Eq. 4.20, which requires the value of $(p^c)_e = (p^c)_E$; this latter is obtained by the upwinding: $(p^c)_E = (p^c)_P$.

4.5.4 Wall boundary

Wall function approach

The wall function approach (*Launder and Spalding, 1974*) is applied to the cell whose face is a rigid wall. Major assumptions used in this approach merit to be put forward before presenting the derivation of the wall function; they are: (1) the no-slip flow condition prevails at the wall with the universal logarithmic velocity distribution normal

to the wall, (2) the production of the turbulent kinetic energy is merely due to the (turbulent) shear stress, thus neglecting the effect of the normal stress, and (3) a local energy balance exists, i.e. the dissipation of the turbulent kinetic energy is equal to the production. Given in the following paragraphs are the derivations of the wall function in which those assumptions are further highlighted.

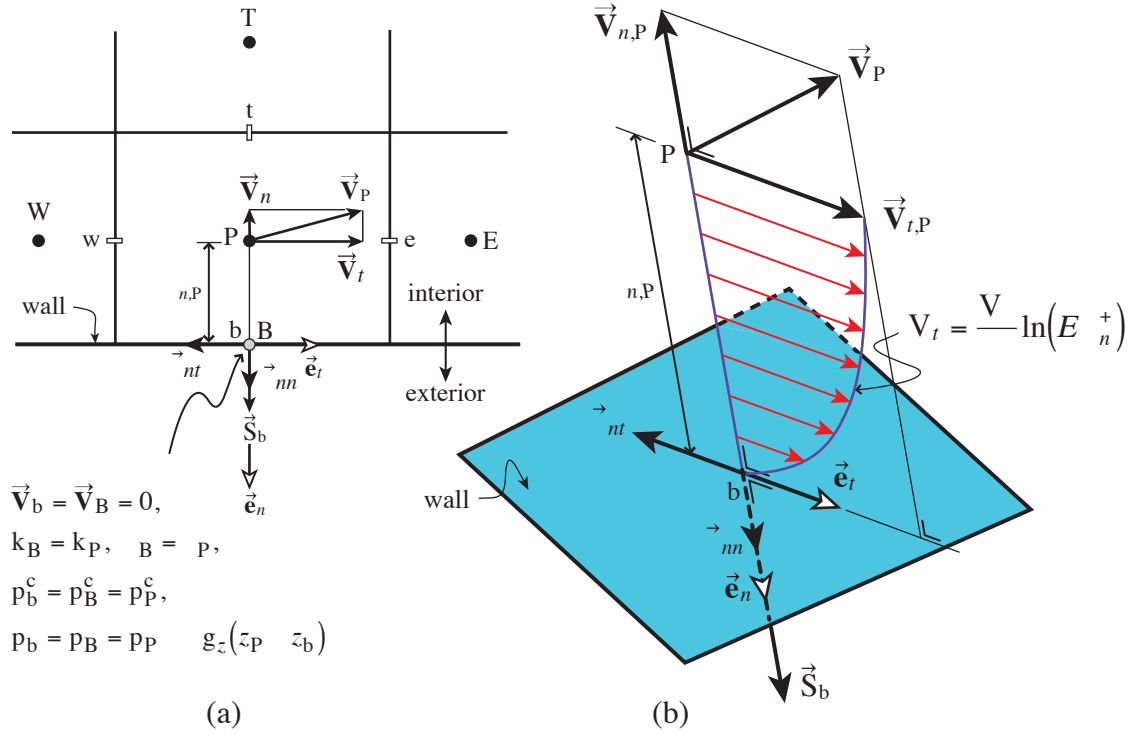


Fig. 4.8 Wall boundary.

- (1) In wall boundaries, the center of the cells is located sufficiently close to the wall but outside the viscous sub-layer (see Fig. 4.8a); the universal logarithmic velocity distribution then prevails in the region \overline{Pb} (see Fig. 4.8b):

$$V_t = \frac{V_*}{\kappa} \ln(E \delta_n^+) \quad \text{or} \quad V_* = \frac{\kappa V_t}{\ln(E \delta_n^+)}, \quad \text{with} \quad V_t = \|\vec{V}_t\| \quad \text{and} \quad \delta_n^+ = \frac{V_* \delta_n}{\nu} \quad (4.74)$$

in which V_* is the shear velocity, V_t is the velocity component parallel to the wall, κ is the Karman universal constant, δ_n is the normal distance from the wall, ν is the molecular viscosity of water, and \mathcal{E} is the wall roughness coefficient. The coefficient \mathcal{E} in the above relation accounts for all flow regimes, either hydraulically smooth, rough, or transition. Note the directions of stresses on the wall face b ; the shear stress, $\vec{\tau}_{nt}$, is to the opposite direction of \vec{e}_t , whereas the normal stress, $\vec{\tau}_{nn}$, is according to \vec{e}_n . These directions are consistent with the convention that shear forces are in the direction of positive increases of velocity (positive velocity gradients). In the direction of \vec{e}_n of the

(local) wall coordinate, (\vec{e}_t, \vec{e}_n) , the velocity gradient $\partial V_t / \partial n$ is negative, whereas $\partial V_n / \partial n$ is positive.

- (2) The equation of the turbulent kinetic-energy production, Eq. 4.12, written in the wall coordinate system, (\vec{e}_t, \vec{e}_n) , shown in Fig. 4.8b, is:

$$G = \nu_t \left[2 \left(\frac{\partial V_t}{\partial t} \right)^2 + \left(\frac{\partial V_n}{\partial t} + \frac{\partial V_t}{\partial n} \right)^2 + 2 \left(\frac{\partial V_n}{\partial n} \right)^2 \right] \quad (4.12a)$$

Since the velocity is zero everywhere along the wall, $V_t = 0$ (no-slip condition) and $V_n = 0$ (no-flow across the wall), all velocity gradients along the wall (the tangential components) disappear. The above expression thus reduces into:

$$G = \nu_t \left(\frac{\partial V_t}{\partial n} \right)^2 + \nu_t 2 \left(\frac{\partial V_n}{\partial n} \right)^2 \quad (4.75a)$$

The first and the second terms depict the turbulent kinetic-energy production due to the shear and normal stresses, respectively. In the wall function, the second term is neglected, which implies that the normal velocity component cannot develop in the wall region. This yields the following:

$$G = \nu_t \left(\frac{\partial V_t}{\partial n} \right)^2 \quad (4.75b)$$

The validity of the above equation is strictly limited at the wall b, but is generally extended to the cell center P where it constitutes an approximation. The omission of the normal velocity gradient in Eq. 4.75a and the extension of Eq. 4.75b to the cell center are, of course, a rather rude approximation, notably in the case of flow around a cylinder. Measurement data in front of the cylinder (see Chapters 2 and 3) show that the radial velocity (the normal component) and the downward velocity (the tangential component) have the same order of magnitude. The normal velocity gradient, therefore, should be accounted for in the turbulent kinetic-energy production. However, the present model adopts Eq. 4.75b since it leads to a numerical simplification.

- (3) The third assumption in the wall function is the existence of a local balance between the turbulent kinetic-energy dissipation and its production, $\varepsilon = G$ (see *Launder and Spalding, 1974; Versteeg and Malalasekera, 1995, p. 73*). This yields:

$$\varepsilon_P = G_P = \left[\nu_t \left(\frac{\partial V_t}{\partial n} \right)^2 \right]_P \quad (4.76)$$

The eddy viscosity can be found from the Boussinesq concept, Eq. 4.5, and the definition of the friction velocity, $\tau_{m,b} = \rho V_*^2$. Considering that the variation of the shear stress is

negligible in the wall region, $\tau_{nt,b} = \tau_{nt,P}$, and that the velocity gradient along the wall is negligible, $(\partial V_n / \partial t)_P \approx 0$, one may write:

$$V_*^2 = \frac{\tau_{nt,b}}{\rho} = \frac{\tau_{nt,P}}{\rho} = \left[v_t \left(\frac{\partial V_t}{\partial n} + \frac{\partial V_n}{\partial t} \right) \right]_P = \left(v_t \frac{\partial V_t}{\partial n} \right)_P \quad (4.77)$$

The velocity gradient is obtainable from the logarithmic velocity distribution, Eq. 4.74:

$$\left(\frac{\partial V_t}{\partial n} \right)_P = \frac{V_*}{\kappa \delta_{n,P}} \quad (4.78)$$

Inserting this relation into Eq. 4.77, one obtains:

$$v_{t,P} = V_* \kappa \delta_{n,P} \quad (4.79)$$

Substituting Eqs. 4.78 and 4.79 into the right-hand-side of Eq. 4.76 yields:

$$\varepsilon_P = G_P = V_* \kappa \delta_{n,P} \left(\frac{V_*}{\kappa \delta_{n,P}} \right)^2 = \frac{V_*^3}{\kappa \delta_{n,P}} \quad (4.80)$$

Combining Eqs. 4.79 and 4.80 to the definition of the eddy viscosity in the k- ε model, Eq. 4.9, one gets:

$$\begin{aligned} v_{t,P} &= c_\mu \frac{k_P^2}{\varepsilon_P} \\ V_* \kappa \delta_n &= c_\mu \left(\frac{k_P^2 \kappa \delta_n}{V_*^3} \right) \\ V_* &= c_\mu^{1/4} k_P^{1/2} \end{aligned} \quad (4.81)$$

There are now two expressions of the friction velocity, i.e. Eqs. 4.74 and 4.81. Both relations are used to evaluate the shear stress at the wall:

$$\vec{\tau}_{nt,b} = -\rho V_* V_* = -\rho c_\mu^{1/4} k_P^{1/2} \frac{\kappa \vec{V}_{t,P}}{\ln(E \delta_{n,P}^+)} \vec{e}_t \quad (4.82)$$

The negative sign is required since $\vec{\tau}_{nt}$ is acting to the opposite direction of \vec{e}_t (see Fig. 4.8). This relation is the one necessary to evaluate the contribution of the wall boundary to the flow momentum equation; its implementation will be further presented later.

