Appendix G

Other Wall Function Options Explored

G.1 Subgrid Storage Requirements

Storage requirements of the UMIST-N wall function are similar to those of a full low-Re model approach: values of velocity, turbulence parameters and temperature (if a thermal field is being solved) are stored in each subgrid cell along the length of the wall[1]. Clearly, it would be advantageous to reduce this storage demand. Saving subgrid values is necessary for convection parallel to the wall, calculation of the wall-normal W-velocity and initialization of the subgrid values at each iteration. These three components of the UMIST-N wall function are discussed below.

G.1.1 Wall-Parallel Convection

In the UMIST-N wall function, subgrid convection parallel to the wall is approximated using an upwind differencing scheme which uses upstream subgrid values. Therefore upstream values of velocity, turbulence parameters and temperature must be stored. For practical application of the wall function to complex flows in which one does not have prior knowledge of the flow direction, or in which there is separation and reattachment, this requires storage of subgrid values along the whole length of the wall. An alternative scheme was tested during the development of the UMIST-N wall function which involved using main-grid values for wall-parallel convection. For a wall which is parallel to the x-axis, in Cartesian coordinates, the wall-parallel convection of scalar, ϕ, was approximated as:

$$\rho U \frac{\partial \phi}{\partial x} = \rho U_P \frac{(\phi'_e - \phi'_w)}{\Delta x} \underbrace{\frac{\phi_o}{\overline{\phi}}} \tag{G.1}$$

[1]In fact, the storage demands are likely to be slightly less than a full low-Re model approach since the subgrid pressure distribution does not need to be stored and there is effectively a discontinuity in the cell size due to the use of an embedded grid within the near-wall main-grid cell.

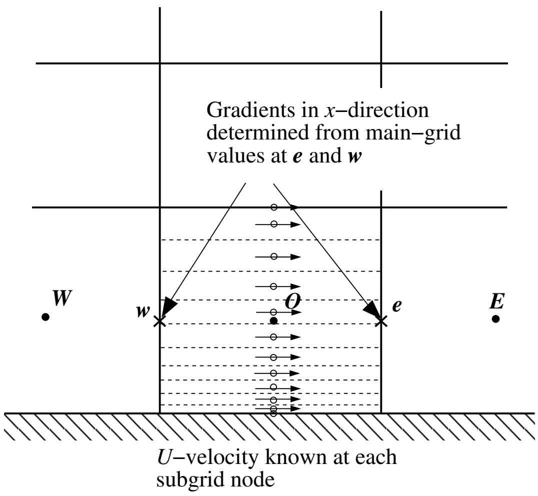

Figure G.1: Calculation of subgrid convection parallel to the wall using main-grid node values

where ϕ represents the wall-parallel velocity component (U), turbulence parameters (k or $\tilde{\varepsilon}$) or temperature (T), ϕ'_e is the main-grid value of ϕ at the eastern boundary of the subgrid, determined from a linear interpolation between main-grid nodes E and O, see Figure G.1, and subscript P refers to the current subgrid node. Rather than assume that the gradients of ϕ are constant across the whole subgrid region a scaling factor is introduced (the underbraced term in Equation G.1), based on the ratio of the main-grid nodal value (ϕ_o) to the average subgrid value ($\overline{\phi}$). This convection scheme was tested in the impinging jet flow in which the average subgrid value $\overline{\phi}$ was non-zero for all of the parameters: U, k, $\tilde{\varepsilon}$ and T. A modified scaling function or some limit would need to be introduced to prevent unrealistic values occurring when $\overline{\phi} \to 0$, for example in situations where flow reversal takes place across the subgrid (so that the average velocity, $\overline{U}$, is close to zero). The above wall-parallel convection term was integrated over the subgrid cell and placed in the discretized subgrid equation source term. The convection source term in the momentum equation and in the k and $\tilde{\varepsilon}$ equations was linearized (by splitting the source into $S = s_U + s_P\phi_P$) to maximize stability. Central differencing was used to discretize the main-grid gradient term in Equation (G.1). In tests with the impinging jet flow, there was no change in stability from switching to an upwind scheme and results using the two schemes were identical.

