

Evaluation of numerical wall functions on the axisymmetric impinging jet using OpenFOAM



J.-A. Bäcker^{a,*}, L. Davidson^b

^a Department of Energy Conversion and Physics, Volvo Group Truck Technology, Gothenburg SE-405 08, Sweden

^b Division of Fluid Dynamics, Department of Applied Mechanics, Chalmers University of Technology, Gothenburg SE-412 96, Sweden

ARTICLE INFO

Article history:

Received 16 December 2016

Revised 3 July 2017

Accepted 4 July 2017

Available online 24 July 2017

Keywords:

Computational fluid dynamics

OpenFOAM

Impinging jet

Near-wall

Numerical wall function

Speed-up

ABSTRACT

Two new robust numerical wall functions are evaluated and the effect of different approximations used in earlier numerical wall functions by Craft et al. (2004) and by Bond and Blottner (2011) are demonstrated. A standard low-Reynolds-number turbulence (LRN) model is used as reference but with different meshing strategies. The objective is to considerably reduce the total central processing unit (CPU) cost of the numerical simulations of wall bounded flows while maintaining the accuracy of any LRN model.

When calculating turbulent flow problems, a tremendous speed-up may be achieved by decoupling the solution of the boundary layer from the bulk region by using a *wall function*. However, most wall functions are quite limited and based on assumptions which are not valid in complex, non-equilibrium flows.

The present wall functions solve full momentum and energy equations on a sub-grid, using face fluxes of advection and diffusion to transfer the solution to and from the sub-grid. The evaluation was carried out on an axisymmetric impinging jet using the turbulence model of Launder and Sharma (1974) with the correction of Yap (1987). Compared to standard LRN calculations, the results show perfect agreement to less than one-sixth of the computational cost. However, the reason for the speed-up is shown to come mainly from the meshing strategy, and none of the evaluated wall functions add much additional value.

© 2017 The Authors. Published by Elsevier Inc.

This is an open access article under the CC BY-NC-ND license.

(<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

1. Introduction

The use of CFD continues to increase in industry, due to the savings that can be achieved in both time and cost over corresponding experiments. To predict industrial flow problems, which often have a turbulent nature, the most common approach is still a Reynolds Averaged Navier Stokes (RANS) simulation together with a turbulence model. Considering accuracy and computational cost for a certain class of flows, dominated by boundary layer effects, the most important aspect of such simulations is how the boundary layer is treated.

The boundary layer is the fluid layer in the immediate vicinity of a wall, in other words, where the viscous effect is not negligible. It extends to the fully turbulent regime and, even though it only occupies a smaller part of the flow, this region may account for the majority of the computing time. The reason for this relatively high computational cost is that boundary layer flow properties change

at a rate typically two or more orders of magnitude faster than elsewhere in the flow.

These high gradients require a very fine computational mesh in order to be resolved accurately. The family of turbulence models that uses this strategy of resolving the boundary layer is called low-Reynolds-number (LRN) models. These models use the same set of equations for all parts of the flow and may be accurate for most types of flows, but the resulting equation system converges slowly, especially at high Reynolds number. The turbulence models span from simple mixing-length schemes, through two-equation eddy-viscosity models of different complexity, to second-moment closure models.

To mitigate the slow convergence of the LRN models, the boundary layer and the fully turbulent region may be decoupled, thus acknowledging the different computational requirements for the two regions. The most common approach is the high-Reynolds-number (HRN) model together with a “wall function”, which uses a coarse mesh where the first cell layer covers the inner boundary layer, including the inner part of the log-layer. Instead of solving partial differential equations on a fine mesh, an analytical expression is used to model the flow in the boundary layer. HRN mod-

* Corresponding author.

E-mail address: jon-anders.backar@chalmers.se (J.-A. Bäcker).

els with wall functions are often less accurate, and they are also sensitive to the mesh resolution close to the wall. Attempts have also been made to analytically integrate the transport equations, but these give restrictions on the geometry to allow for analytical integration.

A more advanced way of decoupling the boundary layer from the fully turbulent region is to use a “numerical wall function”. This wall treatment can be seen as a hybrid of HRN and LRN modelling where the boundary layer is indeed resolved but with a slightly simplified set of partial differential equations compared to what is used in the rest of the domain.