Wall function: the final equations

The link between the wall function and the k equation is achieved through the turbulent kinetic-energy production, Eq. 4.76, the velocity gradient, Eq. 4.78, and the shear velocity, Eq. 4.81. Combining these equations, one gets:

$$G_P = \left[\nu_t \left(\frac{\partial V_t}{\partial n} \right)^2 \right]_P = \frac{\tau_{nt,P}}{\rho} \left(\frac{\partial V_t}{\partial n} \right)_P = \frac{\tau_{nt,b}}{\rho} \left(\frac{V_*}{\kappa \delta_{n,P}} \right) = \frac{\tau_{nt,b}}{\rho} \frac{c_\mu^{1/4} k_P^{1/2}}{\kappa \delta_{n,P}} \quad (4.83)$$

with $\tau_{nt,b} = \|\vec{\tau}_{nt,b}\|$.

This equation is used to define the energy production in the source term of the k equation (see Table 4.2) for cells neighboring the wall.

For the ε equation, the energy dissipation is obtained from Eqs. 4.80 and 4.81:

$$\varepsilon_P = \frac{c_\mu^{3/4} k_P^{3/2}}{\kappa \delta_{n,P}} \quad (4.84)$$

Before detailing the implementation of the wall function to the discretized momentum and k- ε transport equations, two variables need to be defined, namely the wall roughness coefficient, \mathcal{E} , and the parallel velocity components, \vec{V}_t .

Wall roughness coefficient. The wall roughness coefficient, \mathcal{E} , in the logarithmic velocity profile is adjusted according to the equivalent (standard) roughness, k_s , whether it is hydraulically smooth, rough, or transition between smooth and rough. The following relation is used to define the roughness coefficient (*Wu et al., 2000*):

$$\mathcal{E} = \exp[\kappa(\mathcal{B} - \Delta\mathcal{B})] \quad (4.85)$$

where \mathcal{B} is an additive constant and $\Delta\mathcal{B}$ is a roughness function determined according to the standard roughness, k_s , as follows (*Cebeci and Bradshaw, 1977*):

$$\Delta\mathcal{B} = \begin{cases} 0 & \text{for } k_s^+ < 2.25 \\ \left[\mathcal{B} - 8.5 + \frac{1}{\kappa} \ln k_s^+ \right] \sin \left[0.4258 (\ln k_s^+ - 0.811) \right] & \text{for } 2.25 \leq k_s^+ < 90 \\ \mathcal{B} - 8.5 + \frac{1}{\kappa} \ln k_s^+ & \text{for } k_s^+ \geq 90 \end{cases} \quad (4.86)$$

with $\mathcal{B} = 5.2$, $\kappa = 0.4$, and $k_s^+ = V_* k_s / \nu$ being the roughness Reynolds number.

Tangential velocity component. The velocity at the cell center P needs to be decomposed into its normal and tangential components with respect to the wall (see Fig.

4.8b). The unit vector normal to the wall, \vec{e}_n , has an outward direction (see Eq. 4.17), while the unit vector tangential to the wall, \vec{e}_t , has the direction of the projection of \vec{V}_P on the wall. Both unit vectors are perpendicular in such a way that any vector (for example \vec{V}_P) can be decomposed along them in the plane (\vec{e}_t, \vec{e}_n) which also contains \vec{V}_P . For determining \vec{e}_t , it is necessary to find the projection $\vec{V}_{t,P}$, which can be obtained from:

$$\vec{V}_{t,P} = \vec{V}_P - \vec{V}_{n,P} = \vec{V}_P - (\vec{V}_P \cdot \vec{e}_n) \vec{e}_n \quad (4.87)$$

Knowing $\vec{V}_{t,P}$ the unit vector \vec{e}_t can be easily computed:

$$\vec{e}_t = \left(\vec{V}_t / \|\vec{V}_t\| \right)_P \quad (4.88)$$

Implementation of the wall function

Wall function for the momentum equation. Since there is no discharge across the wall, the convective flux does not exist across the wall face; the only flux is due to the diffusion. In the discretized momentum equation, Eq. 4.52, the diffusion term is not evaluated by Eq. 4.33, but by evaluating this term as a normal force (per unit mass) acting on the wall. Similarly the wall shear-stress, Eq. 4.82, is also transformed as a shear force. Both forces are considered as a source term and are evaluated at iteration ℓ , which then linearized such that the velocities at the cell center P become unknown variables.

The force due to the normal stress acting on the wall (see Fig. 4.8) can be computed as:

$$\left(\frac{\vec{F}_n}{\rho} \right)_b^\ell = \left(\frac{\vec{\tau}_{nn}}{\rho} \right)_b^\ell S_b = \left(-v_t 2 \frac{\partial V_n}{\partial n} \vec{e}_n \right)_b^\ell S_b \approx \left(-v_t 2 \frac{\partial V_n}{\partial n} \vec{e}_n \right)_P^\ell S_b = \left(-v_t 2 \frac{V_n}{\delta_n} \vec{e}_n \right)_P^\ell S_b \quad (4.89)$$

The negative sign is necessary since the normal stress, $\vec{\tau}_{nn}$, is in the negative direction of \vec{V}_n (see Fig. 4.8). All terms are evaluated explicitly, that is the velocity is from the ℓ^{th} iteration, the kinetic energy is from the m^{th} iteration, and the geometry is from the n^{th} time iteration.

The force due to the shear stress acting on the wall is obtained from Eq. 4.82, but its form is modified to allow easy computation of the turbulent viscosity later on.

$$\left(\frac{\vec{F}_t}{\rho} \right)_b^\ell = \left(\frac{\vec{\tau}_{nt}}{\rho} \right)_b^\ell S_b = - \underbrace{\left[\frac{c_\mu^{1/4} (k^m)^{1/2} \kappa \delta_n}{\ln(E \delta_n^+)} \right]_P}^{(1)} \underbrace{\left(\frac{V_t}{\delta_n} \vec{e}_t \right)_P^\ell}_{(2)} S_b$$

The terms grouped in the first bracket on the right-hand side of the above expression have together the dimension of a viscosity, thus can be considered as the *wall turbulent-viscosity*, $\nu_{t,\text{wall}}$:

$$\left(\frac{\bar{\mathbf{F}}_t}{\rho}\right)_b^\ell = -\nu_{t,\text{wall}} \left(\frac{\bar{\mathbf{V}}_t}{\delta_n}\right)_P^\ell S_b, \quad \text{with} \quad \nu_{t,\text{wall}} = \left[\frac{c_\mu^{1/4} (k^m)^{1/2} \kappa \delta_n}{\ln(E \delta_n^+)} \right]_P \quad (4.90)$$

Both forces are added to the source term of the boundary cell P, as the contribution from the wall boundary node B. The other coefficients related to the contribution from the boundary node B are then assigned to zero. The following steps are used in evaluating the coefficients in Eq. 4.52:

- set all coefficients related to the contribution of the boundary node B to zero:
 $a_B^C = a_B^D = 0$, $(a_P^C)_B = (a_P^D)_B = (b^D)_B = 0$, and $(b^{2D})_B = 0$
- compute $\bar{\mathbf{e}}_t$ using Eqs. 4.87 and 4.88
- compute the forces due to the normal and shear stresses as source terms and linearise the source:

x-momentum:

$$\begin{aligned} (b + b_P u_P^{\ell+1})_B &= \left(\frac{F_{n,x}}{\rho} + \frac{F_{t,x}}{\rho}\right)_b^\ell \\ &= \underbrace{\left[\nu_t (u e_{n,x} + 2v e_{n,y} + 2w e_{n,z}) e_{n,x} + \nu_{t,\text{wall}} (v e_{t,y} + w e_{t,z}) e_{t,x} \right]_P^\ell \frac{S_b}{\delta_{n,P}}}_{(b)_B} \\ &\quad - \underbrace{\left[\nu_t (e_{n,x} e_{n,x}) + \nu_{t,\text{wall}} (e_{t,x} e_{t,x}) \right]_P^\ell \frac{S_b}{\delta_{n,P}}}_{(b_P)_B} u_P^{\ell+1} \end{aligned} \quad (4.91a)$$

y-momentum:

$$\begin{aligned} (b + b_P u_P^{\ell+1})_B &= \left(\frac{F_{n,y}}{\rho} + \frac{F_{t,y}}{\rho}\right)_b^\ell \\ &= \underbrace{\left[\nu_t (2u e_{n,x} + v e_{n,y} + 2w e_{n,z}) e_{n,y} + \nu_{t,\text{wall}} (u e_{t,x} + w e_{t,z}) e_{t,y} \right]_P^\ell \frac{S_b}{\delta_{n,P}}}_{(b)_B} \\ &\quad - \underbrace{\left[\nu_t (e_{n,y} e_{n,y}) + \nu_{t,\text{wall}} (e_{t,y} e_{t,y}) \right]_P^\ell \frac{S_b}{\delta_{n,P}}}_{(b_P)_B} v_P^{\ell+1} \end{aligned} \quad (4.91b)$$

z-momentum:

$$\begin{aligned}
 (\mathbf{b} + \mathbf{b}_P \mathbf{u}_P^{\ell+1})_B &= \left(\frac{F_{n,z}}{\rho} + \frac{F_{t,z}}{\rho} \right)_b^\ell \\
 &= \underbrace{\left[\mathbf{v}_t \left(2u \mathbf{e}_{n,x} + 2v \mathbf{e}_{n,y} + w \mathbf{e}_{n,z} \right) \mathbf{e}_{n,z} + \mathbf{v}_{t,\text{wall}} \left(u \mathbf{e}_{t,x} + v \mathbf{e}_{t,y} \right) \mathbf{e}_{t,z} \right]_P^\ell \frac{S_b}{\delta_{n,P}}}_{(\mathbf{b})_B} \quad (4.91c) \\
 &\quad - \underbrace{\left[\mathbf{v}_t \left(\mathbf{e}_{n,z} \mathbf{e}_{n,z} \right) + \mathbf{v}_{t,\text{wall}} \left(\mathbf{e}_{t,z} \mathbf{e}_{t,z} \right) \right]_P^\ell \frac{S_b}{\delta_{n,P}} \mathbf{w}_P^{\ell+1}}_{(\mathbf{b}_P)_B}
 \end{aligned}$$

Wall function for the k equation. The wall function is used to evaluate the source term of the k equation, $G - \varepsilon$. The turbulent kinetic-energy production, G , for cells neighboring the wall is not evaluated by Eq. 4.12, but is directly obtained from Eq. 4.83. Since the velocity has been known when solving the k equation, this information can be used in calculating the wall shear-stress term in the G equation, Eq. 4.83. The turbulent kinetic-energy dissipation, ε , is evaluated by Eq. 4.84; this contains a non-linear term in k which is then linearized. The following steps apply:

- set all coefficients related to the contribution from the boundary node B to zero:
 $\mathbf{a}_B^C = \mathbf{a}_B^D = 0$, $(\mathbf{a}_P^C)_B = (\mathbf{a}_P^D)_B = (\mathbf{b}^D)_B = 0$
- compute the source, that is the energy production and dissipation, and linearise the source:

$$\begin{aligned}
 \mathbf{b} + \mathbf{b}_P k_P^{m+1} &= (G_P - \varepsilon_P) \mathbf{V}_P \\
 &= \underbrace{\left(\frac{\tau_{nt}}{\rho} \right)_b^m \frac{c_\mu^{1/4} (k_P^m)^{1/2}}{\kappa \delta_{n,P}} \mathbf{V}_P}_{\mathbf{b}} - \underbrace{\frac{c_\mu^{3/4} (k_P^m)^{1/2}}{\kappa \delta_{n,P}} \mathbf{V}_P}_{\mathbf{b}_P} k_P^{m+1} \quad (4.92)
 \end{aligned}$$

where the magnitude of the shear stress is evaluated with the velocity already computed from the momentum equations (see Eq. 4.82):

$$\begin{aligned}
 \left(\frac{\tau_{nt}}{\rho} \right)_b^m &= \frac{1}{\rho} \left\| \tilde{\boldsymbol{\tau}}_{nt,b} \right\| = \left[\frac{c_\mu^{1/4} (k_P^m)^{1/2}}{\ln(E \delta_n^+)} \right]_P \left\| \vec{\mathbf{V}}_{t,P}^{m+1} \right\| = \left(\frac{\mathbf{v}_{t,\text{wall}}}{\delta_n} \right)_P \left\| \vec{\mathbf{V}}_P^{m+1} \cdot \vec{\mathbf{e}}_t \right\| \\
 &= \left(\frac{\mathbf{v}_{t,\text{wall}}}{\delta_n} \right)_P^m \left| \left(u_P^{m+1} \mathbf{e}_{t,x} + v_P^{m+1} \mathbf{e}_{t,y} + w_P^{m+1} \mathbf{e}_{t,z} \right) \right|
 \end{aligned}$$

Note that k_P^{m+1} is unknown for this computation step, while u_P^{m+1} , v_P^{m+1} , and w_P^{m+1} are already fixed.

Wall function for the ϵ equation. The turbulent kinetic-energy dissipation for cells neighboring the wall is defined by Eq. 4.84. This can be easily implemented as follows:

- set all coefficients related to the neighboring cells to zero:
 $a_E = a_W = a_N = a_S = a_T = a_B = 0$
- set the coefficient at cell P to unity: $a_P = 1$
- set the source terms by (see Eq. 4.84):

$$b = \epsilon_P^{m+1} = \frac{c_\mu^{3/4} (k_P^{m+1})^{3/2}}{\kappa \delta_{n,P}} \quad (4.93)$$

where k_P^{m+1} is already computed in the previous step

Wall function and pressure correction. Since the discharge across the wall is zero, the coefficient of the boundary node B in Eq. 4.68 is set to zero: $a_B^p = 0$. The pressure correction at the boundary node is obtained by direct extrapolation from the cell center P:
 $p_B^c = p_P^c$.

4.5.5 Symmetry boundary

At the symmetry plane, for example at the east face (see Fig. 4.9), the convective transport across the plane and the shear stress along the plane are zero. These properties make the velocity at E be easily obtained from the projection of the velocity at P to the plane. For the scalar variables, k and ϵ , an approximation is used by extrapolating the values at P to the boundary E. The following expressions thus apply at symmetry boundaries:

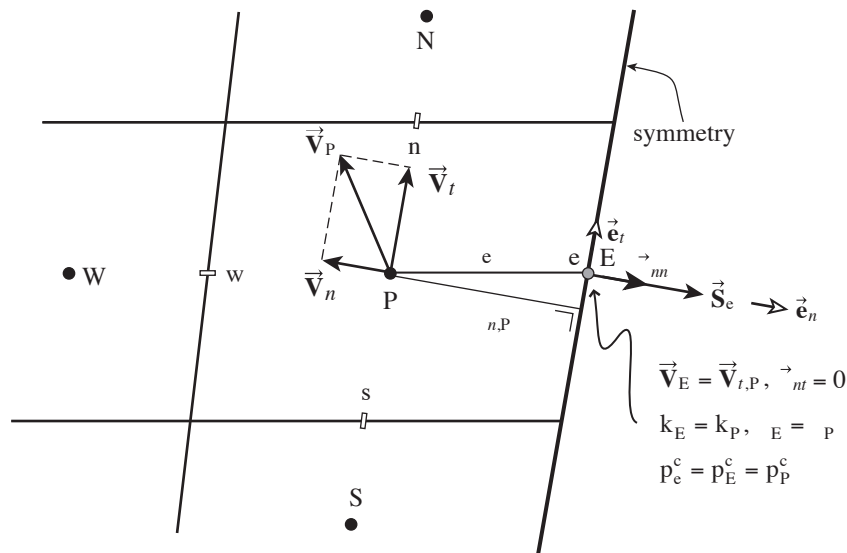


Fig. 4.9 Symmetry boundary.

$$\left. \begin{array}{l} (\mathbf{F}^C)_e = 0 \text{ for all } \phi \\ \tau_{nt,e} = 0 \end{array} \right\} \Rightarrow \vec{\mathbf{V}}_E = \vec{\mathbf{V}}_{t,P}$$

$$\phi_E = \phi_P \text{ for } \phi = k \text{ and } \varepsilon$$

The diffusive term in the momentum equations, which is due to the normal stress, is evaluated with the same method as that for the wall function approach. This term is considered as a force per unit mass acting on the plane.