It was found in the impinging jet flow that using scaled main-grid-values for wall-parallel convection (Equation G.1) led to instabilities early in the solution process when there were abrupt changes in the mass flux between neighbouring cells. To obtain a stable solution with a linear $k-\varepsilon$ model, subgrid convection of momentum, and in some cases convection of k and $\tilde{\varepsilon}$, was only activated once the total mass residual fell below a threshold value (in the impinging jet case, this was when the normalized mass residual fell to approximately 5×10^{-4}, which was achieved in roughly one-third of the total computational time). In addition, it was found that for moderately large near-wall main-grid cell sizes ($y^+ > 250$) some under-relaxation of the subgrid momentum equations was required (typically, a factor of $\alpha = 0.9$ was used). When the cubic non-linear EVM of Craft *et al.* [67] was used the above convection treatment was found to be highly unstable. No amount of under-relaxation could stabilize the numerical solution for large near-wall cells.

This behaviour contrasts to that observed using the recommended approach of saving subgrid values along the wall and using these to calculate convection (as described in Section 4). Using the recommended approach, it is not necessary to wait until the flow-field has settled down before activating subgrid convection. No under-relaxation of the subgrid momentum equations is required (even for large near-wall main-grid cells). One is also able to capture changes in the sign of the gradients across the subgrid, for instance where the k-gradient parallel to the wall ($\partial k/\partial x$) is positive at the outer edge of subgrid but negative near wall. In the scaled main-grid approach the sign of the gradient across the whole of the subgrid is determined by the main-grid gradient (i.e. $\partial k/\partial x$ would be the same sign across the whole subgrid). Whilst this, in itself, did not seem to be source of the instability problems encountered with the scaled main-grid convection scheme, it was observed to lead to "wiggles" in the wall shear stress and Nusselt number profiles in the impinging jet flow.

G.1.2 Calculation of the Subgrid Wall-Normal Velocity

In the UMIST-N wall function, the subgrid wall-normal V-velocity is calculated from continuity within each of the subgrid control volumes and then the resulting subgrid V-profile is scaled in order to match the main-grid boundary condition (as described in Chapter 4). In order to interpolate subgrid velocities to the cell faces, which are used in the discretized continuity equation, one needs to store the subgrid velocities in the neighbouring subgrid nodes (which, practically, involves storing the subgrid velocity in each subgrid cell along the length of the wall). An alternative approach is simply to prescribe a V-velocity profile across the subgrid. Low-Reynolds-number model results for the impinging jet flow showed that the V-velocity varies almost linearly across the subgrid region (see Figure 4.6 on Page 81). As a first approximation, one could use a linear V-velocity profile, with the boundary values of $V = 0$ at the wall (assuming the wall to be non-porous) and, at the opposite face, the main-grid V-velocity boundary condition (as shown in Figure G.2). However, the gradient of a linear V-velocity profile in the wall-normal direction would then be constant ($\partial V/\partial y = constant$), whereas continuity implies that $\partial V/\partial y \rightarrow 0$ at the wall, since $\partial U/\partial x = 0$. In order to satisfy this constraint one can either introduce near-wall damping of the V-velocity, or one can assume a non-linear profile across the whole subgrid. The latter approach was found to be more successful in the impinging jet flow, where the following power law was assumed:

$$V = ay^{1.1} \tag{G.2}$$

The term a is calculated from $a = V'_n / y_n^{1.1}$ in order to satisfy the boundary conditions, where V'_n is the main-grid velocity at the outer subgrid boundary ($y = y_n$). This prescription was close enough to a linear profile to agree well with low-Reynolds-number model predictions in the impinging jet flow, whilst satisfying the condition that $\partial V / \partial y \rightarrow 0$ as $y \rightarrow 0$.