The first numerical wall function in a RANS context, called UMIST-N, was developed by Gant (2002) and Craft et al. (2004). They divided the wall-adjacent cells into a sub-grid where simplified RANS equations, using some sort of boundary layer assumptions, were solved, including tangential velocity and turbulent quantities. Furthermore, the pressure gradient was assumed to be constant in the wall-normal direction over the sub-grid and could hence be interpolated from the main-grid. A Dirichlet condition, with interpolated values from the main-grid’s first and second wall-adjacent cells, was set on the boundary of the sub-grid, opposite to the wall, for all solved quantities. The calculated wall shear stress, averaged turbulent production and dissipation terms from the sub-grid were then used to replace the corresponding terms in the main-grid equations. This yielded results close to a default LRN solution at computing times of an order less in magnitude.

A few studies have investigated variations of the UMIST-N model. Myers and Walters (2005) simplified the sub-grid equations even further by using a linear profile for the wall-normal velocity and used the 2-D continuity equation to calculate the stream-wise velocity gradient. The convection was neglected in the turbulence equations. In this way, the 2-D boundary layer equations were reduced to 1-D equations for the tangential velocity and turbulent model quantities. Bond and Blottner (2011) proposed a similar model for compressible and transient flow by neglecting convection in all transport equations. Chedevergne (2010) also developed a similar 1-D model but implemented it in an unstructured code where the sub-grid only covered the main-grid’s wall-adjacent cells from the wall up to the centroid of the main cells. He also included compressibility terms in the model equations. Lastly, Wald (2016) tried to adapt the UMIST-N for a second-moment closure turbulence model which gave similar results in accuracy as Craft et al. reported earlier on an axisymmetric impinging jet. However, Wald (2016) also concludes that the model is unstable and chooses not to pursue with other geometries. It is not clear from his thesis whether the robustness issues arise from the use of UMIST-N itself or only in combination with the turbulence model used.

Even though the processing speed of computers is continuously increasing, the CFD community is generally far from satisfied with available computing resources, regardless of whether they act in industry or elsewhere. As e.g. Spalart (2000) describes, HRN and LRN modelling belong to the simpler variants of methods that solve turbulent transport equations. Nevertheless, with the use of these relatively simple models for large and complex problems, the computational resources often set a limit to what can be done. If the same models are used in design-of-experiments or optimisation loops, the computing resources will obviously always be a limitation to what can be achieved with simulations for the next decades.

With this background, it is important to acknowledge and deploy turbulence modelling techniques that offer the best compromise between accuracy and computing requirements. The numerical wall function strategy deployed in RANS modelling has existed since at least 2004 but has not yet been widely adopted by the CFD community despite its excellent features of supplying a sweet-spot between HRN and LRN modelling. The most important reasons for

this are probably the cost of implementation and the close connection to the turbulence model. To support a turbulence model, earlier numerical wall functions need to implement each model’s specific source and sink terms, making the implementation and maintenance more awkward.

The purpose of this investigation is twofold: first to make an implementation in an openly available and unstructured CFD code and relax the dependence between the implementation and specific supported turbulence models.¹ The second purpose is to evaluate different near-wall strategies including commonly used assumptions in earlier numerical wall functions.

Two new numerical wall functions are built upon the work from Craft et al., but they use a more mathematically stringent coupling which is independent on choice of turbulence model. This has been achieved by an innovative use of face fluxes, making a two-way connection between the main-grid and the sub-grid. These new numerical wall functions are evaluated on a turbulent axisymmetric impinging jet with and without assumptions made in earlier numerical wall functions, but also with standard integration to the wall using a similar mesh cell distribution which is normally used in numerical wall functions. It is found that an advanced mesh strategy gives a similar speed-up as a decoupled approach, i.e. the numerical wall function, and we demonstrate that it is also the most robust alternative. Thus, in this study, no added value is found for the concept of the numerical wall function.

2. Method

The effect of different meshing strategies is first investigated using a standard wall treatment, i.e. integration to the wall, used with LRN turbulence models. Second, the implementation of the robust wall functions is verified. Third, the new wall functions are evaluated regarding their sensitivity with respect to how far the interface is placed from the wall in y^+ units. Last, the effect of the assumptions in earlier numerical wall functions Craft et al. (2004) and Bond and Blottner (2011) are compared with the new robust wall functions and to a standard LRN set-up deploying the same local mesh density as with the numerical wall functions.

All tests and implementations have been done in OpenFOAM® (2015), Open Field Operation and Manipulation, CFD Toolbox, which is a free and open source CFD software package. It uses a co-located methodology on unstructured polyhedral meshes. This methodology is used in both the main-grid and the sub-grid of the numerical wall functions. However, a restriction is introduced for the wall-adjacent cells in the main-grid to be prismatic.

