Chapter 3

Numerical Implementation

Two finite-volume-based CFD codes have been used in the present work: Turbulent Elliptic Algorithm of Manchester (TEAM) [70, 71] and Simulation of Turbulent Reynolds-averaged Equations for All Mach numbers (STREAM) [72, 73]. Only the main elements of the two codes are presented which have direct relevance to the current study. More information on the SIMPLE algorithm, finite-volume discretization etc. can be found in numerous books on CFD techniques (for example, Patankar [74], Versteeg & Malalasekera [75] and Ferziger & Perić [76]).

3.1 The Finite-Volume Method

In the finite-volume method, the governing equations are integrated over each of the finite control volumes in the flow domain and the resulting integrated transport equations are then discretized, using finite-difference-type formulas, to give a set of algebraic equations which are solved using an iterative method (e.g. TDMA). To illustrate this procedure, one can consider a simple one-dimensional convection-diffusion transport equation for steady flow, which is written as follows:

$$\frac{\partial(\rho U\phi)}{\partial x} = \frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right) + C \tag{3.1}$$

where ϕ is the transport variable, Γ is the diffusion coefficient and C contains the source terms. This is integrated over a finite volume as follows:

$$\int_{\Delta Vol}\frac{\partial(\rho U\phi)}{\partial x}\mathrm{d}Vol = \int_{\Delta Vol}\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right)\mathrm{d}Vol + \int_{\Delta Vol} C\mathrm{d}Vol \tag{3.2}$$

where the ΔVol is the cell volume. Considering the cell to have constant cross-sectional area, A, between east and west faces (denoted e and w, respectively), the above equation can be written:

$$(\rho UA\phi)_e - (\rho UA\phi)_w = \left(\Gamma A\frac{\partial\phi}{\partial x}\right)_e - \left(\Gamma A\frac{\partial\phi}{\partial x}\right)_w + C_P\Delta Vol \tag{3.3}$$

The diffusion terms are discretized using central differencing, with quantities evaluated at the cell faces, $()_e$ and $()_w$, being replaced by expressions involving nodal values, as follows:

$$\left(\Gamma A\frac{\partial\phi}{\partial x}\right)_e = (\Gamma A)_e \frac{(\phi_E - \phi_P)}{\Delta x_{PE}} \tag{3.4}$$

$$\left(\Gamma A\frac{\partial\phi}{\partial x}\right)_w = (\Gamma A)_w \frac{(\phi_P - \phi_W)}{\Delta x_{PW}} \tag{3.5}$$

where subscripts P, E and W refer to values at the current node, the neighbouring eastern and western nodes, respectively and Δx_{PW} is the distance between nodes P and W. Substituting these expressions into Equation (3.3), the discretized transport equation becomes:

$$F_e\phi_e - F_w\phi_w = D_e(\phi_E - \phi_P) - D_w(\phi_P - \phi_W) + S \tag{3.6}$$

where $F = \rho UA$ is the convective mass flux, $D = \Gamma A/\Delta x$ is the diffusion coefficient and $S = C_P\Delta Vol$ is the integrated source term.

Central differencing is not used to approximate convective fluxes for reasons of boundedness and transportiveness. A number of different discretization schemes for convection are outlined in Sections 3.2.2 and 3.3.3. For the present purposes of illustration the simplest scheme is adopted: the upwind scheme, which approximates the boundary value as the upstream nodal value. For positive U-velocity (i.e. from west to east) and for negative U-velocity (east to west) the value of ϕ on the eastern cell face is approximated as:

$$\phi_e = \phi_P \quad U > 0 \tag{3.7}$$

$$\phi_e = \phi_E \quad U < 0 \tag{3.8}$$

Similar expressions can be written for the ϕ at the western cell face. The convective flux terms can thus be expressed compactly as follows:

$$F_e\phi_e = \phi_P \max(F_e, 0) + \phi_E \min(F_e, 0) \tag{3.9}$$

$$F_w\phi_w = \phi_W \max(F_w, 0) + \phi_P \min(F_w, 0) \tag{3.10}$$

The discretized 1-D convection-diffusion equation can now be written (substituting Equations 3.9 and 3.10 into 3.6):

$$a_P\phi_P = a_E\phi_E + a_W\phi_W + S \tag{3.11}$$

where:

$$a_E = D_e - \min(F_e, 0) \tag{3.12}$$

$$a_W = D_w + \max(F_w, 0) \tag{3.13}$$

$$a_P = a_E + a_W + (F_e - F_w) \tag{3.14}$$

and $F_e - F_w = 0$, from continuity (Equation 2.1).

The discretized governing equation (Equation 3.11) is typically solved using the Tri-Diagonal Matrix Algorithm (TDMA)[1]. In order to improve the stability of the TDMA, the source term (S) is decomposed into two parts:

$$S = s_U + s_P \phi_P \tag{3.15}$$

In general, if s_P is negative then it is transferred to the left-hand-side of the discretized equation:

$$[a_P - \min(s_P, 0)]\,\phi_P = a_E \phi_E + a_W \phi_W + s_U + \max(s_P, 0)\,\phi_P \tag{3.16}$$

which improves the diagonal dominance of the coefficient matrix and hence the stability.

For those flow variables that, by definition, must always be positive, such as k and ε, slightly different rules are applied: when the source term, S, is negative it is transferred to the left-hand-side of the discretized equation to prevent physically unrealistic negative values:

$$\left[a_P - \frac{\min(S, 0)}{\phi_P}\right] \phi_P = a_E \phi_E + a_W \phi_W + \max(S, 0) \tag{3.17}$$

It is relatively straight-forward to extend the above derivation to two and three-dimensional cases. The resulting discretized transport equations have the following general form:

$$a_P \phi_P = \sum_{nb} a_{nb} \phi_{nb} + S \tag{3.18}$$

where:

$$a_P = \sum_{nb} a_{nb} \tag{3.19}$$

and *nb* indicates the neighbouring nodes (E, W, N, S, T and B).

3.1.1 SIMPLE Pressure-Correction Algorithm

For the RANS equations there is no direct method of specifying an equation for pressure. Instead, pressure is determined indirectly using the continuity equation: if the correct pressure field is used to solve the momentum equations then the continuity equation will be satisfied. In both the TEAM and STREAM codes, the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) of Patankar & Spalding [78] is used to handle the pressure-velocity coupling.

The discretized U-momentum equation for a two-dimensional geometry can be written:

$$a_P U_P = \sum_{nb} a_{nb} U_{nb} + (P_w - P_e)\,\Delta y \tag{3.20}$$

where *nb* indicates the neighbouring nodes (E, W, N and S). The SIMPLE algorithm starts by assum-

[1] Details of the TDMA (or Thomas' algorithm) are not provided here but can be found in most CFD textbooks, for example Anderson [77].

ing a guessed pressure field, P^*, and guessed velocity field, U^*:

$$a_P U_P^* = \sum_{nb} a_{nb} U_{nb}^* + (P_w^* - P_e^*)\,\Delta y \tag{3.21}$$

It is assumed that to satisfy the continuity equation, one must make a correction to the guessed velocity and pressure fields. The correct velocity and pressure fields are given by:

$$U = U^* + U^{'} \tag{3.22}$$

$$P = P^* + P^{'} \tag{3.23}$$

where $U^{'}$ and $P^{'}$ are the necessary corrections. One can derive an equation for the necessary correction, $U^{'}$, by subtracting Equation (3.21) from (3.20):

$$U_P^{'} = \underbrace{\frac{\sum_{nb} a_{nb} U_{nb}^{'}}{a_P}} + \frac{\Delta y}{a_P}\left(P_w^{'} - P_e^{'}\right) \tag{3.24}$$

The SIMPLE algorithm assumes that the underbraced term in Equation (3.24) is negligible. The correct velocity is thus given by:

$$U_P = U_P^* + \frac{\Delta y}{a_P}\left(P_w^{'} - P_e^{'}\right) \tag{3.25}$$

The discretized continuity equation for a two-dimensional Cartesian grid can be written:

$$(\rho UA)_e - (\rho UA)_w + (\rho VA)_n - (\rho VA)_s = 0 \tag{3.26}$$

where *e*, *w*, *n* and *s* denote values at the east, west, north and south cell faces, respectively. The expressions for the correct velocity through the cell faces (U_e, U_w, V_n and V_s) can be obtained in a similar manner to that described above for U_P, and substituted into the continuity equation. The resulting expression can be rearranged in terms of the pressure-correction, in the form:

$$a_P P_P^{'} = \sum_{nb} a_{nb} P_{nb}^{'} + S \tag{3.27}$$

This equation is solved to find the pressure-correction $P^{'}$ at all points throughout the flow domain. Once the pressure correction is known, it is then possible to update the guessed pressure (P^*) to its correct value (P) and update the velocity field from Equation (3.25).