Momentum equations

- set all coefficients related to the boundary node E in Eq. 4.52 to zero:

$$a_E^C = a_E^D = (a_P^C)_E = (a_P^D)_E = (b^D)_E = (b^{1D})_E = (b^{2D})_E = 0$$

- compute $\bar{\mathbf{e}}_t$ using Eqs. 4.87 and 4.88; its components are $e_{t,x}, e_{t,y}, e_{t,z}$.
- compute the forces due to the normal stress as source terms:

x-momentum:

$$\begin{aligned} (b + b_P u_P^{\ell+1})_E &= \left(\frac{F_{n,x}}{\rho} \right)_e \\ &= \underbrace{\left[v_t (u e_{n,x} + 2v e_{n,y} + 2w e_{n,z}) e_{n,x} \right]_P \frac{S_e}{\delta_n}}_{(b)_E} - \underbrace{\left[v_t (e_{n,x} e_{n,x}) \right]_P \frac{S_e}{\delta_{n,P}}}_{(b_P)_E} u_P^{\ell+1} \end{aligned} \quad (4.94a)$$

y-momentum:

$$\begin{aligned} (b + b_P u_P^{\ell+1})_E &= \left(\frac{F_{n,y}}{\rho} \right)_e \\ &= \underbrace{\left[v_t (2u e_{n,x} + v e_{n,y} + 2w e_{n,z}) e_{n,y} \right]_P \frac{S_e}{\delta_n}}_{(b)_E} - \underbrace{\left[v_t (e_{n,y} e_{n,y}) \right]_P \frac{S_e}{\delta_{n,P}}}_{(b_P)_E} v_P^{\ell+1} \end{aligned} \quad (4.94b)$$

z-momentum:

$$\begin{aligned} (b + b_P u_P^{\ell+1})_E &= \left(\frac{F_{n,z}}{\rho} \right)_e \\ &= \underbrace{\left[v_t (2u e_{n,x} + 2v e_{n,y} + w e_{n,z}) e_{n,z} \right]_P \frac{S_e}{\delta_n}}_{(b)_E} - \underbrace{\left[v_t (e_{n,z} e_{n,z}) \right]_P \frac{S_e}{\delta_{n,P}}}_{(b_P)_E} w_P^{\ell+1} \end{aligned} \quad (4.94c)$$

k and ε equations

- set all coefficients related to the boundary node E in Eq. 4.52 to zero:

$$a_E^C = a_E^D = (a_P^C)_E = (a_P^D)_E = (b^D)_E = (b^{1D})_E = (b^{2D})_E = 0$$
- extrapolate k and ε at P to E: $k_E = k_P$ and $\varepsilon_E = \varepsilon_P$

Pressure and velocity corrections

Since the discharge across the boundary is zero, the coefficients related to the contribution from the boundary node E in the discretized pressure correction equation, Eq. 4.68, are set to zero: $a_E^p = 0$ and $(b^p)_e = 0$.

The pressure correction gradient, $\partial p^c / \partial x_i$, needed for the velocity correction, Eq. 4.57, is computed by the finite-volume technique, Eq. 4.20, which requires the value of $(p^c)_e = (p^c)_E$. This latter is obtained by: $(p^c)_E = (p^c)_P$. The velocity at E has to be corrected such that it is parallel to the symmetry plane, since the flux across the boundary is zero. This is similar to assuming that the velocity at E is the same as the projection of the velocity vector at P on a plane parallel to the symmetry boundary: $\vec{V}_E = \vec{V}_{i,P}$.

4.5.6 Surface boundary

At the (water) surface (see Fig. 4.10) the velocity is parallel to the boundary, the discharge across the (water) surface is zero, and thus there is no convective transport across this boundary. The shear stress along the surface, in addition, is neglected. This allows the specification of the velocity along the surface boundary the same as the projection of the velocity at the cell center. The water surface does not create turbulence; therefore, the kinetic energy along the surface boundary is set to zero. The energy dissipation, ε , at cell center is obtained in a similar manner as that at the wall boundary; a correction may be given to reduce the computed value as has been reported in some previous works (*Krishnappan and Lau, 1986*).

For the pressure, a hydrostatic distribution is assumed between the surface and the cell center. The pressure at the surface is supposed to be atmospheric; if it is not the case, the surface is moved according to the pressure defect, relative to a reference pressure, which is prescribed at a particular cell. This reference cell is normally defined at the top-most cell of the outflow boundary. This is similar to prescribe a constant flow-depth condition at the outflow. The surface correction is done at the end of each time step. An under-relaxation factor and a limitation may be imposed to avoid excessive change of the computational domain. The procedures to handle surface boundary are described in the following paragraphs.

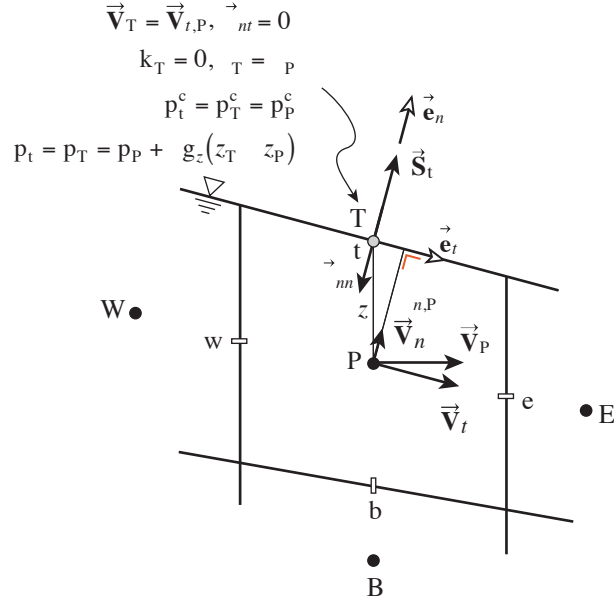


Fig. 4.10 Surface boundary.

Momentum equations

- set all coefficient related to the boundary node T in Eq. 4.52 to zero:
 $a_T^C = a_T^D = (a_P^C)_T = (a^D)_T = (b^D)_T = (b^{2D})_T = 0$
- compute \vec{e}_t using Eqs. 4.87 and 4.88
- compute the forces due to the normal stress as source terms:

x-momentum:

$$\begin{aligned}
 (b + b_P u_P^{\ell+1})_T &= \left(\frac{F_{n,x}}{\rho} \right)_t^\ell \\
 &= \underbrace{\left[v_t (u e_{n,x} + 2v e_{n,y} + 2w e_{n,z}) e_{n,x} \right]_P^\ell \frac{S_t}{\delta_n}}_{(b)_T} - \underbrace{\left[v_t (e_{n,x} e_{n,x}) \right]_P^\ell \frac{S_t}{\delta_{n,P}}}_{(b_P)_T} u_P^{\ell+1} \quad (4.95a)
 \end{aligned}$$

y-momentum:

$$\begin{aligned}
 (b + b_P u_P^{\ell+1})_T &= \left(\frac{F_{n,y}}{\rho} \right)_t^\ell \\
 &= \underbrace{\left[v_t (2u e_{n,x} + v e_{n,y} + 2w e_{n,z}) e_{n,y} \right]_P^\ell \frac{S_t}{\delta_n}}_{(b)_T} - \underbrace{\left[v_t (e_{n,y} e_{n,y}) \right]_P^\ell \frac{S_t}{\delta_{n,P}}}_{(b_P)_T} v_P^{\ell+1} \quad (4.95b)
 \end{aligned}$$

z-momentum:

$$\begin{aligned}
 (\mathbf{b} + \mathbf{b}_P \mathbf{u}_P^{\ell+1})_T &= \left(\frac{\mathbf{F}_{n,z}}{\rho} \right)_t^\ell \\
 &= \underbrace{-\left[\mathbf{v}_t (2u \mathbf{e}_{n,x} + 2v \mathbf{e}_{n,y} + w \mathbf{e}_{n,z}) \mathbf{e}_{n,z} \right]_P^\ell \frac{S_t}{\delta_n}}_{(\mathbf{b})_T} - \underbrace{\left[\mathbf{v}_t (\mathbf{e}_{n,z} \mathbf{e}_{n,z}) \right]_P^\ell \frac{S_t}{\delta_{n,P}}}_{(\mathbf{b}_P)_T} w_P^{\ell+1} \quad (4.95c)
 \end{aligned}$$

k equation

- set all coefficient related to the boundary node T in Eq. 4.52 to zero:
 $a_T^C = a_T^D = (a_P^C)_T = (a^D)_T = (b^D)_T = (b^{2D})_T = 0$
- set the surface kinetic energy at T to zero: $k_T = 0$

ε equation

- set all coefficients related to all neighboring cells to zero:
 $a_E = a_W = a_N = a_S = a_T = a_B = 0$
- set the coefficient at cell P to unity: $a_P = 1$
- set the source terms by (see Eq. 4.84 and also *Krishnappan and Lau, 1986*):

$$\varepsilon_P^{m+1} = c_f \frac{c_\mu^{3/4} (k_P^{m+1})^{3/2}}{\kappa \delta_{n,P}} \quad (4.96)$$

where c_f is an empirical constant, which is set to 0.164

Pressure and velocity corrections

- set the coefficient at T in Eq. 4.68 to zero: $a_T^p = 0$
- extrapolate the pressure correction at P to T: $p_T^c = p_P^c$
- correct the pressure and the velocity
- extrapolate the velocity at P to T and correct this velocity: $\vec{V}_T = \vec{V}_{t,P}$