2.1. Governing equations

The full 3D-RANS equations are solved in both the main-grid and the sub-grid, with the assumption that the pressure gradient in the wall-normal direction is constant in the sub-grid. Incompressible Reynolds averaged Navier–Stokes in tensor notation reads

$$\bar{u}_{i,i} = 0, \quad (1)$$

$$\bar{u}_{i,t} + (\bar{u}_i \bar{u}_j)_{,j} = -\frac{\bar{p}_{,i}}{\rho} + \left[(v + v_t)(\bar{u}_{i,j} + \bar{u}_{j,i}) \right]_{,j} \quad (2)$$

$$T_{,t} + (\bar{u}_i T)_{,i} = \left[\left(\frac{v}{\sigma} + \frac{v_t}{\sigma_t} \right) T_{,i} \right]_{,i} \quad (3)$$

with the LRN model of Launder and Sharma (1974) including the correction of Yap (1987),

$$k_{,t} + (\bar{u}_i k)_{,i} = \left[\left(v + \frac{v_t}{\sigma_k} \right) k_{,i} \right]_{,i} + P_k - \tilde{\epsilon} - D \quad (4)$$

¹ A cleaned up version of the implementation is planned to be published under <https://github.com/backar>.

$$\tilde{\varepsilon}_{,t} + (\bar{u}_i \tilde{\varepsilon})_{,i} = \left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon} \right) \tilde{\varepsilon}_{,i} \right]_{,i} + C_{\varepsilon_1} P_k \tilde{\varepsilon} / k - C_{\varepsilon_2} f_2 \tilde{\varepsilon}^2 / k + E + S_\varepsilon \quad (5)$$

where²

$$\begin{aligned} P_k &= \nu_t S^2, \quad S \equiv \sqrt{2S_{ij}S_{ij}}, \quad S_{ij} = 0.5(\bar{u}_{j,i} + \bar{u}_{i,j}), \\ \sigma_k &= 1.0, \quad \sigma_\varepsilon = 1.3, \quad C_{\varepsilon_1} = 1.44, \quad C_{\varepsilon_2} = 1.92, \\ f_2 &= -0.3 \exp(-\min[50, k^2 \tilde{\varepsilon} / \nu]), \\ D &= 2\nu \left[(\sqrt{k})_{,i} \right]^2, \quad E = 2\nu \nu_t (\bar{u}_{i,jk})^2. \end{aligned} \quad (6)$$

The Yap correction includes the normal distance to the nearest wall, y_n ,

$$S_\varepsilon = 0.83 \frac{\varepsilon^2}{k} \left(\frac{k^{1.5}}{\varepsilon l_e} - 1 \right) \left(\frac{k^{1.5}}{\varepsilon l_e} \right)^2, \quad l_e = c_\mu^{-0.75} \kappa y_n \quad (7)$$

and the turbulent viscosity is expressed as

$$\nu_t = C_\mu f_\mu \frac{k^2}{\varepsilon}, \quad C_\mu = 0.09, \quad f_\mu = \exp\left(\frac{-3.4}{1 + k^2 \tilde{\varepsilon} / (50\nu)}\right). \quad (8)$$

The standard HRN model $k - \varepsilon$ reads:

$$k_{,t} + (\bar{u}_i k)_{,i} = \left[\left(\nu + \frac{\nu_t}{\sigma_k} \right) k_{,i} \right]_{,i} + P_k - \varepsilon \quad (9)$$

$$\varepsilon_{,t} + (\bar{u}_i \varepsilon)_{,i} = \left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon} \right) \varepsilon_{,i} \right]_{,i} + C_{\varepsilon_1} P_k \varepsilon / k - C_{\varepsilon_2} \varepsilon^2 / k \quad (10)$$

with the turbulent viscosity as $\nu_t = C_\mu \frac{k^2}{\varepsilon}$ and constants $\sigma_k, \sigma_\varepsilon, C_{\varepsilon_1}, C_{\varepsilon_2}$ as earlier defined. Together with the HRN model, a standard wall function is used which sets the value, in the wall adjacent cells, of the turbulent dissipation and the turbulent production³

$$\begin{aligned} \varepsilon &= C_\mu^{3/4} k^{3/2} / (\kappa y_n) \\ P_k &= (\nu + \nu_t) |\nabla_{N_w} U| C_\mu^{1/4} \sqrt{k} / (\kappa y_n) \end{aligned} \quad (11)$$

where $\nabla_{N_w} U$ is the wall-normal gradient of the velocity, $\kappa = 0.41$, and y_n is, as before, the near-wall distance.