The steps involved in the SIMPLE algorithm can be summarized as follows:

1. Initially a pressure field is guessed, P^*, which used in the discretized momentum equations to find the guessed velocity field.

2. An equation is solved for the pressure correction, $P^{'}$, which gives the pressure difference between the guessed pressure and pressure necessary to satisfy the continuity condition.

3. The pressure and velocity fields are then updated based on the pressure correction (with some under-relaxation).

4. Other scalar transport equations are solved (e.g. k and ε).

5. Steps 1 – 4 are repeated (using the pressure field from the previous step as the initial guess) until the calculation has converged.

3.1.2 Under-Relaxation

Under-relaxation is a means of slowing down the updating process of the variables and is necessary for convergence of coupled non-linear equations. Variables are updated in the following manner:

$$\phi_P^{new} = \alpha\phi_P^{calc} + (1-\alpha)\phi_P^{old} \tag{3.28}$$

where ϕ_P^{old} is the value of ϕ from the previous iteration and ϕ_P^{calc} is the result of the current calculation. Applying the above formula to the discretized transport equation (Equation 3.18) gives:

$$\frac{a_P}{\alpha}\phi_P^{new} = \sum_{nb} a_{nb}\phi_{nb} + S + (1-\alpha)\frac{a_P}{\alpha}\phi_P^{old} \tag{3.29}$$

In practice, under-relaxation involves the modification of the coefficient a_P and source S:

$$(a_P)_{new} = \frac{a_P}{\alpha} \tag{3.30}$$

$$(S)_{new} = S + (1-\alpha)(a_P)_{new}\phi_P^{old} \tag{3.31}$$

Its effect, therefore, is to increase the diagonal dominance of the coefficient matrix and add in a source term.

Since the pressure is updated by means of a pressure correction, its under-relaxation takes the form:

$$P_P^{new} = P_P^* + \alpha P_P^{'} \tag{3.32}$$

where P' is the pressure correction.

If the under-relaxation factor, α, in the above equations is set to unity there is no under-relaxation and the new value is equivalent to the calculated value. Typical values for the under-relaxation factors used in the impinging jet, spinning disc and Ahmed body studies are shown in Tables 5.1, 6.1 and 7.2, respectively (see pages 97, 114 and 136).

3.2 TEAM Code

TEAM is a finite-volume code for the simulation of steady two-dimensional or axisymmetric turbulent elliptic flows. It was used to study the fully-developed channel flow, impinging jet and spinning disc

flows, documented in Chapters 4 to 6.

3.2.1 Storage Arrangement

A staggered Cartesian grid is used with scalars stored at nodal positions and velocities at the cell faces (see Figure 3.1). This arrangement is used to prevent a "checker-board" pressure field being developed, where neighbouring pressure nodes have alternating high-low values but the cell face pressures calculated using linear interpolation are constant across the flow field (so in effect the velocity field "feels" a constant pressure). The grid is specified algebraically by setting the locations of the cell faces, with nodes then being placed in the centre of each cell. Boundary nodes are located along the edges of the flow domain which are used to apply the boundary conditions. Transport equations are solved up to, but not including, the boundary nodes (i.e. in Figure 3.1 transport equations solved from node *P* northwards but not at node *S*).

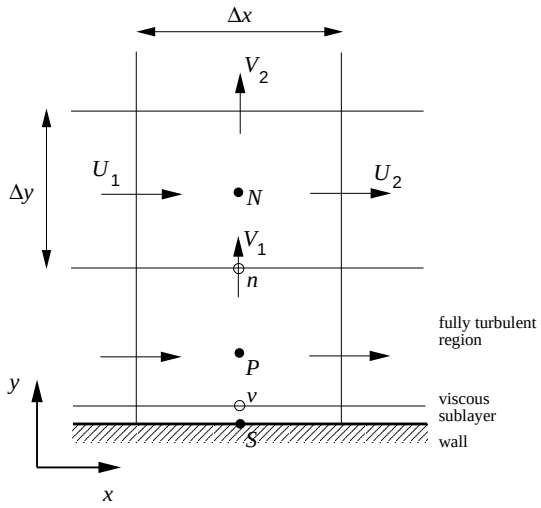

Figure 3.1: Staggered grid arrangement used by TEAM: scalars stored at nodes (e.g. *P* and *N*), velocities stored at cell faces (e.g. V_1 stored at *n*), edge of viscous sublayer located at *v*, boundary node *S* located on wall. In axisymmetric coordinates the *x*-axis becomes the radial *r*-axis.

In the staggered grid arrangement, the wall shear stress is evaluated at cell faces (where the velocity is calculated) whilst the average turbulent scalar source terms ($\overline{P_k}$, $\overline{\varepsilon}$ etc.) and wall heat flux, q_{wall}, are evaluated at the scalar node in the centre of the cell. The UMIST-*N* wall function nodes are positioned in the centre of the subgrid cells, in the equivalent position to the main-grid scalar nodes. The wall shear stress is obtained at the scalar node position and is then linearly interpolated to the cell boundary location. It was found that this gave identical results to those obtained when the wall function was applied twice per cell (at both scalar and velocity nodes).

3.2.2 Differencing Schemes

In the TEAM code, diffusion terms appearing in the transport equations for momentum and turbulence parameters are discretized using second-order central differencing. There are two options for approximating the convection terms: the Power-Law Differencing Scheme (PLDS) of Patankar [74] and the Quadratic Upstream Interpolation for Convective Kinetics (QUICK) scheme of Leonard [79].

PLDS

PLDS is first-order accurate with respect to its Taylor series truncation error and is unconditionally bounded. For a uniform mesh and positive U-velocity (from west to east), the eastern boundary value ϕ_e is approximated as:

$$\begin{aligned} \phi_e &= \phi_P + \frac{\phi_E - \phi_P}{2} \frac{(1-0.1Pe_e)^5}{(1-0.05Pe_e)^5} \quad \text{for} \quad 0 \le Pe \le 10 \\ \phi_e &= \phi_P \qquad\qquad\qquad\qquad\qquad\quad \text{for} \quad Pe > 10 \end{aligned} \tag{3.33}$$

where subscript E refers to the eastern nodal value. Pe is the non-dimensional cell Peclet number, a measure of the relative strength of convection to diffusion:

$$Pe = \frac{F}{D} = \frac{\rho U}{\Gamma/\Delta x} \tag{3.34}$$

For Peclet number greater than 10, the scheme is equivalent to a simple upwind approach (see discussion in Section 3.3.3).