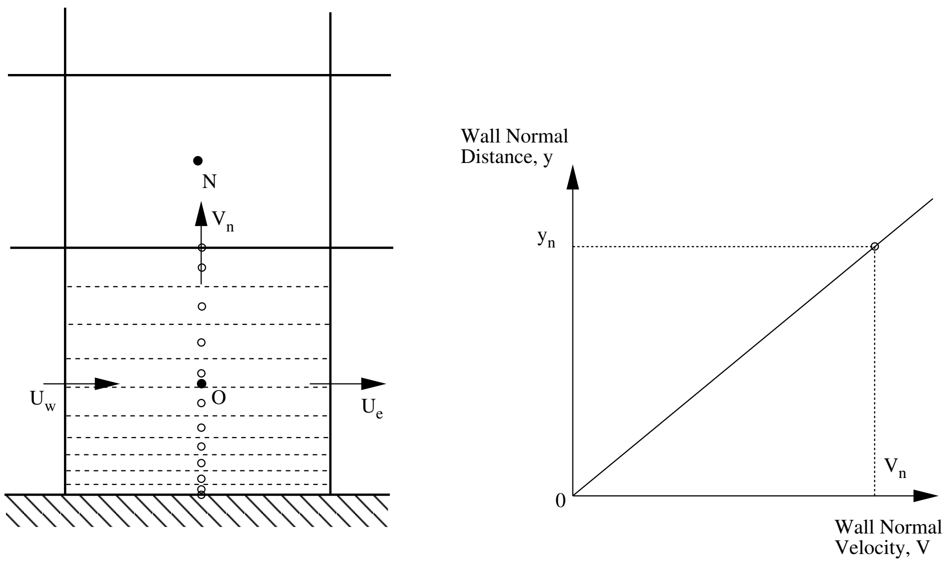

Figure G.2: Calculation of the subgrid wall-normal V-velocity

Nusselt number profiles for the impinging jet flow using the prescribed subgrid wall-normal V-velocity profile are shown in Figures G.3 and G.4, using the linear and non-linear $k - \varepsilon$ models, respectively. These can be compared to Figures 5.20 and 5.46 which used the same models with the recommended scaled-continuity approach to calculate the subgrid V-velocity (as presented in Chapter 4). With the linear model, a slightly higher Nusselt number is predicted with the prescribed V-velocity approach near the stagnation point ($r/D < 2$) than with the scaled-continuity approach. The prescribed velocity profile result is therefore in slightly better agreement with the low-Re model prediction. However, with the non-linear model, the prescribed V-velocity approach overpredicts the low-Re model Nusselt number near the stagnation point ($r/D < 2$) by around 6% and shows some sensitivity to the near-wall cell size. In comparison, the scaled-continuity approach gave results in excellent agreement with the low-Re model and showed practically no sensitivity to the near-wall cell size. Modest changes in the prescribed subgrid V-profile were also found to lead to significant changes in the predicted Nusselt number in the impinging jet flow. Whilst prescribing a V-velocity profile is a conceptually simple approach which is easy to implement and does not require additional storage, the recommended practice of calculating V from subgrid continuity removes