2.2. Mesh strategies

The difference between typical near-wall mesh strategies in HRN and LRN modelling is the size of the near-wall cells, see Fig. 1a and b. The cell sizes are generally non-uniform and increase in size with increasing distance to the wall using a simple geometric expansion. It is commonly good practice to use a constant and sufficiently small mesh expansion ratio, resulting in a change in mesh spacing, which is continuous. When numerical wall functions have been used, a sub-grid has been added to an HRN mesh (Craft et al., 2004; Myers and Walters, 2005; Bond and Blottner, 2011; Chedeveigne, 2010) to be able to resolve the turbulent boundary layer as in LRN modelling. By doing this, a distinct discontinuity in cell size appears between the sub-grid cell farthest from the wall and the second main-grid cell. To mitigate this discontinuity, this layer of the main-grid cells adjacent to the sub-grid can be refined. All in all, deploying this strategy in a monolithic mesh would create an AMS, Advanced Meshing Strategy, mesh having

three layers with different cell expansion ratios, see Fig. 1c, that can be optimised individually. Normally, CFD software allows the user to adapt the value of r_w and recommend a value typically less than 1.2 and emphasize that this region should cover the whole boundary layer. Here we set the value of $r_w = 1.07$ to ensure that discretization errors are small in this region containing very high gradients. This seems to be especially important for the gradient of turbulent dissipation, ε which may be huge close to the wall. To optimize for speed-up we use a relatively high cell expansion ratio outside the boundary layer, $r_i = 2$, where all gradients are much lower than those near-wall. Even further away from the wall a uniform grid is used, or with a low cell expansion ratio, $r_o = 1.07$. In Section 3.1, this AMS mesh will be compared with an LRN mesh regarding accuracy and computational lead-time for a few set-ups at three different Reynolds numbers and variations of cell expansion ratios.

2.3. New numerical wall functions

Two different robust numerical wall functions have been developed to solve full Navier–Stokes equations, both in a main-grid and a sub-grid. The sub-grid equations are assembled and solved prior to the assembling of the main-grid equations. In contrast to earlier numerical wall functions (Craft et al., 2004; Myers and Walters, 2005; Bond and Blottner, 2011; Chedeveigne, 2010), local continuity is enforced and the main-grid and sub-grid use identical advective and diffusive fluxes at the interface, see Fig. 2. The first model, labelled NFF, Numerical Face Flux, employs face fluxes in both directions, i.e. to and from the sub-grid, for all solved entities. The only difference from the second model called NWF, Numerical Wall Flux, is that the velocity and temperature from the sub-grid to the main-grid use wall fluxes, which is also how Craft et al. (2004) coupled these entities in this direction.⁴ Before going into detail about the coupling strategies, the mesh set-up is depicted.

2.3.1. Mesh

The sub-grid is defined to overlap the wall-adjacent cells in the main-grid, see Fig. 2. However, the concept of having a sub-grid with multiple cell layers in the wall-adjacent cells of the main-grid will lead to mesh size discontinuities at the interface between the main-grid and sub-grid. The two cell layers closest to the wall of the main-grid are normally close in size. Still, the top cell layer in the sub-grid facing the second cell layer in this main-grid is far smaller in size, dependent on the number of cell layers in the sub-grid. Bäcker (2016) investigated the magnitude of this possible discretization problem using three different mesh strategies for the main-grid, as displayed in Fig. 3. It was found that the total number of cells in the extended *eMesh* can be kept almost as low as in the discontinuous *dMesh* at the same time as discretization errors are almost completely mitigated. The purpose of the continuous *cMesh*, see Fig. 3a, is to resemble a standard LRN mesh as close as possible. This mesh is only used during verification of the implementation of the robust numerical wall functions.

2.3.2. Numerical face flux model (NFF)

In both the main-grid and the sub-grid, boundary conditions for all entities are set up as in a standard wall treatment of the LRN model with one exception. At the *interface boundary*, i.e. at the boundary of the sub-grid which faces the bulk flow, see Fig. 2, all

² Observe that the evaluation of P_k , involving velocity gradients, at the interface between the main-grid and the sub-grid needs special treatment, see relevant paragraph in 2.3.2.

³ This implementation uses total viscosity in the expression of the turbulent production instead of the more common use of the molecular viscosity.

⁴ In the opposite direction, where the NWF model uses face fluxes, Craft et al. set a Dirichlet boundary condition for the sub-grid entities with values interpolated from the main-grid except for the wall-normal velocity. This velocity component is instead calculated from continuity within each of the sub-grid cells but then scaled to ensure that its value at the outer edge of the sub-grid matches the corresponding value in the main-grid.