Surface correction

- extrapolate the pressure at P to T by assuming a hydrostatic distribution:
 $p_T = p_P + \rho g_z \Delta z_{PT}$ (see Fig. 4.11a), with g_z the z -component of \vec{g} (generally negative)
- compute the surface correction based on the pressure-defect relative to the reference pressure: $\Delta h^{(P)} = (p_T - p_{ref}) / (\rho |g_z|)$

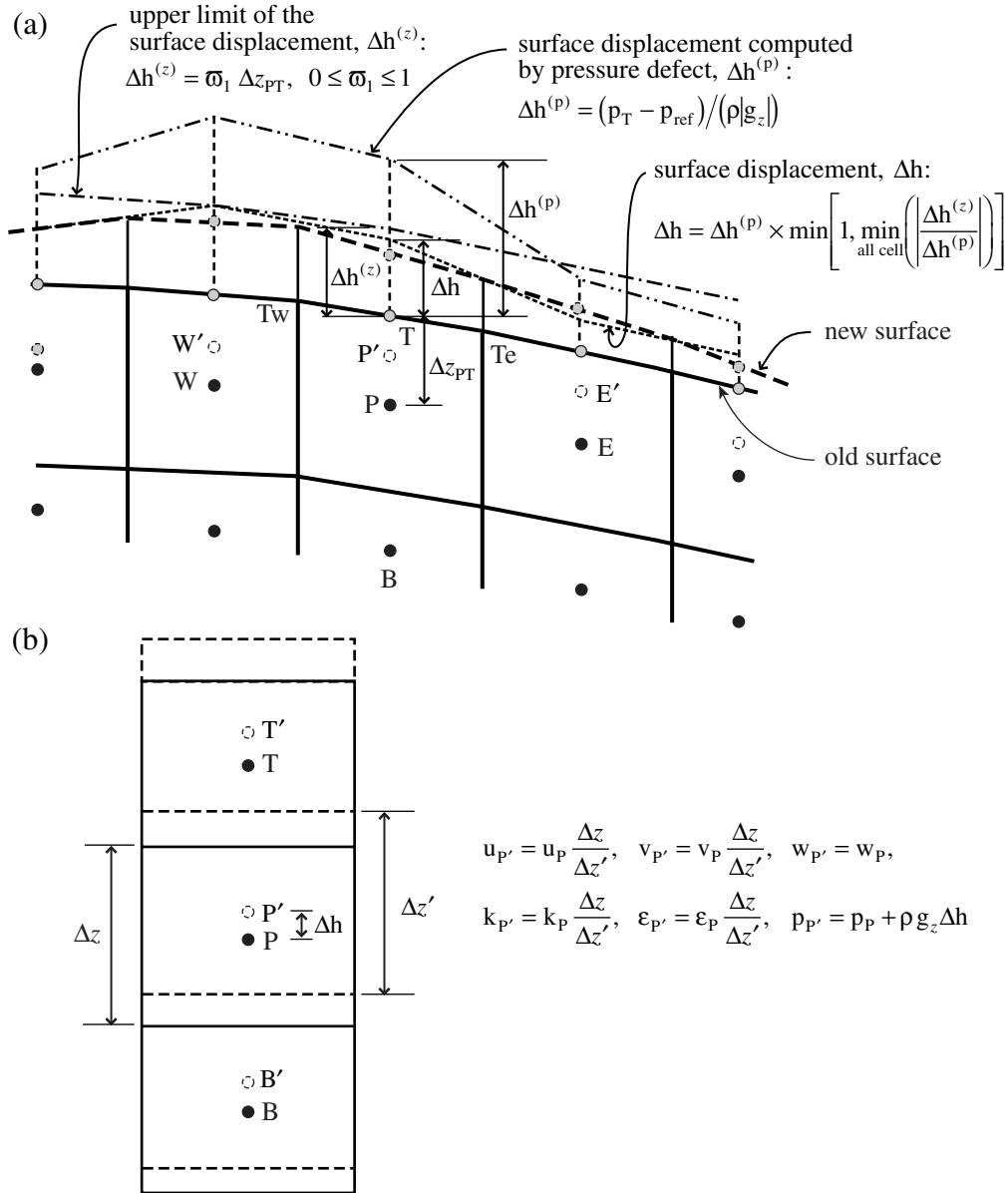


Fig. 4.11 Surface correction: (a) displacement of the vertices of the surface boundary cells, (b) modification of the variables at the new cell centers.

- compute the upper limit of the surface correction based on the cell thickness:
 $\Delta h^{(z)} = \varpi_1 \Delta z_{PT}, 0 \leq \varpi_1 \leq 1$
- compute the displacement of the boundary node T by:

$$\Delta h = \Delta h^{(p)} \times \min \left[1, \min_{\text{all cells}} \left(\frac{\Delta h^{(z)}}{\Delta h^{(p)}} \right) \right]$$
- since a cell is defined by its vertices, Δh at T needs to be distributed over the vertices, T_e, T_w, T_n, T_s (see Fig. 4.11a); linear interpolation is used
- move the cell vertices of the free-surface boundary cells according to the new coordinates of T_e, T_w, T_n, T_s

- reconstruct the mesh, maintain the number of cells in the vertical and their relative positions to the local depth
- use linear interpolation to get variables at the new cell centers (see Fig. 4.11b) in order to maintain the continuity:

$$u_{P'} = u_P \frac{\Delta z}{\Delta z'}, \quad v_{P'} = v_P \frac{\Delta z}{\Delta z'}, \quad w_{P'} = w_P,$$

$$k_{P'} = k_P \frac{\Delta z}{\Delta z'}, \quad \varepsilon_{P'} = \varepsilon_P \frac{\Delta z}{\Delta z'}, \quad p_{P'} = p_P + \rho g_z \Delta h$$

- continue the computation to the next time level

4.6 Solution procedures

4.6.1 Spatial discretisation

Applying the discretized governing equations, Eqs. 4.52 and 4.68 requires spatial discretisation of the computational domain. The domain is divided into N_i-2 , N_j-2 , and N_k-2 cells in the x -, y -, z -directions, respectively, from which there are N_i , N_j , and N_k nodes (interior, boundary, and dummy nodes) in the corresponding directions. A typical spatial discretisation is shown in Fig. 4.12. The cell and node are denoted by a single index; the cells are indexed by $ijk = 2,3,\dots,N_{ijkm}$, going along the y -direction ($j = 2,3,\dots,N_j-1$), the x -direction ($i = 2,3,\dots,N_i-1$), and the z -direction ($k = 2,3,\dots,N_k-1$), while the nodes are indexed by $ijk = 1,2,\dots,N_{ijk}$. With this indexing, the single-index ijk for a cell and for its six-neighbors can be easily obtained from their position in the (x,y,z) space (see Table 4.5).

Writing Eq. 4.52 or 4.68 for all cells, $ijk = 2,3,\dots,N_{ijkm}$, produces a series of algebraic linear equations which can be presented in a matrix form as follows:

$$\mathbf{A} \Phi = \mathbf{B} \tag{4.97}$$

The matrix \mathbf{A} contains the coefficients of the equations, a_{nb} and a_p , Φ is a column matrix of the dependent variable, and \mathbf{B} is a column matrix of the source terms. The matrix \mathbf{A} has diagonal blocks which are themselves tridiagonal, and sub- and super-diagonal blocks in which each block has two diagonals, thus it has only 7 non-zero diagonals while the other elements are zero. It is then not necessary to store all elements of the matrix \mathbf{A} ; only those seven diagonals need to be stored. To facilitate the storage, each non-zero diagonal is stored in a separate column matrix, i.e. \mathbf{A}_{nb} , $nb = EWNSTB$, and \mathbf{A}_p . Fig. 4.13 shows the form of the matrix \mathbf{A} .