QUICK

QUICK is a third-order accurate unbounded scheme, where the interpolation for the cell face value, ϕ_e or ϕ_w, is based on a quadratic interpolation using three nodal values, one on the downstream side of the cell face and two on the upstream side. For a uniform mesh, if the U-velocity is positive:

$$\phi_e = \frac{1}{2}(\phi_E + \phi_P) - \frac{1}{8}(\phi_E - 2\phi_P + \phi_W) \tag{3.35}$$

$$\phi_w = \frac{1}{2}(\phi_P + \phi_W) - \frac{1}{8}(\phi_P - 2\phi_W + \phi_{WW}) \tag{3.36}$$

where subscript W refers to the western nodal value and WW is the node to the west of the W node. Boundary values, ϕ_e and ϕ_w, obtained by QUICK can be thought of as a combination of linear interpolation and upstream-weighted correction. Since the discretized equations are solved using a TDMA which only treats the near-diagonal terms implicitly (a_P, a_E and a_W), the other terms (a_{EE} and a_{WW}) are included in the source term. Whilst QUICK is a more accurate scheme than PLDS, it is also unbounded and can suffer from over- or under-shoots. In the impinging jet flow, discussed later, values of k and ε are close to zero near the entrainment boundary. As a consequence, small undershoots

can lead to negative values of k and ε which are physically unrealistic and cause stability problems. Therefore PLDS was used for turbulence scalars and QUICK for the velocity field.

3.2.3 Wall-Function Implementation

Wall-Parallel Velocity

The coding of wall functions into TEAM is much simpler than for STREAM owing to the use of a Cartesian grid. To implement the wall-function conditions for the wall-parallel U-velocity, there are two parts. In the first, the diffusional flux to the wall calculated by the near-wall main-grid cell is set to zero, by setting $a_S = 0$ (for a wall on the south face) in the discretized momentum equation. To understand the effect of setting $a_S = 0$, one can consider the diffusion term:

$$\frac{\partial}{\partial y}\left(\mu_{eff}\frac{\partial U}{\partial y}\right) \tag{3.37}$$

This is discretized, using central differences, and integrated over the cell using the finite-volume method to give:

$$\int \frac{\partial}{\partial y}\left(\mu\frac{\partial U}{\partial y}\right)\mathrm{d}Vol = \left(\mu A\frac{\partial U}{\partial y}\right)_n - \underbrace{\left(\mu A\frac{\partial U}{\partial y}\right)_s} \tag{3.38}$$

where A is the area of the cell face parallel to the wall and n and s refer to north and south boundary values. By setting $a_S = 0$, one is setting the underbraced term to zero. This is necessary since the $\partial U/\partial y$ part of the underbraced term would otherwise be calculated assuming there to be a linear change in velocity between the southern node S (on the wall surface) and the near-wall node, P (see Figure 3.1). When a fine near-wall grid is used with a low-*Re* model, a linear U-velocity profile provides an adequate approximation of the velocity variation between nodes S and P, but when a large near-wall cell is used one needs to assume, or calculate, a more accurate profile. Standard wall functions replace $\partial U/\partial y$ with a value obtained from assuming a logarithmic U-velocity profile whilst the UMIST-*N* wall function finds $\partial U/\partial y$ from a local solution of the flow field. Once a_S has been set to zero, the second part of the wall-function implementation is to add in a replacement for the underbraced term. The replacement term consists of the wall shear stress (effectively the $\mu_{eff}\partial U/\partial y$ part of Equation 3.38) multiplied by the area of the cell face parallel to the wall:

$$F_{wall} = -\tau_{wall}A \tag{3.39}$$

The negative sign is introduced since the diffusive flux in Equation (3.38) is negative and the complete term is called the "wall force", F_{wall}. This is added into the source term in the discretized wall-parallel momentum equations at the near-wall node. If a log-law wall function is used, the wall force is expressed in terms of the velocity at the near-wall node: for a near-wall node in the fully-turbulent

region of the boundary layer, $y^+ > 11.6$, the wall force is given by:

$$F_{wall} = -\frac{\rho\kappa c_\mu^{1/4} k^{1/2} U_P A}{\ln(Ey^+)} \tag{3.40}$$

or if $y^+ < 11.6$, the wall force becomes:

$$F_{wall} = -\mu\frac{U_P}{y_P}A \tag{3.41}$$

Since U_P appears explicitly in the above expressions and the term (F_{wall}/U_P) is always negative, one can place the wall force into the linearized source term, s_P:

$$s_P = \frac{F_{wall}}{U_P} \tag{3.42}$$

This increases the diagonal dominance of the coefficient matrix for the discretized U-momentum equation, improving stability.

The UMIST-*N* wall function does not calculate τ_{wall} as a function of U_P and therefore one cannot linearize the source term. Instead the wall force calculated by UMIST-*N* wall function $(F_{wall} = -\tau_{wall}A)$ is added into the source term s_U, i.e.:

$$s_U = F_{wall} \tag{3.43}$$

Tangential Velocity

In the spinning-disc case examined in Chapter 6, the wall is rotating at constant tangential velocity, $W_{wall} = \Omega r$. The tangential velocity log-law can be written, for a stationary reference frame:

$$\frac{W - W_{wall}}{W_\tau} = \frac{1}{\kappa}\ln\left(Ey^+\right) \tag{3.44}$$

Using a standard wall function, the wall force in the tangential direction, $F_{wall,\phi}$, is calculated for $y^+ > 11.6$:

$$F_{wall,\phi} = -\frac{\rho\kappa c_\mu^{1/4} k^{1/2}\left(W_P - W_{wall}\right)A}{\ln(Ey^+)} \tag{3.45}$$

The source term is split into the s_U and s_P source terms as follows:

$$s_U = \frac{\rho\kappa c_\mu^{1/4} k^{1/2} W_{wall} A}{\ln(Ey^+)} \tag{3.46}$$

$$s_P = -\frac{\rho\kappa c_\mu^{1/4} k^{1/2} A}{\ln(Ey^+)} \tag{3.47}$$

Similarly, if the near-wall node is within the viscous sublayer ($y^+ < 11.6$):

$$F_{wall,\phi} = -\mu \frac{(W_P - W_{wall})}{y_P} A \tag{3.48}$$

and the source terms become:

$$s_U = \mu \frac{W_{wall}}{y_P} A \tag{3.49}$$

$$s_P = -\frac{\mu}{y_P} A \tag{3.50}$$

The tangential wall shear stress calculated by the UMIST-*N* wall function is not a function tangential velocity at the near-wall node, W_P, and therefore the tangential wall force calculated by UMIST-*N* is added into the source term s_U.

Wall-Normal Velocity

In a staggered-grid arrangement, the wall-normal velocity, V, is calculated at the wall boundary node and at the north face of the control volume (see Figure 3.1). For a non-porous wall, the V-velocity is simply set to zero on the wall surface. One does not need to make any other modifications to the V-momentum equation.