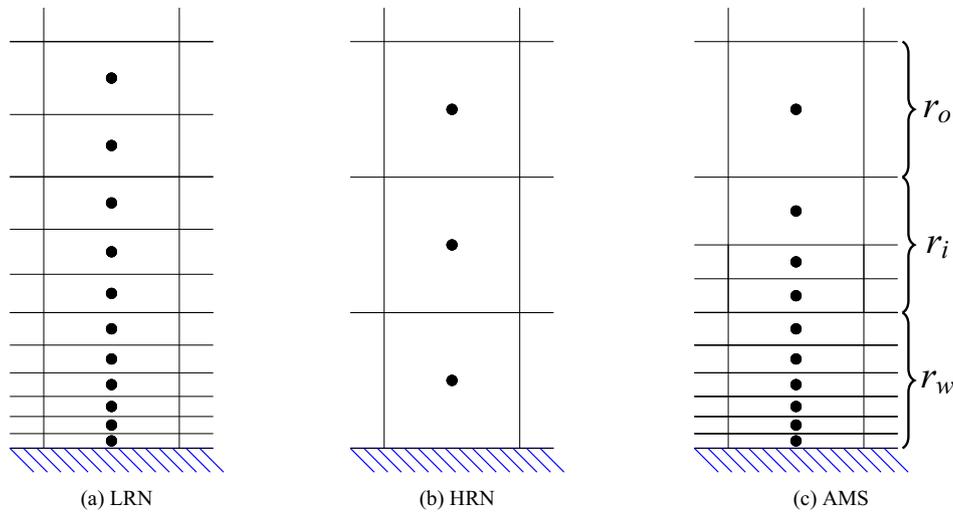


Fig. 1. Typical near-wall mesh strategies for different wall treatments, where HRN and LRN have a constant cell expansion ratio, r . In contrast, the AMS has three regions with different ratios.

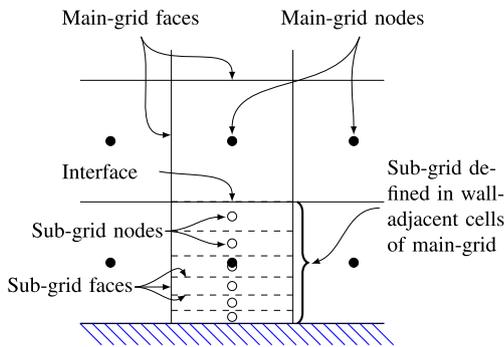


Fig. 2. Sketch of the sub-grid arrangement. The height of the sub-grid's cells increases with increasing distance to the wall.

entities solved for are set to a zero-flux condition.⁵ New interface fluxes are then added as source terms, see below.

Continuity. As the pressure is not solved on the sub-grid, the continuity needs to be achieved in another way. The mass flux, stored at the wall-normal faces of the sub-grid, is normalized with the corresponding cell faces of the main-grid, see Fig. 4. This is done so that the sum of the mass flux of sub-faces equals the mass flux to the overlapping main face. If the number of layers in the sub-grid is larger than one, there are also wall-parallel faces in the sub-grid not overlapping any face of the main-grid. For these sub-faces the mass flux is updated to ensure that the sum of the mass flux per cell is zero, which is the same as a divergence-free velocity field.

The pressure gradient in the sub-grid is calculated by mapping it from the main-grid. The normalizing of the mass flux, as well

⁵ The zero-flux condition is implemented as a derivative of a homogeneous Neumann condition, i.e. *zeroGradient* condition in OpenFOAM, where both the advective flux and the diffusive flux are set to zero. The motive is to cancel any flux arising from the boundary condition and let the wall model add the calculated flux from the main-grid as a source term, see Eqs. (12), (13), and (15).