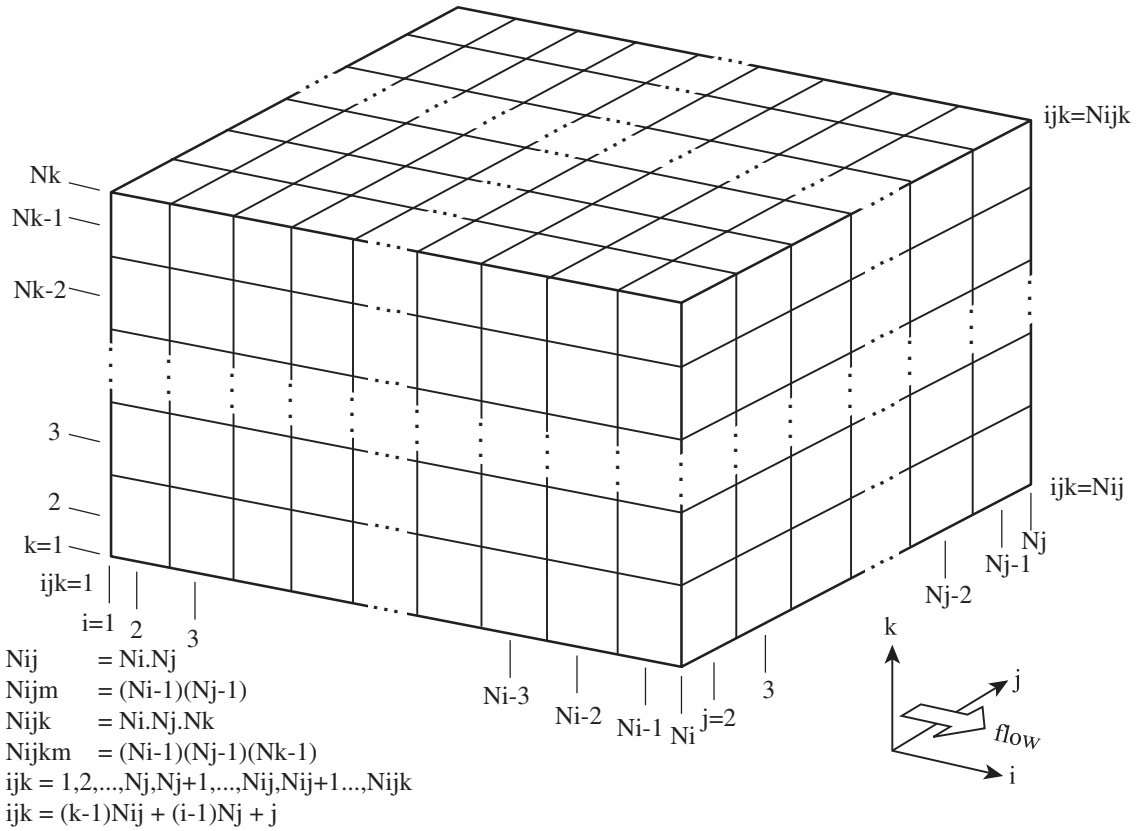


Fig. 4.12 Typical spatial discretisation of the computational domain.

Table 4.5 Triple and single indexing of computational cells.

Direction	Nodal index	Cell index
<i>x</i> -direction	$i = 1, 2, \dots, N_i$	$i = 2, 3, \dots, N_i - 1$
<i>y</i> -direction	$j = 1, 2, \dots, N_j$	$j = 2, 3, \dots, N_j - 1$
<i>z</i> -direction	$k = 1, 2, \dots, N_k$	$k = 2, 3, \dots, N_k - 1$
Cells in the <i>xy</i> -plane	$ij = 1, 2, \dots, N_{ij}$ ($N_{ij} = N_i \times N_j$)	$ij = 2, 3, \dots, N_{ijm}$ [$N_{ijm} = (N_i - 1) \times (N_j - 1)$]
Cells in the domain	$ijk = 1, 2, \dots, N_{ijk}$ ($N_{ijk} = N_{ij} \times N_k$)	$ijk = 2, 3, \dots, N_{ijkm}$ [$N_{ijkm} = N_{ijm} \times (N_k - 1)$]
Cell	Triple-index	Single-index
P	i, j, k	$ijk = (k - 1)N_{ij} + (i - 1)N_j + j$
E	$i + 1, j, k$	$ijk + N_j$
W	$i - 1, j, k$	$ijk - N_j$
N	$i, j + 1, k$	$ijk + 1$
S	$i, j - 1, k$	$ijk - 1$
T	$i, j, k + 1$	$ijk + N_{ij}$
B	$i, j, k - 1$	$ijk - N_{ij}$

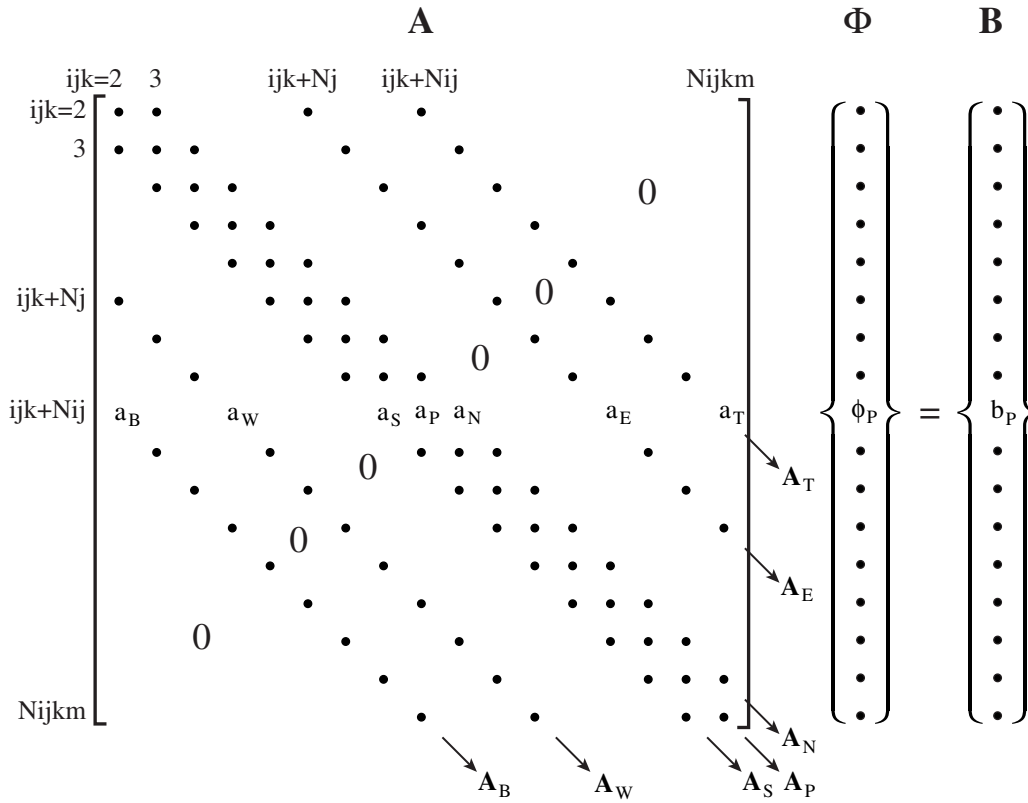


Fig. 4.13 The structure of the matrix coefficient \mathbf{A} ; all elements not shown are zero.

4.6.2 Matrix solvers

The solution of Eq. 4.97 is achieved by iteration techniques. Two matrix-solver routines are available in this model, i.e. the SIPSOL, for the **Strongly-Implicit Procedure** method (Stone, 1968), and the CGSTAB, for the **Conjugate Gradient Stabilized** method of Van den Vorst (see Ferziger and Peric, 1997, pp. 105-106). For the first solver, the SIPSOL, the solution procedure follows the one proposed by Jesshope (Jesshope, 1979) with some modification to suit for the 7-point computational-cells of the three-dimensional cases. The CGSTAB routine is taken from the reference (Ferziger and Peric, 1997, pp. 105-106) without any major modification. Without going into detail, the iterative procedure of the routines can be summarized as follows:

- given an initial solution, define a residue, that is the difference between the left- and right-hand side of Eq. 4.97,
- calculate an increment based on the initial solution and residue,
- update the solution by adding the increment to the initial solution,
- repeat the procedure until convergence.

The iterative procedure to get the solution of Eq. 4.97 is commonly denoted as *inner-iteration*. This is to be distinguished from the *outer-iteration*, that is the one necessary to seek the solution satisfying the momentum and continuity equations (the ℓ -iteration) or all the governing equations, the momentum, continuity, and k- ϵ equations (the m-

iteration). The inner-iteration seeks the solution for each dependent variable, ϕ , across the computational domain for given coefficient and source terms, \mathbf{A} and \mathbf{B} , which are constants. The solution for every dependent variable, ϕ ($\phi = u, v, w, p, k, \varepsilon$), is sought one after another. In the outer-iteration, solution is obtained for every dependent variable that all together satisfies the governing equations. In each outer-iteration, the coefficient and source terms are adjusted, where as in the inner-iteration, they are kept constant.