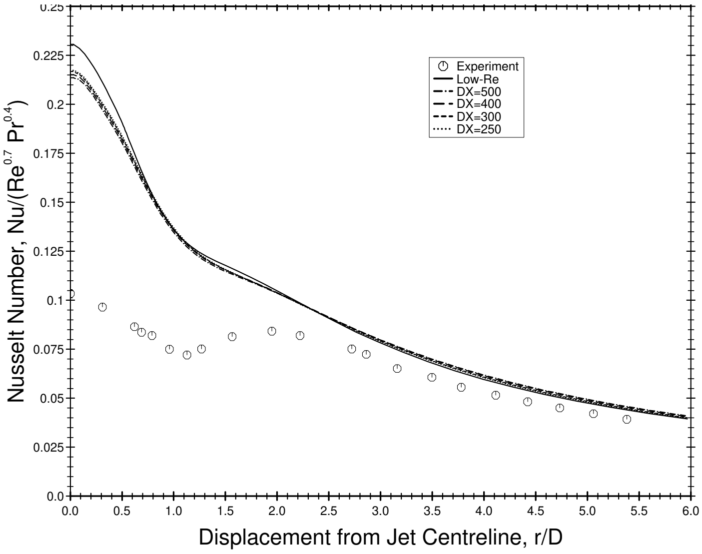

Figure G.3: Nusselt number profiles obtained for the impinging jet flow using the UMIST-N wall function with the saved-subgrid convection scheme parallel to the wall (Equation 4.62), prescribed wall-normal velocity profile (Equation G.2), linear $k-\varepsilon$ model and standard Yap correction. Solid line: low-Re Launder-Sharma model; broken lines: wall function results for different near-wall cell widths; symbols: experiments of Baughn *et al.* [94].

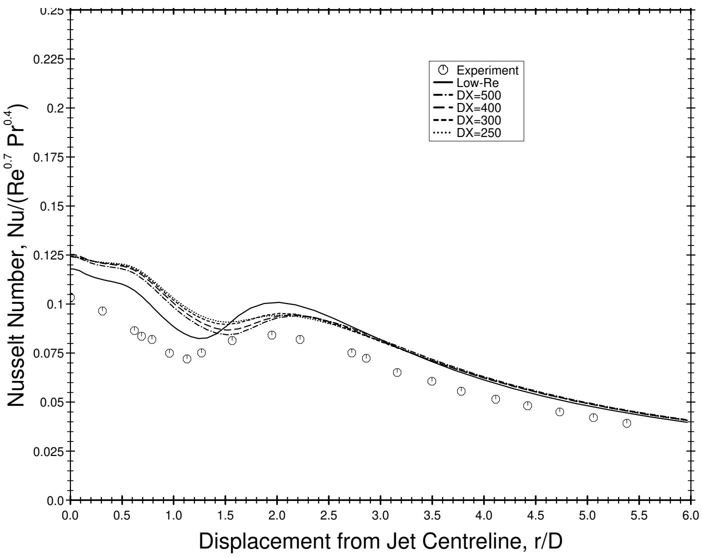

Figure G.4: Nusselt number profiles obtained using the subgrid wall function with the NLEVM of Craft *et al.* [67], differential Yap correction, saved-subgrid convection scheme parallel to the wall (Equation 4.62) and prescribed wall-normal velocity profile (Equation G.2). Solid line: low-Reynolds-number model; broken lines: wall function results for different near-wall cell widths; symbols: experiments of Baughn *et al.* [94].

the reliance of the UMIST-N wall function on any assumed profiles, which should make the wall function applicable to a wider range of flows.

Another approach to finding the subgrid V-velocity which was tested in the impinging jet flow involved the solution of a simplified V-velocity transport equation, which was derived using the same assumptions that were used to obtain the wall-parallel U-velocity equation (see Chapter 4). This approach was found to be extremely sensitive to the assumed pressure profile within the subgrid. One cannot obtain a pressure profile across the subgrid by solving a pressure-correction equation as this would introduce additional coupling and lead to computing times for the subgrid wall function approaching that of a full low-Reynolds-number model solution.

G.1.3 Initialization of Subgrid Values

In the UMIST-N wall function, all the subgrid values of velocity, turbulence parameters and temperature are stored along the length of the wall. This enables the values from the previous subgrid iteration to be used as initial conditions for the current iteration. If an alternative treatment is employed in which subgrid values are not stored, the initial conditions for a subgrid calculation can be taken from neighbouring subgrid values, since the subgrid calculation proceeds sequentially along the wall (see Figure G.5). One can also scale the adjoining subgrid cell values by a factor corresponding to the difference in neighbouring main-grid nodal values. This approach was tested in the impinging jet flow. The scaling factor applied to the subgrid profile at the previous location to provide initial conditions for the current subgrid solution was based on the difference between the previous and the current subgrid boundary conditions. In addition to starting from more realistic values, this scaling removed the possibility of a discontinuity at the outer edge of the subgrid, which may have otherwise occurred due to the updated boundary conditions. If the current nodal position is assumed to be node j and the previous position node $(j-1)$ then the factor with which the previous subgrid profile was scaled, is given by:

$$\frac{\phi_j^b}{\phi_{j-1}^b} \tag{G.3}$$

where the superscript b refers to the subgrid boundary condition for ϕ at the j or $(j-1)$ node. This scaling was applied to subgrid profiles for U, k, $\tilde{\varepsilon}$ and T in the impinging jet flow. The scaling factor may need to be modified in other flows if the denominator in the above expression, ϕ_{j-1}^b, was close to zero, for example in situations involving flow reversal (where velocity U_{j-1}^b may be zero).

The problem with starting the subgrid calculation using initial values scaled from a neighbouring subgrid cell is that one must perform a number of iterations of the subgrid calculation per main-grid iteration in order to obtain a converged solution. In the impinging jet

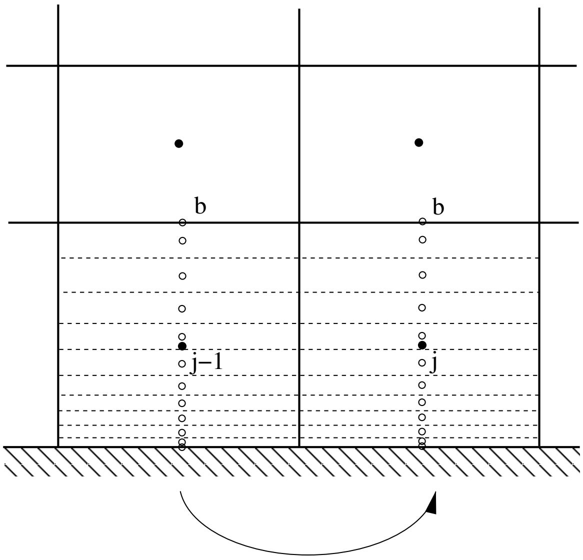

Figure G.5: Updating subgrid nodal values for initial conditions

flow, around 5 subgrid iterations were necessary for each main-grid iteration which led to an increase in the overall computing time of 42% compared to the saved-subgrid approach which used one subgrid iteration per main-grid iteration.

G.1.4 Summary

There are three main reasons why the UMIST-N wall function saves subgrid values of velocity, turbulence parameters and temperature (if a thermal field is being solved) at each subgrid node along the length of the wall. Firstly, the alternative convection treatment involving main-grid values is less numerically stable. Secondly, using a prescribed wall-normal velocity profile is likely to be less generally applicable and has been shown to give poorer results in the impinging jet flow with the NLEVM. Thirdly, the subgrid calculation cannot be initialized accurately without storing subgrid values. If subgrid values are not stored a number of subgrid iterations must be performed per main-grid iteration, which leads to prolonged computing times.

In addition to these considerations, if one is storing the subgrid parameters at each subgrid node, one is able to calculate all the components of the production term in the turbulent kinetic energy equation (which requires velocity gradients parallel to the wall). When a non-orthogonal grid arrangement is used, one is also able to calculate the subgrid pressure gradient $\partial P/\partial\zeta$ from gradients in the Reynolds stresses (as described in Chapter 4).