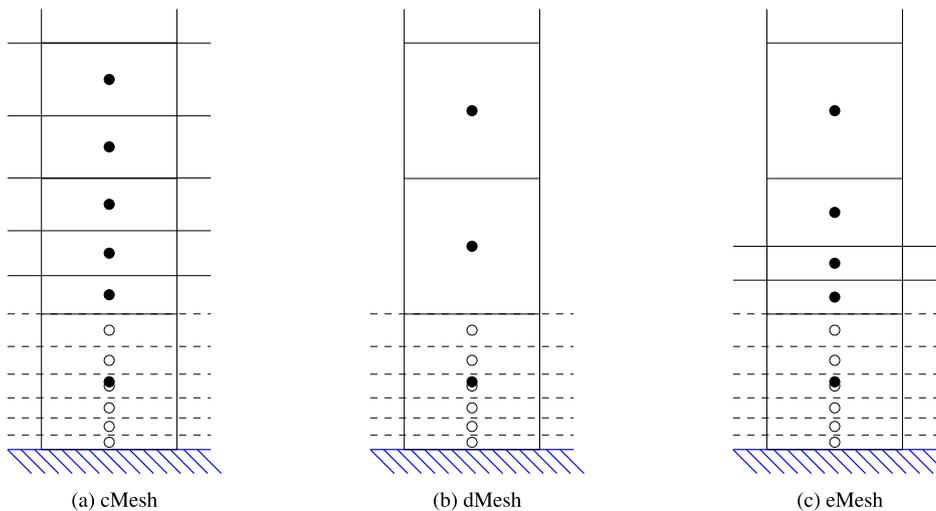


Fig. 3. Sketch of the topology of three different main meshes with difference in cell size discontinuities. Observe that the sub-grid cells, represented by non-filled circles at the node centres and dashed lines at the faces, overlap a larger main cell whose node centre is displayed with a filled circle.

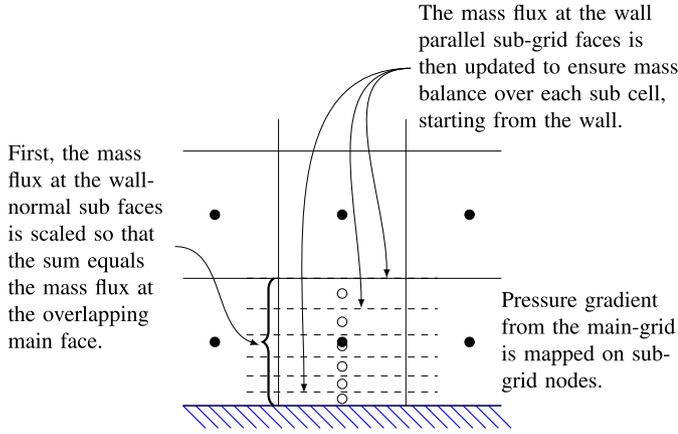


Fig. 4. A principal sketch of the mapping of mass flux and the mapping of the pressure gradient from the main-grid to the sub-grid.

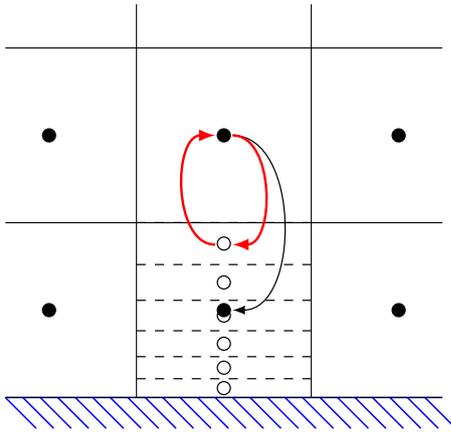


Fig. 5. Principal sketch of the coupling in the face flux model where all entities are coupled with face fluxes: to and from the sub-grid's top node, —, and within the main-grid, —. Observe the lack of coupling from the main wall cell, which has been decoupled.

as the mapping of the pressure gradient, is done before the momentum equations are assembled every iteration. In this way, continuity is secured in all cells in the sub-grid. This is in contrast to the UMIST-N model by Gant (2002) where the mass flux across the wall-normal faces in the sub-grid is not normalized with the corresponding mass flux in the main-grid. However, this model does calculate the wall-normal velocity component based on the wall-parallel velocity component using continuity which temporarily generates a divergence free velocity in the sub-grid. Later, when the wall-normal velocity component is scaled with the corresponding component in the main-grid to achieve matching velocities at the interface between the main-grid and the sub-grid, the local continuity in the sub-grid is destroyed. However, Gant shows that the resulting wall-normal velocity component is unaffected at the stagnation zone of an impinging jet even though local continuity is not rigorously satisfied in the sub-grid.

Discretization at the interface. The coupling in the main-grid between the wall cell and its neighbour away from the wall is decoupled for all entities but pressure and replaced by couplings to and from the sub-grid, see Fig. 5. The couplings between the main-grid and sub-grid of all entities solved for on the sub-grid are handled with face fluxes, almost in the same way as grid internal cells are connected within both grids. These fluxes include advection and diffusion. For the diffusion the explicit, S_p , and the implicit, S_u , sources can be written as, super-scripts refer to main or sub, re-

spectively, and x refers to both:

$$S_p^x = [(1 - w)v_s + wv_m] \frac{1}{\delta y_s + \delta y_m} |S_f| \quad (12)$$

$$S_u^s = S_p^x \phi^m \quad (13)$$

$$S_u^m = S_p^x \phi^s \quad (14)$$

Here, v is the total viscosity, δy_s and δy_m are the distances from the interfacing face, with a face normal area of S_f , to the closest node in the sub-grid and the main-grid, respectively, see Fig. 6. w is a weighting factor calculated as $w = \delta y_s / (\delta y_s + \delta y_m)$, and ϕ is the value at the neighbouring cell centre, across the grid interface, of the entity in question. In this implementation, the cross-term of the diffusion has been neglected, see last term in Eq. (2), as it is generally small compared to the main diffusion term.