When performing the inner-iteration, it is important to decide when to quit the solver. Since the solution of Eq. 4.97 for a particular variable, u for example, at a particular iteration level, ℓ , does not necessarily satisfy all governing equations for $u, v, w, p, k, \varepsilon$, it is inefficient to carry out a rigorous iteration at this stage. A restricted number of iterations and a moderate convergence criterion will do. A single or at most a two inner-iteration is generally sufficient to solve the momentum equation for u, v , and w since their equations are of convection types. The pressure correction however requires a number of sweeps over the entire domain to have a solution error within a sufficiently small allowable limit. The convergence of the k and ε equations, being of convection-diffusion types, may also be slow. This is due to the need of a small relaxation factor in the iteration process to avoid oscillations. The residue of the solution of Eq. 4.97 is used as the basis to detect the solution error; it is defined as:

$$\mathfrak{R}_\phi = \sum_{\text{all cells}} \left| b - a_p \phi_p - \sum a_{nb} \phi_{nb} \right| \quad (4.98)$$

The calculation in the inner-iteration is stopped when either one of the following criteria is satisfied:

$$\mathfrak{R}_\phi^{mi} \leq \lambda_1 \quad (4.99a)$$

$$\mathfrak{R}_\phi^{mi} / \mathfrak{R}_\phi^{mi=1} \leq \lambda_2 \quad (4.99b)$$

$$mi > NIT_\phi \quad (4.99c)$$

in which \mathfrak{R}_ϕ^{mi} is the sum of absolute residues over all cells after mi^{th} iterations for any variable ϕ ; λ_1 and λ_2 are prescribed convergence criteria; and NIT is the maximum number of inner-iterations. Table 4.6 gives the default values of these criteria for each variable ϕ .

Table 4.6 Criteria to stop the inner iteration

ϕ_ℓ	λ_1	λ_2	NIT
u,v,w	10^{-6} [m ⁴ /s ²]	10^{-3}	2
p	10^{-5} [m ³ /s]	10^{-2}	20
k	10^{-6} [m ⁵ /s ³]	10^{-3}	5
ε	10^{-6} [m ⁵ /s ⁴]	10^{-3}	5

The calculation sequence is presented in the calculation diagram depicted in Fig. 4.14, and is summarized as follows:

- a) Initialize all dependent variable: $\phi = \phi^o$ (u,v,w,p,k, ε are given).
- b) Define the geometrical properties, discretise the spatial domain.
- c) Compute initial discharges, q_{cf} , and eddy viscosity, ν_t .
- d) Assign estimated pressures p^* .
- e) Construct coefficients of the discretized momentum equations.
- f) Solve the momentum equations for u^* , v^* , and w^* , consecutively.
- g) Construct coefficients of the discretized pressure correction equation.
- h) Solve the pressure-correction equation for p^c and subsequently update the pressure, p, and velocities at the cell centers, u, v, and w.
- i) Compute the new discharge across cell faces, q_{cf} .
- j) Return to step ‘d’ if the velocity components and pressure do not satisfy the momentum and continuity equations.
- k) Construct coefficients of the discretized k transport equation and solve for k; do the same procedure to get ε .
- l) Compute the eddy viscosity, ν_t , from the new k and ε
- m) Update the water surface if the solution has converged, otherwise assign the new u, v, w, p, k, and ε as the new ‘old’ values and return to step ‘d’.
- n) Stop the iteration if the steady-state solution has been reached, otherwise proceed to the next time step, return to step ‘d’.

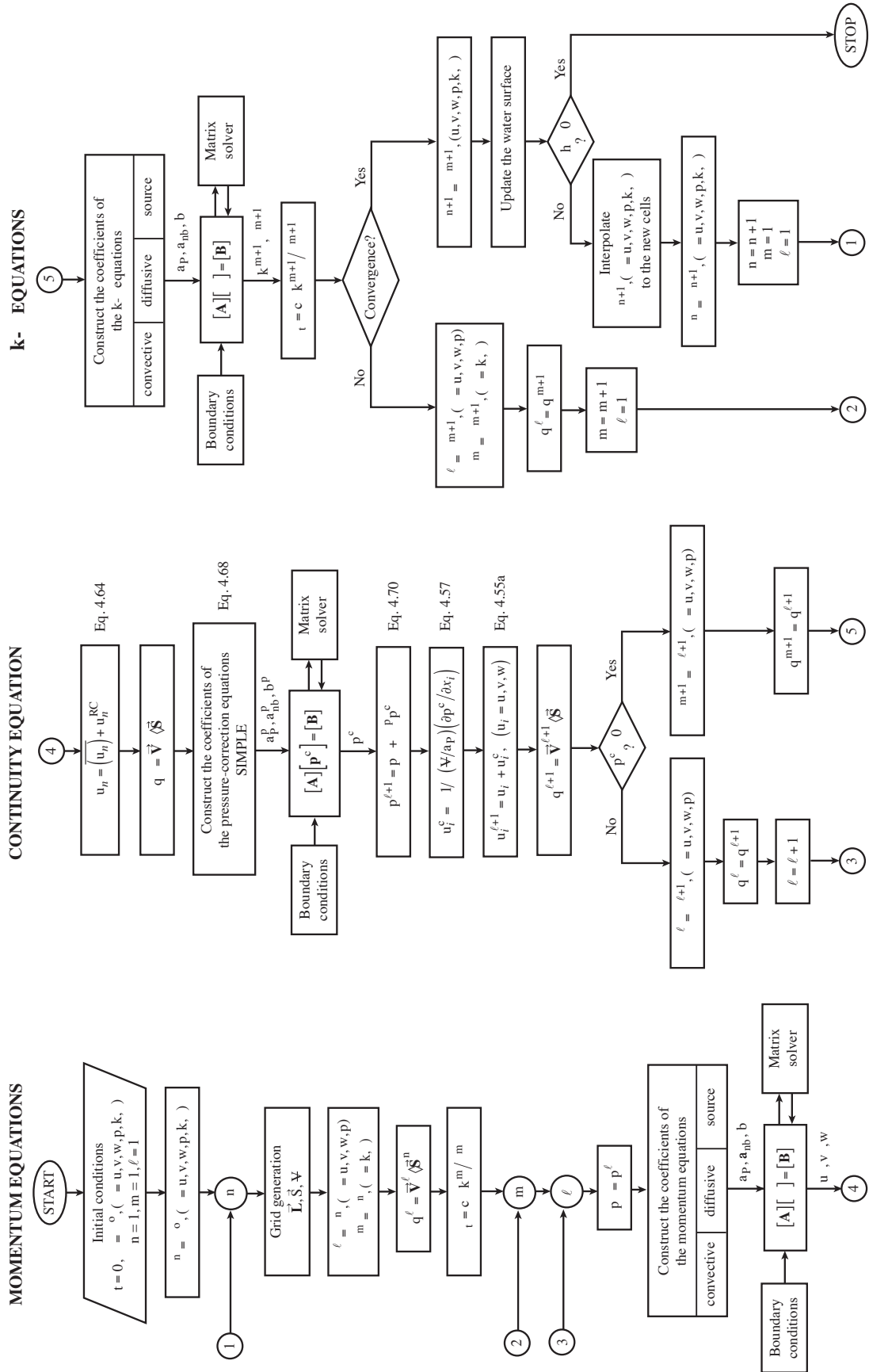


Fig. 4.14 Computational procedures.

4.7 Summary

Development of a three-dimensional numerical flow model has been presented. The model is based on the approximate solution of the Reynolds averaged Navier-Stokes equations, the continuity equations, and the k - ϵ turbulence closure model. These equations are expressed in a general convective-diffusive transport equation on a Cartesian coordinate system. The working equation of the model is obtained by discretizing this transport equation by using finite volume techniques on a structured, collocated, boundary-fitted, hexahedral control-volume grid. The hybrid (*Spalding, 1972*) or power-law (*Patankar, 1980*) upwind-central difference scheme, combined with the deferred correction method (*Ferziger and Peric, 1997*), is employed in the discretisation of the governing equations. The solution of the working equation is achieved by an iterative method according to SIMPLE algorithm (*Patankar and Spalding, 1972*). Along solid boundaries, use is made of the wall function method, while along surface boundaries the pressure defect is used to define the surface position. On other boundaries, namely inlet, outlet, and symmetry boundaries, classical methods are used, such as zero gradients, zero stresses, or known functions.

The model is applicable for steady state flow cases, but not for transient ones. The time step is used as an iteration step to mark, notably, the change of the computational domain due to the moving surface boundary.

References

- Cebeci, T., and Bradshaw, P. (1977). *Momentum Transfer in Boundary Layers.*, Hemisphere Publ. Co., Washington, USA.
- Ferziger, J. H., and Peric, M. (1997). *Computational Methods for Fluid Dynamics.*, Springer-Verlag, Berlin, Germany.
- Fletcher, C. A. J. (1997). *Computational Techniques for Fluid Dynamics, Vol. 1.*, Springer, Berlin, Germany.
- Hirsch, C. (1988). *Numerical Computation of Internal and External Flows, Vol. 1: Fundamentals of Numerical Discretization.*, John Wiley & Sons, Chichester, England.
- Jesshope, C. R. (1979). "SIPSOL – A suite of subprograms for the solution of the linear equations arising from elliptical partial differential equations." *Computer Physics Communications*, 17383-391.
- Kobayashi, M. H., and Pereira, J. C. F. (1991). "Numerical comparison of momentum interpolation methods and pressure-velocity algorithms using non-staggered grids." *Communications in Applied Numerical Methods*, 7173-186.
- Krishnappan, B. G., and Lau, Y. L. (1986). "Turbulence modeling of flood plain flows." *ASCE, J. Hydr. Engrg.*, 112(4), 251-266.