G.2 Convection Treatment in Curvilinear Coordinates

The convection term in the momentum equations for non-orthogonal curvilinear coordinates, derived in Appendix C, is given by:

$$(\mathbf{U} \cdot \nabla)\,\mathbf{U} = \left(\frac{U^{(j)}}{\sqrt{g_{jj}}} \frac{\partial U^{(i)}}{\partial \xi^j} - U^{(i)} U^{(j)} \frac{\Gamma^m_{ij} g_{im}}{g_{ii}\sqrt{g_{jj}}} + U^{(j)} U^{(m)} \frac{\Gamma^i_{mj}\sqrt{g_{ii}}}{\sqrt{g_{jj} g_{mm}}} \right) \mathbf{g}_{(i)} \tag{G.4}$$

In Appendix D, an alternative approach to calculating convection was recommended, rather than expand the lengthy expression given above. This involved the transformation of the velocity components in the upstream cell (assuming an upwind convection scheme is employed) from the coordinate system used in the upstream cell into the coordinate system used in the current cell. This gave the following expression:

$$(\mathbf{U} \cdot \nabla)\,\mathbf{U} = \frac{U^{(j)}}{\sqrt{g_{jj}}} \left(\frac{\partial U^{(i)}}{\partial \xi^j} \right)^{*} \mathbf{g}_{(i)} \tag{G.5}$$

The asterisk is introduced around the velocity gradient term to denote that upstream values of $U^{(i)}$ are transformed into the coordinate system used in the current cell.

Both of the above convection treatments were tested in the Ahmed body flow, discussed in Chapter 7. Equation (G.4) was found to lead to numerical instability in regions where the grid was highly skewed, such as the 90° corner between the base and the underside of the car (where, at the apex of the corner, the grid was skewed at 45° to the wall). The alternative and considerably simpler convection treatment, given by Equation (G.5), was not found to cause stability problems. The precise cause of the convergence problem with Equation (G.4) was not traced. Analysis of the flow around the 90° rear corner of the Ahmed body becomes very complex once one has expanded all the geometric terms. It may be that problems were introduced by using an upwind scheme for the velocity gradient term in Equation (G.4) whilst using central differencing for the gradients of the metric tensors (used in the Christoffel symbols). Coding errors cannot be ruled out, especially when dealing with such lengthy expressions, although every effort was made to ensure that the equations were coded correctly[2]. In order to investigate the matter further it would be easier to study a simpler flow than the Ahmed body, such as a two-dimensional backward-facing step flow. In such a test-case, one could examine the effects of using a contoured grid as opposed to a Cartesian grid (as shown in Figure G.6), the code would run significantly faster and one would not have to consider the effects of the convection treatment upstream of the corner (as one does for the Ahmed car body). Whilst it would be useful to find out why the convection treatment of

[2]Channel flows were examined using the same code with grids skewed in different planes to ensure that the geometric parameters were calculated correctly. In addition, the code was run using several different FORTRAN compilers using various debugging options to see if any array boundaries were exceeded or values used before they were initialized etc.

Equation (G.4) was numerically unstable, the alternative treatment given by Equation (G.5) is to be preferred since it is conceptually simpler, easier to code and has been shown to be more robust.

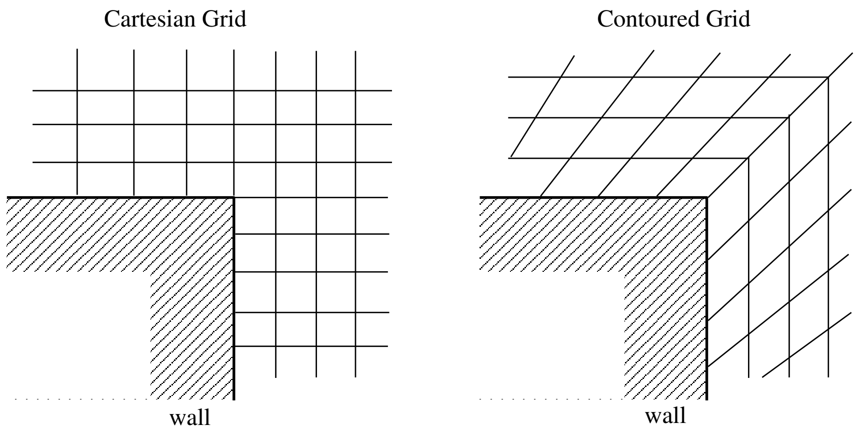

Figure G.6: Different grid arrangements for a 90° corner