Turbulence Parameters

In the four "standard" wall functions discussed in Chapter 2, the *k*-equation is solved in the near-wall cell with modified source terms ($\overline{P_k}$ and $\overline{\varepsilon}$), whilst the ε-equation is not solved but instead its value is prescribed at the near-wall node. Introducing modified source terms into the *k*-equation is simply a matter of removing the old calculated near-wall cell values of production and dissipation, P_k and ε, from the source term of the *k*-equation and then adding in the wall-function values. Since the standard wall functions involve expressions for the production due purely to shear stress, only the shear stress component of P_k is removed. The wall function values for average production due to shear stress, $\overline{P_{kuv}}$, and average dissipation rate, $\overline{\varepsilon}$, are placed into source terms s_U and s_P as follows:

$$s_U = \max\left[\left(\overline{P_{kuv}} - \overline{\rho\varepsilon}\right), 0\right] \Delta Vol \tag{3.51}$$

$$s_P = \frac{\min\left[\left(\overline{P_{kuv}} - \overline{\rho\varepsilon}\right), 0\right]}{k_P} \Delta Vol \tag{3.52}$$

where ΔVol is the volume of the near-wall cell. To set the value of ε_P, source terms are added to the discretized equation as follows:

$$s_U = 10^{30} \varepsilon_P^* \qquad\qquad s_P = -10^{30} \tag{3.53}$$

where 10^{30} is an arbitrary large number and ε_P^* is the wall function value for ε at the near-wall node. Substituting these source terms into the discretized equation leads to the following expression:

$$\left(a_P + 10^{30}\right)\varepsilon_P = a_N\varepsilon_N + a_S\varepsilon_S + a_E\varepsilon_E + a_W\varepsilon_W + 10^{30}\varepsilon_P^* \tag{3.54}$$

and, since the neighbouring coefficients $a_N\phi_N$, $a_S\phi_S$ are much smaller than 10^{30} , the expression becomes:

$$\varepsilon_P = \varepsilon_P^* \tag{3.55}$$

The UMIST-*N* wall function involves the solution of both k and $\tilde{\varepsilon}$ equations in the near-wall cell with modified source terms. A similar approach to that suggested above for the k-equation (Equations 3.51 and 3.52) is therefore applied for both k and $\tilde{\varepsilon}$ equations. The cell-averaged production term calculated across the subgrid, $\overline{P_k}$, includes both shear and normal components and so the entire production term, P_k, is removed from the near-wall cell prior to adding in the wall function value.

Using both standard and UMIST-*N* treatments, the diffusive fluxes of k and ε to the wall are set to zero, by setting $a_S = 0$.

Temperature

Implementing the temperature wall function is similar to the procedure used for wall-parallel momentum but instead of replacing the wall shear stress one replaces the wall heat flux, and instead of assuming a no-slip condition at the wall one prescribes (or calculates) the wall temperature. The temperature equation is solved at the near-wall node, suppressing the heat flux to the wall calculated using a linear temperature gradient by setting $a_S = 0$. If constant heat flux conditions are prescribed, one knows the heat flux to the wall, q_{wall}, and must calculate the wall temperature T_{wall}. Conversely, if constant temperature conditions are applied, one knows T_{wall} and has to calculate q_{wall}. The expressions for T_{wall} and q_{wall} used by standard wall functions are presented in Section 2.4 and those calculated by the UMIST-*N* wall function in Section 4.3. The temperature at the node on the wall surface is set to T_{wall} and the following source term is added into the discretized temperature equation in the near-wall cell:

$$s_U = -\frac{q_{wall}}{c_p}A \tag{3.56}$$

where A is the area of the cell face parallel to the wall.

3.2.4 Convergence Criteria

If the discretized transport equations are solved exactly then the left and the right sides of the discretized transport equation (Equation 3.18) will be exactly equal. However, since the set of non-linear coupled equations requires an iterative solution, an imbalance or residual may exist between the two sides of the discretized equations. The residual for parameter ϕ in a particular cell is calculated from

the imbalance as follows:

$$R_\phi = a_P \phi_P - \sum_{nb} a_{nb} \phi_{nb} - S \tag{3.57}$$

where *nb* indicates the neighbouring nodes (E, W, N and S). If the calculation is completely converged then the R_ϕ value at all nodal positions is zero. In the TEAM code, an overall residual for each variable is determined by summing the absolute values of the residuals over all the main-grid nodes and normalizing with bulk inlet quantities (for definitions, see [70]). A mass imbalance is calculated rather than a pressure-residual. The calculation is said to have converged when the normalized velocity and mass residuals are less than a prescribed value. For the impinging jet flow this was 5×10^{-4} which corresponded to a reduction in the residuals of approximately four orders-of-magnitude from the start to the end of a calculation. It was verified that using more stringent convergence criteria had no effect on results.

3.3 STREAM Code

The STREAM code is significantly more sophisticated than the TEAM code described above. It can be used in either 2-*D*, 3-*D* or axisymmetric modes using a non-orthogonal body-fitted grid if necessary. Both steady and unsteady flows can be analyzed and mass-weighted averaging can be activated if compressibility effects are significant. STREAM was used to model the flow around the Ahmed body, considered in Chapter 7. For the purposes of this work the STREAM code was only used in steady, incompressible mode.

The STREAM code has many similarities to the TEAM code, described above. The significant issues of difference that are important to the present work are discussed below. Transport equations using curvilinear coordinates, which are solved by STREAM, are presented in Appendix F. For a full description of the code, see Lien [73].

3.3.1 Grid Arrangement

The STREAM code uses a fully-collocated storage arrangement with all velocity components and scalar quantities stored at nodal positions. In order to overcome "checker-board" problems with the pressure field, a Rhie-Chow interpolation is used to calculate velocities at cell faces (see below). A structured non-orthogonal curvilinear grid arrangement is used although the velocity vectors (and hence Reynolds stress components) are always aligned to a Cartesian reference frame. Using Cartesian coordinates for velocity is useful since it allows the momentum equations to be written in conservative form. For complex geometry, the flow domain is decomposed into a number of separate blocks where block-to-block communication is achieved using additional halo nodes. Nodes are located in the centre of each control volume and along the edges of blocks there are boundary nodes. Transport equations are only solved for nodes internal to cells; boundary nodes and halo nodes provide boundary conditions.

3.3.2 Dimensionless Parameters

The STREAM code solves transport equations for dimensionless parameters. The momentum, pressure and turbulence scalar parameters are all non-dimensionalized with bulk quantities. Non-dimensional forms of the parameters are as follows:

$$\underline{U_i} = \frac{U_i}{U'};\ \underline{x_j} = \frac{x_j}{L};\ \underline{t} = t\frac{U}{L} \tag{3.58}$$

$$\underline{\rho} = \frac{\rho}{\rho'};\ \underline{P} = \frac{P}{\rho'(U')^2};\ \underline{\overline{u_i u_j}} = \frac{\overline{u_i u_j}}{(U')^2} \tag{3.59}$$

where underlined characters denote dimensionless parameters, ρ', U' and L are characteristic density, velocity and length scales and the pressure P and dimensionless pressure $\underline{P}$ are measured from the same reference pressure. The momentum equation in Cartesian tensors is given by:

$$\begin{aligned}&\frac{\partial}{\partial t}(\rho U_i) + \frac{\partial}{\partial x_j}(\rho U_i U_j) = \\ &-\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial U_m}{\partial x_m}\right)\right] + \frac{\partial}{\partial x_j}(-\rho\overline{u_i u_j})\end{aligned} \tag{3.60}$$