For the advection, using the central-differencing scheme, the sources become:

$$S_p^s = (1 - w)(\bar{u}_f \cdot S_f) \quad S_u^s = -w(\bar{u}_f \cdot S_f) \phi^m \quad (15)$$

$$S_p^m = -w(\bar{u}_f \cdot S_f) \quad S_u^m = (1 - w)(\bar{u}_f \cdot S_f) \phi^s \quad (16)$$

where \bar{u}_f is the face velocity, interpolated from the main nodes on each side of the face,⁶ and ϕ means, as before, the cell-centre value of the entity in question.

Every iteration is started by solving all the transport equations in the sub-grid followed by the same in the main-grid. All source terms are calculated using the current values from the current grid, together with the earlier values calculated from the other grid.

Velocity gradients at the interface. The gradient in a cell is calculated from the face values using the Green-Gauss theorem. Thus, to calculate the gradient correctly, it is sufficient to interpolate the cell values to face values. For accurate interpolation to the interface between the main-grid and sub-grid the values of the interface adjacent cells in both the sub-grid and the main-grid must be used. One example containing gradients is the cross-terms of the diffusion, which in this implementation have been neglected, see the previous paragraph. However, the turbulent production includes the gradient of the velocity, see Eqs. (6) and (11), and must also be calculated at the interface. To evaluate this gradient correctly on the sub-grid we temporarily set a Dirichlet condition at the interface by interpolating the interface-adjacent cell of the sub-grid and the cell of the main-grid on the other side of the interface. Directly after the turbulent production field has been calculated in the sub-grid, the Dirichlet condition for the velocity is set back to a zero-flux condition.

For the velocity in the main-grid we instead use the wall-adjacent cell which is temporarily set to a value so that a cell-to-face interpolation gives the same value as is calculated for the interface boundary of the sub-grid. The wall-adjacent velocity is reset to its original value as soon as the turbulent production field has been calculated.

Turbulent viscosity in wall cells. When face flux is used as coupling between the main-grid and sub-grid for the turbulent entities there is no need to solve any turbulent transport equations in the wall-adjacent cells of the main-grid. However, in the momentum equation, the turbulent viscosity must be calculated also

⁶ Tests have also been made where \bar{u}_f was interpolated from a combination of main and sub nodes but no noticeable difference could be noted. This is not a surprise as the mass flux in the sub-grid has been normalized by the mass flux from the main-grid, see previous paragraph about continuity.

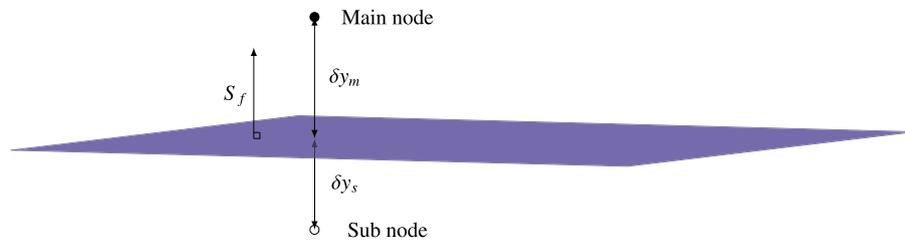


Fig. 6. A sketch of the relation between a face and the related nodes on each side at the interface between the main-grid and the sub-grid.

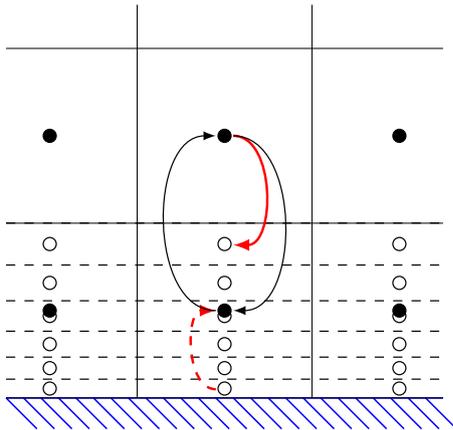


Fig. 7. Principal sketch of the velocity coupling from the main-grid and sub-grid for the wall flux model. The wall shear stress calculated in the sub-grid's wall cell, --- , is added to the main-grid's momentum equation as a wall flux. From the main-grid to the sub-grid the velocity is coupled with face flux, --- , as in the face flux model. In the main-grid, the face fluxes, --- , are untouched. Instead, the Dirichlet wall boundary condition in the main-grid is replaced by a slip condition.