- Launder, B. E., and Spalding, D. B. (1974). “The numerical computation of turbulent flows.” *Computer Methods in Applied Mechanics and Engineering*, 3269-289.
- Obi, S., Peric, M., and Scheuerer, G. (1989). “A finite-volume calculation procedure for turbulent flows with second-order closure and collocated variable arrangement.” *Seventh Symposium on Turbulent Shear Flows*, Stanford University, California, 2, 17.4.1-17.4.6.
- Olsen, N. R. B., and Kjellesvig, H. M. (1998). “Three-dimensional numerical flow modeling for estimation of spillway capacity.” *IAHR, J. of Hydr. Res.*, 36(5), 775-784.
- Patankar, S. V. (1980). *Numerical Heat Transfer and Fluid Flow.*, Hemisphere Publishing Corp., New York, USA.
- Patankar, S. V., and Spalding, D. B. (1972). “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows.” *Int. J. Heat Mass Transfer*, 151787-1806.
- Rhie, C. M., and Chow, W. L. (1983). “Numerical study of the turbulent flow past an airfoil with trailing edge separation.” *AIAA Journal*, 21(11), 1525-1532.
- Rodi, W. (1984). *Turbulence models and their application in hydraulics: A state of the art review.*, IAHR Monograph, Delft, NL.
- Spalding, D. B. (1972). “A novel finite difference formulation for differential expressions involving both first and second derivatives.” *Int. Journal for Numerical Methods in Engineering*, 4551-559.
- Stone, H. L. (1968). “Iterative solution of implicit approximation of multidimensional partial differential equations.” *SIAM J. on Numerical Analysis*, 5530-558.
- Versteeg, H. K., and Malalasekera, W. (1995). *An Introduction to Computational Fluid Dynamics: The Finite Volume Method.*, Longman Group, Essex, England.
- Wu, W., Rodi, W., and Wenka, T. (2000). “3D Numerical modeling of flow and sediment transport in open channels.” *ASCE, J. Hydr. Engrg.*, 126(1), 4-15.

Notations

Capital letters

A	[m ³ /s]	coefficient matrix of the momentum and k-ε equations.
A^p	[m ³ /s/Pa]	coefficient matrix of the pressure-correction equation.
B		vector matrix of the source terms:
	[m ⁴ /s ²]	for the momentum equation,
	[m ⁵ /s ³]	for the k-equation,
	[m ⁵ /s ⁴]	for the ε-equation.
B^p	[m ³ /s]	vector matrix of the source terms for the pressure-correction equation.

$\mathcal{B}, \Delta\mathcal{B}$	[-]	constant and wall roughness function.
\mathcal{E}	[-]	wall roughness coefficient.
E_{Δ}	[-]	time step factor.
F, F^C, F^D		total, convective, and diffusive transports:
	$[m^4/s^2]$	of the momentum flux,
	$[m^5/s^3]$	of the turbulent kinetic-energy flux,
	$[m^5/s^4]$	of the dissipation of kinetic-energy flux.
G	$[m^2/s^3]$	turbulence kinetic-energy production.
\vec{L}	[m]	length vector.
NIT	[-]	number of inner iterations.
P	[-]	cell center.
Pe	[-]	grid Peclet number, the ratio between the convective and diffusive conductance.
Q	$[m^3/s]$	discharge across sectional area of channels.
R		source terms:
	$[m/s^2]$	of the momentum equation,
	$[m^2/s^3]$	of the k-equation,
	$[m^2/s^4]$	of the ϵ -equation.
Re	[-]	Reynolds number.
\vec{S}	$[m^2]$	cell-face surface vector.
\vec{V}	[m/s]	velocity vector.
\vec{V}_n	[m/s]	normal velocity vector.
\vec{V}_t	[m/s]	parallel (tangential) velocity vector.
\vec{V}_*	[m/s]	friction velocity vector.
Ψ	$[m^3]$	cell volume.
Lower case letters		
a_p, a_{nb}	$[m^3/s]$	matrix coefficients of the momentum and k- ϵ equations.
a_p^C, a_{nb}^C	$[m^3/s]$	matrix coefficients of the momentum and k- ϵ equations due to the convective transport.
a_p^D, a_{nb}^D	$[m^3/s]$	matrix coefficients of the momentum and k- ϵ equations due to the diffusive transport.
a_p^p, a_{nb}^p	$[m^3/s/Pa]$	matrix coefficients of the pressure-correction equations.
\tilde{a}_p	$[m^3/s]$	under-relaxed matrix coefficient.
b, \tilde{b}		source term and under-relaxed source term:
	$[m^4/s^2]$	of the momentum equation,
	$[m^5/s^3]$	of the k-equation,
	$[m^5/s^4]$	of the ϵ -equation.
b^D	$[m^4/s^2]$	source term due to the diffusive-correction.
b_p	$[m^3/s]$	source term coefficient due to the source linearisation.
b^p	$[m^3/s]$	source term coefficient of the pressure-correction equation.
c_1, c_2, c_u	[-]	constants of the k- ϵ equation.
\vec{e}	[m]	directional unit vector.
f^D	[-]	coefficient of the hybrid and power-law convective-diffusive schemes, a function of the grid Peclet number.

g_x, g_y, g_z	[m/s ²]	Cartesian components of the gravitational acceleration.
k	[m ² /s ²]	turbulent kinetic energy.
k_s	[m]	wall roughness height.
k_s^+	[-]	Reynolds number based on the friction velocity and wall roughness.
n	[m]	normal direction.
p, p^*, p^c	[Pa]	pressure, estimated pressure, and pressure correction.
q	[m ³ /s]	discharge.
q^*	[m ³ /s]	estimated discharge obtained from (u^*, v^*, w^*) .
q^c, q^{RC}, q^{no}	[m ³ /s]	discharge corrections due to the velocity-correction, cell-face interpolation, and non-orthogonal terms.
$t, \Delta t$	[s]	pseudo-time and pseudo-time step.
u, v, w	[m/s]	Cartesian velocity components.
u^*, v^*, w^*	[m/s]	Cartesian velocity components obtained with estimated pressures.
u', v', w'	[m/s]	fluctuating parts of Cartesian velocity components.
u_*, v_*, w_*	[m/s]	Cartesian friction-velocity components.
u_n^*	[m/s]	velocity obtained with estimated pressure.
u_n^c	[m/s]	pressure corrections.
$u_n^{impl}, u_n^{RC}, u_n^{no}$	[m/s]	pressure correction components: implicit, Rhie-and-Chow, non-orthogonal.
x, y, z	[m]	Cartesian coordinate components.
Greek characters		
Γ	[m ² /s]	diffusion coefficient.
β	[-]	linear interpolation factor.
Δh	[m]	surface displacement.
$\Delta h^{(p)}$	[m]	surface displacement due to the pressure.
$\Delta h^{(z)}$	[m]	surface displacement limitation according to the cell size.
δ_n	[m]	normal distance.
δ_n^+	[-]	dimensionless normal-distance.
ε	[m ² /s ³]	dissipation of the turbulent kinetic energy.
ϕ		flow variable, the dimensional unit depends on the variable of which it represents.
Φ		matrix of the flow variables.
κ	[-]	Karman constant.
λ_1		convergence criterion: allowable maximum residue of the solution:
	[m ⁴ /s ²]	of the momentum equation,
	[m ⁴ /s ²]	of the k-equation,
	[m ⁴ /s ²]	of the ε -equation,
	[m ³ /s]	of the pressure-correction equation.
λ_2	[-]	convergence criterion: allowable relative maximum-residue of the solution.
ν, ν_t	[m ² /s]	kinematic viscosity and turbulent eddy viscosity.
$\nu_{t,wall}$	[m ² /s]	wall eddy-viscosity.
ρ	[kg/m ³]	density of water.
$\varpi, \varpi_1, \varpi^p$	[-]	under-relaxation factors.

$\sigma_k, \sigma_\epsilon$	[-]	constants of the k- ϵ model.
τ	[N/m ²]	Reynolds stress.
ξ, η, ζ	[m]	local coordinate directions.
Other characters		
\mathfrak{R}_ϕ		total residue:
	[m ⁴ /s ²]	of the momentum equation,
	[m ⁴ /s ²]	of the k-equation,
	[m ⁴ /s ²]	of the ϵ -equation,
	[m ³ /s]	of the pressure-correction equation.
$\nabla, \vec{\nabla}$	[1/s]	divergent and gradient nabla operators.
Superscripts		
c		corrected value.
$\ell, \ell + 1$		iteration level indices.
m, m+1		iteration level indices.
n, n+1		time level indices.
*		estimated value.
Subscripts		
P		cell center.
i, j, k		Cartesian component indices.
cf = e,w,n,s,t,b		cell faces: the east, west, north, south, top, and bottom.
ℓ		dependent variable index.
n, t		normal and tangential direction component indices.
nb = E,W,N,S,T,B		neighboring cells: the East, West, North, South, Top, and Bottom.