Substituting for dimensionless parameters gives:

$$\begin{aligned}&\frac{\rho'(U')^2}{L}\frac{\partial}{\partial \underline{t}}(\rho\underline{U_i}) + \frac{\rho'(U')^2}{L}\frac{\partial}{\partial \underline{x_j}}\left(\rho\underline{U_i}\underline{U_j}\right) = \\ &-\frac{\rho'(U')^2}{L}\frac{\partial \underline{P}}{\partial \underline{x_i}} + \frac{U'}{L^2}\frac{\partial}{\partial \underline{x_j}}\left[\mu\left(\frac{\partial \underline{U_i}}{\partial \underline{x_j}} + \frac{\partial \underline{U_j}}{\partial \underline{x_i}} - \frac{2}{3}\delta_{ij}\frac{\partial \underline{U_m}}{\partial \underline{x_m}}\right)\right] \\ &+\frac{\rho'(U')^2}{L}\frac{\partial}{\partial \underline{x_j}}\left(-\rho\underline{\overline{u_i u_j}}\right)\end{aligned} \tag{3.61}$$

which can be rearranged as:

$$\begin{aligned}&\frac{\partial}{\partial \underline{t}}(\rho\underline{U_i}) + \frac{\partial}{\partial \underline{x_j}}\left(\rho\underline{U_i}\underline{U_j}\right) = \\ &-\frac{\partial \underline{P}}{\partial \underline{x_i}} + \frac{\partial}{\partial \underline{x_j}}\left[\frac{1}{Re}\left(\frac{\partial \underline{U_i}}{\partial \underline{x_j}} + \frac{\partial \underline{U_j}}{\partial \underline{x_i}} - \frac{2}{3}\delta_{ij}\frac{\partial \underline{U_m}}{\partial \underline{x_m}}\right)\right] + \frac{\partial}{\partial \underline{x_j}}\left(-\rho\underline{\overline{u_i u_j}}\right)\end{aligned} \tag{3.62}$$

where Re is the Reynolds number given by:

$$Re = \frac{\rho' U' L}{\mu} \tag{3.63}$$

In the STREAM code, the molecular viscosity appearing in the subgrid momentum and scalar equations is therefore given by the inverse of the bulk flow Reynolds number, $\mu \rightarrow (Re)^{-1}$. To simulate the same flow at different Reynolds numbers one simply modifies the molecular viscosity term. For

example, a channel flow with Reynolds number $Re = 100,000$ (based on the wall-to-wall channel height and bulk velocity) has the flow domain set up with a wall-to-wall channel height of unity, an inlet bulk velocity of unity and a molecular viscosity of $\mu = 1/100,000$. To switch to a Reynolds number of $Re = 1000$ one simply sets the molecular viscosity to $\mu = 1/1000$.

3.3.3 Differencing Schemes

The STREAM code uses central differences to approximate diffusion terms. There are two options in the code for approximating convective transport: the upwind scheme [80] and the Upstream Monotonic Interpolation for Scalar Transport (UMIST) scheme [81].

Upwind

The upwind scheme simply approximates the boundary value as the upstream nodal value. For a positive U-velocity (from west to east) the values of ϕ on the eastern and western boundaries are approximated as:

$$\phi_e = \phi_P \tag{3.64}$$

$$\phi_w = \phi_W \tag{3.65}$$

The upwind scheme is unconditionally bounded but is only first-order accurate. The first-order truncation error term in the Taylor series expansion for the gradient $\partial\phi/\partial x$ contains a second-order derivative term, $\partial^2\phi/\partial x^2$. If insufficient grid nodes are employed, the upwind scheme is therefore liable to introduce unacceptable levels of numerical diffusion (or artificial viscosity). This is only a problem if the flow is not aligned to the grid, when the numerical inaccuracy of the upwind scheme can introduce cross-flow diffusion far in excess of the physical value if the cell Peclet number is greater than $Pe = 2$ [73].

UMIST

UMIST is a TVD (Total Variation Diminishing) scheme based on third-order accurate scheme QUICK (described in Section 3.2.2). A limiter is used to diminish oscillations caused by dispersive truncation errors occurring with QUICK. The resulting scheme returns solutions which are close to those of QUICK but without its oscillatory features. Since the scheme is not oscillatory it can be used for convective transport of both mean-flow and turbulence scalars. For details of the scheme see Lien & Leschziner [81].

3.3.4 Wall-Function Implementation

The implementation of wall functions into STREAM follows the same general procedure presented above for TEAM. There are three notable points of difference. Firstly, when a standard wall function is used, such as the simplified Chieng & Launder scheme, the cell averaged production term, $\overline{P_k}$, used

in the near-wall cell k-equation includes only the shear stress component. Secondly, the use of a body-fitted non-orthogonal grid introduces some complications into the calculation of the wall shear stress with standard wall functions and, thirdly, since a collocated grid arrangement is employed one needs to calculate the pressure on the wall surface. Points two and three are examined below in greater detail.

In STREAM, the main-grid velocity vectors are aligned with Cartesian axes. The velocity vector, $\mathbf{U}$, has components:

$$\mathbf{U} = U_x\hat{\mathbf{i}} + U_y\hat{\mathbf{j}} + U_z\hat{\mathbf{k}} \tag{3.66}$$

where $\left(\hat{\mathbf{i}},\hat{\mathbf{j}},\hat{\mathbf{k}}\right)$ are the Cartesian unit vectors and (U_x,U_y,U_z) the velocity components. The wall-normal unit-vector $\hat{\mathbf{n}}$ has Cartesian components:

$$\hat{\mathbf{n}} = \hat{n}_x\hat{\mathbf{i}} + \hat{n}_y\hat{\mathbf{j}} + \hat{n}_z\hat{\mathbf{k}} \tag{3.67}$$

The magnitude of the projection of vector $\mathbf{U}$ in the wall-normal direction, $|\mathbf{U}^n|$, is calculated from the scalar product of $\mathbf{U}$ and $\hat{\mathbf{n}}$:

$$\begin{aligned} |\mathbf{U}^n| = \mathbf{U}\cdot\hat{\mathbf{n}} &= |\mathbf{U}|\cdot|\hat{\mathbf{n}}|\cos\theta \\ &= U_x\hat{n}_x + U_y\hat{n}_y + U_z\hat{n}_z \end{aligned} \tag{3.68}$$

The magnitude of the velocity vector parallel to the wall, $|\mathbf{U}^t|$, can be found from Pythagoras:

$$\begin{aligned} \left|\mathbf{U}^t\right| &= \left(\mathbf{U}^2 - |\mathbf{U}^n|^2\right)^{1/2} \\ &= \left(U_x^2 + U_y^2 + U_z^2 - |\mathbf{U}^n|^2\right)^{1/2} \end{aligned} \tag{3.69}$$

where $\mathbf{U}^2$ is the dot product $(\mathbf{U}\cdot\mathbf{U})$. It is this term, the magnitude of the velocity component which is tangential to the wall $|\mathbf{U}^t|$, which is used by standard log-law wall functions to determine the wall shear stress, τ_{wall}, i.e. in the standard log-law expression:

$$U^+ = \frac{1}{\kappa}\ln\left(Ey^+\right) \tag{3.70}$$

the dimensionless velocity, U^+, is given by:

$$U^+ = \frac{|\mathbf{U}^t|}{U_\tau} \tag{3.71}$$

The dimensionless wall-normal distance $(y^+ = U_\tau y/\nu)$ from the near-wall node to the wall is determined using:

$$y = \frac{1}{2}\frac{\Delta Vol}{A} \tag{3.72}$$

where ΔVol is the near-wall cell volume and A is the area of the cell face in contact with the wall.

The tangential velocity $\mathbf{U}^t$ is aligned to the tangential unit vector, $\hat{\mathbf{t}}$, the direction of which is dependent upon the local velocity field. It is necessary to find an expression for $\hat{\mathbf{t}}$ in terms of the

Cartesian components so that the wall shear force, calculated from ($F_{wall} = \tau_{wall} A$), can be distributed amongst the U, V and W momentum equations. To find $\hat{\mathbf{t}}$ one first obtains an expression for the velocity vector in the wall-normal direction, $\mathbf{U}^n$. This is simply the product of its magnitude, $|\mathbf{U}^n|$, and the wall-normal unit-vector, $\hat{\mathbf{n}}$:

$$\begin{aligned} \mathbf{U}^n &= (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{\mathbf{n}} \\ &= (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_x\hat{\mathbf{i}} + (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_y\hat{\mathbf{j}} + (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_z\hat{\mathbf{k}} \end{aligned} \tag{3.73}$$

The resultant velocity vector, $\mathbf{U}$, is the sum of the wall-normal and wall-parallel velocity components, $\mathbf{U}^n$ and $\mathbf{U}^t$. Therefore, the tangential velocity vector, $\mathbf{U}^t$, is given by:

$$\begin{aligned} \mathbf{U}^t &= \mathbf{U} - \mathbf{U}^n \\ &= \left(U_x\hat{\mathbf{i}} + U_y\hat{\mathbf{j}} + U_z\hat{\mathbf{k}}\right) - \left[(\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_x\hat{\mathbf{i}} + (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_y\hat{\mathbf{j}} + (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_z\hat{\mathbf{k}}\right] \\ &= \left[U_x - (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_x\right]\hat{\mathbf{i}} + \left[U_y - (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_y\right]\hat{\mathbf{j}} + \left[U_z - (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_z\right]\hat{\mathbf{k}} \\ &= U_x^t\hat{\mathbf{i}} + U_y^t\hat{\mathbf{j}} + U_z^t\hat{\mathbf{k}} \end{aligned} \tag{3.74}$$

This vector, $\mathbf{U}^t$, is equal to the product of its magnitude, $|\mathbf{U}^t|$, and the tangential unit vector, $\hat{\mathbf{t}}$, and so one can derive an expression for $\hat{\mathbf{t}}$ from:

$$\begin{aligned} \hat{\mathbf{t}} &= \frac{\mathbf{U}^t}{|\mathbf{U}^t|} \\ &= \frac{1}{|\mathbf{U}^t|}\left(U_x^t\hat{\mathbf{i}} + U_y^t\hat{\mathbf{j}} + U_z^t\hat{\mathbf{k}}\right) \\ &= \hat{t}_x\hat{\mathbf{i}} + \hat{t}_y\hat{\mathbf{j}} + \hat{t}_z\hat{\mathbf{k}} \end{aligned} \tag{3.75}$$

The wall force calculated by the wall function acts in the direction of the unit vector parallel to the wall, $\hat{\mathbf{t}}$, and hence the wall force vector can be expressed:

$$\mathbf{F}_{wall} = \tau_{wall} A\hat{\mathbf{t}} = (\tau_{wall} A)\,\hat{t}_x\hat{\mathbf{i}} + (\tau_{wall} A)\,\hat{t}_y\hat{\mathbf{j}} + (\tau_{wall} A)\,t_z\hat{\mathbf{k}} \tag{3.76}$$

where the component of this force in the Cartesian x-direction, included as a source term in the U-momentum equation, is as follows:

$$s_u = (\tau_{wall} A)\,\hat{t}_x \tag{3.77}$$

and similarly for the V- and W-momentum equations. The components of the tangential unit vector are calculated from:

$$\hat{t}_x = \frac{U_x^t}{|\mathbf{U}^t|} = \frac{\left[U_x - (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_x\right]}{\left(U_x^2 + U_y^2 + U_z^2 - |\mathbf{U}^n|^2\right)^{1/2}} \tag{3.78}$$

$$\hat{t}_y = \frac{U_y^t}{|\mathbf{U}^t|} = \frac{\left[U_y - (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_y\right]}{\left(U_x^2 + U_y^2 + U_z^2 - |\mathbf{U}^n|^2\right)^{1/2}} \tag{3.79}$$

$$\hat{t}_z = \frac{U_z^t}{|\mathbf{U}^t|} = \frac{[U_z - (\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_z]}{\left(U_x^2 + U_y^2 + U_z^2 - |\mathbf{U}^n|^2\right)^{1/2}} \tag{3.80}$$

where the magnitude of the wall-normal velocity vector is:

$$|\mathbf{U}^n| \;=\; \left\{[(\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_x]^2 + [(\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_y]^2 + [(\mathbf{U}\cdot\hat{\mathbf{n}})\,\hat{n}_z]^2\right\}^{1/2} \tag{3.81}$$

and the magnitude of the projection of vector $\mathbf{U}$ in the wall-normal direction $(\mathbf{U}\cdot\hat{\mathbf{n}})$ is given by Equation (3.68).

The above approach assumes that the wall-parallel velocity component at the near-wall node acts in the same direction as the wall shear stress. In a three-dimensional flow, it therefore ignores any skewing of the velocity profile between the near-wall node and the wall. In order to account for skewing of the velocity profile one needs to solve for two independent wall-parallel velocity components. This is the approach used by the UMIST-*N* wall function, which solves for velocity components parallel to the grid lines and obtains two wall-parallel wall shear stress components, τ_{wall}^{ξ} and τ_{wall}^{η} (parallel to the ξ and η grid-lines). The Jacobian matrix $[J]$ is used to transform the wall force components from grid-aligned to Cartesian coordinates:

$$\begin{bmatrix} \tau_{wall}^{x} \\ \tau_{wall}^{y} \\ \tau_{wall}^{z} \end{bmatrix} = \underbrace{\begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix}}_{[J]} \begin{bmatrix} \tau_{wall}^{\xi} \\ \tau_{wall}^{\eta} \\ 0 \end{bmatrix} \tag{3.82}$$

The use of the Jacobian matrix to transform between coordinate systems is discussed in greater detail in Appendix E. The wall force vector is then simply:

$$\mathbf{F}_{wall} = (\tau_{wall}^{x} A)\,\hat{\mathbf{i}} + (\tau_{wall}^{y} A)\,\hat{\mathbf{j}} + (\tau_{wall}^{z} A)\,\hat{\mathbf{k}} \tag{3.83}$$

If a standard wall function is used, this approach of calculating two wall shear stress components, τ_{wall}^{ξ} and τ_{wall}^{η}, can be shown to be identical the use of a single resultant wall shear stress[2]. However, in more sophisticated wall functions, such as UMIST-*A* [61], the shear stress is dependent not only upon the velocity vector at the near-wall node but also on the pressure gradient and convection. In principle, it should therefore be possible to derive an analytical wall function which accounts for skewing of the near-wall velocity profile.

[2]This is because the wall shear stress vector is only a function of the velocity vector at the near-wall node (all other terms are scalars) so that any directional terms vanish in the dimensionless velocity U^+ where one divides velocity by the shear stress.

3.3.5 Pressure on the Wall Surface

When a collocated storage arrangement is used, as in STREAM, it is necessary to set the pressure on the wall surface. This is required in order to calculate the pressure gradient along the grid line that intersects the wall plane which is used in the solution of the momentum equations at the near-wall node[3]. The value of the pressure on the wall is set as follows. If one examines a fully-developed channel flow using a 2-*D* Cartesian grid with zero V-velocity in the wall-normal y-direction, the V-momentum equation simplifies to the following:

$$\frac{\partial P}{\partial y} + \frac{\partial \left(\rho \overline{v^2}\right)}{\partial y} = 0 \tag{3.84}$$

When a low-Reynolds-number turbulence model approach is used, the near-wall node is well within the viscous sublayer and the Reynolds stress at the node is practically zero $\left(\overline{v_p^2} = 0\right)$. The pressure at the wall in this case can be set equal to the pressure at the near-wall node ($P_{wall} = P_P$ and $\partial P/\partial y = 0$). However, when wall functions are used the Reynolds stress at the wall-adjacent node is no longer zero. If one simply set the pressure on the wall equal to the value at the near-wall node ($\partial P/\partial y = 0$), the large gradient in $\overline{v^2}$ from the node to the wall would lead to a spuriously large source term in the V-momentum equation, which would lead to a non-zero V-velocity (whereas, for channel flow, the V-velocity should be zero). In order to avoid this problem the above expression (Equation 3.84) is used to set the wall value of the pressure when wall functions are used. To implement the above condition using a three-dimensional body-fitted grid, the following expression is used:

$$\nabla P \cdot \hat{\mathbf{n}} + (\nabla \cdot \rho \overline{\mathbf{u} \otimes \mathbf{u}}) \cdot \hat{\mathbf{n}} = 0 \tag{3.85}$$

where $\hat{\mathbf{n}}$ is the unit wall-normal vector, $\overline{\mathbf{u} \otimes \mathbf{u}}$ is the Reynolds stress in vector form and the gradient operator, ∇, can be written for non-orthogonal coordinates as:

$$\nabla = \frac{\partial \xi^j}{\partial x^i} \frac{\partial}{\partial \xi^j} \mathbf{e}_i \tag{3.86}$$

where $\mathbf{e}_i$ is the Cartesian unit vector and $\partial \xi^j / \partial x^i$ is a property of the grid. This expression is identical to that used by the UMIST-*N* wall function to determine the pressure gradient across the subgrid cells, as discussed in Chapter 4. The value of the pressure on the wall surface is calculated from the known wall-parallel pressure gradients ($\partial P/\partial \xi$ and $\partial P/\partial \eta$) and the gradients of Reynolds stresses in all three coordinate directions (for details, see Appendix E).

If the wall-parallel gradients of pressure and Reynolds stress are negligible, the calculation of the

[3] In a staggered grid arrangement the velocity is calculated at cell faces and one does not need to use the pressure at the wall surface, only the pressure at the near-wall node. The first wall-normal velocity that is calculated is therefore at the top face of the wall-adjacent cell.

wall-pressure (Equation 3.85) can be simplified to:

$$\frac{\partial P}{\partial n} + \frac{\partial \left(\rho \overline{u_n^2}\right)}{\partial n} = 0 \tag{3.87}$$

where n is the wall-normal direction and $\overline{u_n^2}$ represents the components of the Reynolds stress tensor in the wall-normal direction. The fluctuating velocity component in the wall normal direction is calculated from the scalar product, $u_n = (\mathbf{u} \cdot \hat{\mathbf{n}})$ and hence the normal stress, $\overline{u_n^2}$, is given by:

$$\begin{aligned} \overline{u_n^2} = \overline{(\mathbf{u} \cdot \hat{\mathbf{n}})^2} &= \overline{(u_x \hat{n}_x + u_y \hat{n}_y + u_z \hat{n}_z)^2} \\ &= \overline{u_x^2} \hat{n}_x^2 + \overline{u_y^2} \hat{n}_y^2 + \overline{u_z^2} \hat{n}_z^2 + 2\left(\overline{u_x u_y} \hat{n}_x \hat{n}_y + \overline{u_x u_z} \hat{n}_x \hat{n}_z + \overline{u_y u_z} \hat{n}_y \hat{n}_z\right) \end{aligned} \tag{3.88}$$

To find the pressure on the wall, if one first discretizes Equation (3.87):

$$\frac{\left(P + \rho \overline{u_n^2}\right)_P - \left(P + \rho \overline{u_n^2}\right)_{wall}}{\Delta n} = 0 \tag{3.89}$$

where subscripts $_P$ and $_{wall}$ refer to values at the near-wall node and at the wall and, since the Reynolds stress $\overline{u_n^2}$ is zero on the wall surface:

$$\begin{aligned} P_{wall} &= \left(P + \rho \overline{u_n^2}\right)_P \\ &= \left(P' + \rho \overline{u_n^2} - \frac{2}{3}\rho k\right)_P \end{aligned} \tag{3.90}$$

where $P' = P + 2\rho k/3$ is the pressure stored by the STREAM code and $P'_{wall} = P_{wall}$ as $k_{wall} = 0$. This simplified calculation of the wall-pressure (Equations 3.87 to 3.90) is the standard formulation coded into STREAM and was used in the calculations performed by Robinson [35] for the Ahmed body flow. It is not used in the present study as it was found to introduce instability with the UMIST-*N* wall function. Instead, Equation (3.85) is used, which is consistent with the pressure calculation employed in the UMIST-*N* wall function (Equation 4.34).

The pressure gradient term appearing in the U-momentum equation, $\partial P/\partial x$, can be written using the chain-rule:

$$\frac{\partial P}{\partial x} = \frac{\partial P}{\partial \xi}\frac{\partial \xi}{\partial x} + \frac{\partial P}{\partial \eta}\frac{\partial \eta}{\partial x} + \frac{\partial P}{\partial \zeta}\frac{\partial \zeta}{\partial x} \tag{3.91}$$

The source term due to the pressure gradient which appears in the integrated U-momentum equation can therefore be written:

$$\begin{aligned} S &= -\int_{\Delta Vol} \frac{\partial P}{\partial x} \mathrm{d}Vol \\ &= -\frac{\partial P}{\partial \xi} A_x^\xi - \frac{\partial P}{\partial \eta} A_x^\eta - \frac{\partial P}{\partial \zeta} A_x^\zeta \\ &= -(P_e - P_w) A_x^\xi - (P_n - P_s) A_x^\eta - (P_t - P_b) A_x^\zeta \end{aligned} \tag{3.92}$$

where ΔVol is the cell volume (equivalent to the Jacobian, J), subscripts e, w, n, s, t, b refer to the cell

faces, the cell dimensions in computational space are unity $(\Delta\xi = \Delta\eta = \Delta\zeta = 1)$ and the area vector components are given by:

$$A_x^\xi = J\frac{\partial\xi}{\partial x}\ ;\ A_x^\eta = J\frac{\partial\eta}{\partial x}\ ;\ A_x^\zeta = J\frac{\partial\zeta}{\partial x} \tag{3.93}$$

Physically, A_j^i components are the fraction of the area of the i face which is in the plane orthogonal to the j-direction. For a Cartesian grid in which the ξ^i- and x^j-axes are aligned, $A_1^1 \equiv A_x^\xi$ would be total area of the east face of the cell (since, for this geometry, the east face is orthogonal to the x-axis). The pressure-gradient source term is treated as a combination of separate fluxes through each of the cell faces, where for example the flux through the western face is $\left(P_w A_x^\xi\right)$. Once the pressure on the wall surface has been calculated, using Equation (3.85), the pressure gradient flux term for the wall face is included in the source term in the momentum equation. For the U-momentum equation, with a wall on the bottom face of a cell this is written simply:

$$S_{wall} = P_{wall} A_x^\zeta \tag{3.94}$$

it can be shown[4] that this is equivalent to:

$$S_{wall} = P_{wall} A \hat{n}_x \tag{3.100}$$

where $\hat{n}_x$ is the Cartesian x-direction component of the unit-vector in the wall-normal direction, $\hat{\mathbf{n}}$.

[4]The unit vector acting in the wall-normal direction, $\hat{\mathbf{n}}$, is calculated from:

$$\hat{\mathbf{n}} = \frac{\nabla\zeta}{|\nabla\zeta|} \tag{3.95}$$

where $\zeta = constant$ defines the wall surface. The gradient, $\nabla\zeta$, and its magnitude are given by:

$$\nabla\zeta = \frac{\partial\zeta}{\partial x^i}\mathbf{e}_i = \frac{\partial\zeta}{\partial x}\hat{\mathbf{i}} + \frac{\partial\zeta}{\partial y}\hat{\mathbf{j}} + \frac{\partial\zeta}{\partial z}\hat{\mathbf{k}} \tag{3.96}$$

$$|\nabla\zeta| = \sqrt{\left(\frac{\partial\zeta}{\partial x}\right)^2 + \left(\frac{\partial\zeta}{\partial y}\right)^2 + \left(\frac{\partial\zeta}{\partial z}\right)^2} = \sqrt{g^{33}} = \frac{A}{J} \tag{3.97}$$

where g^{33} is the contravariant metric tensor component acting in the ζ-direction, A is the area of the cell face in the $\xi-\eta$ plane (i.e. the area of the cell face in contact with the wall) and J is the Jacobian which is equivalent to the cell volume. The wall-normal unit vector is then:

$$\begin{aligned}\hat{\mathbf{n}} &= \hat{n}_x\hat{\mathbf{i}} + \hat{n}_y\hat{\mathbf{j}} + \hat{n}_z\hat{\mathbf{k}} \\ &= \frac{J}{A}\left(\frac{\partial\zeta}{\partial x}\hat{\mathbf{i}} + \frac{\partial\zeta}{\partial y}\hat{\mathbf{j}} + \frac{\partial\zeta}{\partial z}\hat{\mathbf{k}}\right)\end{aligned} \tag{3.98}$$

The expression for the source term can therefore be written:

$$S = P_{wall} J\frac{\partial\zeta}{\partial x} = P_{wall} A \hat{n}_x \tag{3.99}$$

3.3.6 Rhie-Chow Interpolation

If a uniform grid is used with a collocated storage arrangement and a linear interpolation is used to find the pressure on cell boundaries, the pressure gradient across a cell is dependent only upon the pressure at the nodes in the surrounding cells and is independent of the pressure in the current cell. This can lead to a pressure field developing in which neighbouring nodes have alternating high-low values. A contour plot of pressure would reveal a checker-board pattern. To prevent this occurring, Rhie & Chow [82] proposed an interpolation for the cell-face velocity (used by the convective fluxes in the momentum equations) which incorporates a pressure-smoothing term.

The discretized U-momentum transport equation can be written:

$$a_P U_P = \sum_{nb} a_{nb} U_{nb} + S_C - J\frac{\partial P}{\partial x} \tag{3.101}$$

which can be rearranged as:

$$U_P = \underbrace{\frac{\sum_{nb} a_{nb} U_{nb}}{a_P} + S_C}_{H_P} - (P_e - P_w)\underbrace{\frac{J\partial\xi/\partial x}{a_P}}_{D_P^U} \tag{3.102}$$

where S_C includes the cross-diffusion terms and the pressure gradient terms parallel to axes other than the x-axis, the Jacobian, J, is equivalent to the cell volume and $\partial\xi/\partial x$ appears from the chain-rule: $\partial P/\partial x = (\partial\xi/\partial x)(\partial P/\partial\xi)$. This formula may also be written:

$$U_P = H_P - D_P^U (P_e - P_w) \tag{3.103}$$

A corresponding expression can be written for the neighbouring node to the east:

$$U_E = H_E - D_E^U (P_{ee} - P_e) \tag{3.104}$$

where subscript *ee* is the eastern face of the cell which has node E at its centre. The velocity at the eastern cell face can also be written:

$$U_e = H_e - D_e^U (P_E - P_P) \tag{3.105}$$

This final expression for the velocity at the eastern face includes the terms H_e and D_e^U. In the Rhie-Chow interpolation these terms are linearly interpolated from the neighbouring P and E values, taken

from Equations (3.103) and (3.104):

$$\begin{aligned} U_e &= \frac{1}{2}\left(H_P + H_E\right) - \frac{1}{2}\left(D_P^U + D_E^U\right)\left(P_E - P_P\right) \\ &= \frac{1}{2}\left[U_P + D_P^U\left(P_e - P_w\right) + U_E + D_E^U\left(P_{ee} - P_e\right)\right] \\ &\quad -\frac{1}{2}\left(D_P^U + D_E^U\right)\left(P_E - P_P\right) \end{aligned} \tag{3.106}$$

which may also be written:

$$\begin{aligned} U_e &= \underbrace{\frac{1}{2}\left(U_P + U_E\right)}_{linear\,interpolation} \\ &\quad + \underbrace{\frac{1}{2}\left[D_P^U\left(P_e - P_w\right) + D_E^U\left(P_{ee} - P_e\right) - \left(D_P^U + D_E^U\right)\left(P_E - P_P\right)\right]}_{pressure\,smoothing} \end{aligned} \tag{3.107}$$

The U-velocity through the eastern face, U_e, can therefore be considered as consisting of two parts: a straight-forward linear interpolation and a smoothing term which is a function of the pressure at the neighbouring nodes. If the *ee* location coincides with a wall boundary, the pressure is extrapolated to the wall surface[5].

Extension of Rhie & Chow Interpolation to Include Normal Stresses

The steady momentum equation in Cartesian coordinates at high Reynolds numbers can be written as follows:

$$\frac{\partial}{\partial x_j}\left(\rho U_i U_j\right) = -\frac{\partial P}{\partial x_i} - \frac{\partial}{\partial x_j}\left(\rho\overline{u_i u_j}\right) \tag{3.108}$$

Taking the trace of this expression $(i = j)$, the pressure and Reynolds stress terms are both of the same form, i.e. expressing the gradient of a scalar. It is therefore possible to extend the Rhie-Chow interpolation to include gradients of the normal stresses with the pressure gradient. The discretized U-momentum equation is re-written:

$$U_P = \underbrace{\frac{\sum_{nb} a_{nb} U_{nb}}{a_P} + S_C}_{H_P} - \left(P_e - P_w\right)\underbrace{\frac{J\partial\xi/\partial x}{a_P}}_{D_P^U} - \left(\rho\overline{u^2}_e - \rho\overline{u^2}_w\right)\underbrace{\frac{J\partial\xi/\partial x}{a_P}}_{D_P^U} \tag{3.109}$$

where S_C now excludes the gradient parallel to the x-axis of both pressure and the $\overline{u^2}$ stress. The analysis given above, from Equations (3.103) to (3.107) can be repeated, substituting P with $\left(P + \rho\overline{u^2}\right)$.

[5]The pressure at the wall surface, calculated instead from Equation (3.85), was found to cause instability in some calculations and was therefore not used.

This results in the following expression for the U-velocity through the eastern face of a cell:

$$\begin{aligned} U_e &= \frac{1}{2}(U_P + U_E) \\ &\quad + \frac{1}{2}\left\{ D_P^U \left[\left(P + \rho\overline{u^2}\right)_e - \left(P + \rho\overline{u^2}\right)_w \right] \right. \\ &\quad + D_E^U \left[\left(P + \rho\overline{u^2}\right)_{ee} - \left(P + \rho\overline{u^2}\right)_e \right] \\ &\quad \left. - \left(D_P^U + D_E^U\right) \left[\left(P + \rho\overline{u^2}\right)_E - \left(P + \rho\overline{u^2}\right)_P \right] \right\} \end{aligned} \tag{3.110}$$

Similarly one can find the V-velocity through the cell faces using $\left(P + \rho\overline{v^2}\right)$, and for the W- velocity $\left(P + \rho\overline{w^2}\right)$. This correction can improve the stability of a calculation in which there are steep gradients in the normal Reynolds stresses.

3.3.7 Convergence Criteria

In STREAM, the dimensionless residual is calculated from:

$$R_\phi = \phi_P - \frac{\sum_{nb} a_{nb}\phi_{nb} + S}{a_P} \tag{3.111}$$

where *nb* indicates the neighbouring nodes (E, W, N, S, T and B) and the velocity component or turbulence scalar, ϕ, is dimensionless (as discussed in Section 3.3.2). A mass imbalance is calculated instead of a pressure residual. The convergence criteria is specified using the RMS residual which is calculated as follows:

$$R_\phi^{RMS} = \sqrt{\frac{1}{n}\sum_n R_\phi^2} \tag{3.112}$$

where n is the total number of nodes. The RMS residual tends to give a smaller value than the expressions used in the TEAM code (Section 3.2.4). The quantity, R_ϕ^{RMS} is also independent of the grid size (i.e. number of nodes) which allows the relative degree of convergence between different calculations to be assessed.