for these cells as an input to the diffusion term. This is done by cell averaging each turbulent entity from the overlapping sub-grid cells. From these averaged values, the turbulent viscosity is calculated according to the definition of the specific turbulent model used. At the end of the chain, the OpenFOAM machinery interpolates the cell values onto the relevant face to get the face values of the turbulent viscosity which are needed in the equation.

2.3.3. Numerical wall flux model (NWF)

In the wall flux model, the coupling in the main-grid from the wall cell to its neighbour further away from the wall is not decoupled. Instead, a slip boundary condition for the velocity is set at the wall of the main-grid and the coupling from the sub-grid is in the form of wall shear stress, similar to what is used in the UMIST-N model of Craft et al. (2004), see Fig. 7. The source term added to the momentum equation in the wall-adjacent cells of the main-grid is:

$$S_u = -\nu \frac{1}{\delta y} |S_f| \bar{u}_s, \quad (17)$$

where ν is the physical viscosity, δy is the distance from the wall to the wall-adjacent node in the sub-grid, $|S_f|$ is the cell face area at the wall and \bar{u}_s is the velocity in the wall-adjacent node of the sub-grid. A similar technique outlined here for the velocity is also used for the temperature.

2.4. Approximations in the numerical wall function

The only approximation in the present numerical wall functions, i.e. the NFF and the NWF, is that the pressure gradient in sub-grid cells is constant in the wall normal direction and equal to the corresponding term in the overlapping main-grid cell. Further,

these wall functions solve the equations in the main-grid and in the sub-grid in sequence in an iterative manner. This is in contrast to the AMS where the equations are solved in one monolithic matrix per entity. However, previous numerical wall functions (Craft et al., 2004; Myers and Walters, 2005; Bond and Blotner, 2011; Chedeveigne, 2010), only solve boundary layer equations on the sub-grid and used a few supplementary but different approximations. Specifically, in their UMIST-N model, Craft et al. (2004) solve an equation for the wall-parallel velocity and calculate the wall normal velocity from continuity in the sub-grid. Moreover, the wall-parallel diffusion is neglected and the sub-grid is actually divided in many one-dimensional sub-grids, each one of them is overlapping a wall-adjacent cell in the main-grid. We believe, the intention has been to create a parabolic form of the equation where the downstream flow does not influence the upstream flow. These equations have then been solved using a fast marching scheme in the streamwise direction with TDMA (Tri-Diagonal Matrix Algorithm). We evaluate this assumption in a variant of the NWF model, labelled UMI, but the equations on the sub-grid are still solved monolithically as the sub-grid has not been divided into smaller one-dimensional grids. Myers and Walters (2005) and Bond and Blotner (2011), whose model is named the “diffusion model”, here labelled DIF, simplified the equations in the sub-grid even further by including only the wall-normal diffusion. In this way, they were able to simplify the equations to be completely one-dimensional. In another variant of the NWF model, called NnA (Numerical no Advection), we evaluate this assumption and hence neglect advection for all solved entities. As in UMI, all entities are solved on the full sub-grid in three dimensions. It should be noted that we evaluate what effect these different assumptions will give on their own without using a fast marching scheme, which of course could have been deployed for these parabolic equations, see e.g. Patankar and Spalding (1972). Iacovides and Launder (1984) reports a speed-up of 2–3 of the computational time for PLS, i.e. their implementation of this kind of time-marching scheme, applied on a low-Reynolds-number setup.⁷

The under-relaxation used in each model has been adapted individually to give shortest lead-time for converging the solution. For an overview of solver and modelling differences in the different numeric wall treatments see Tables 1 and 2, respectively.

2.5. Set-up of test cases and used output

Here, we outline the set-up of the test cases used in the verification and validation of the different wall treatments. For all calculations performed using an LRN model, the mesh density is chosen to give an y^+ value close to unity, in practice $y^+ \in [0.75, 1.02]$. For all solved entities, the central differencing scheme, CD, is used for laminar set-ups, and the linear upwind scheme, LUD, is used for turbulent set-ups.

⁷ Craft et al. (2004) report that the PLS approach later was abandoned by its originators after encountering difficulties when going to more complex geometries.

Table 1

Overview of solver differences between the different wall treatments. Acronyms used in table; BCG (pre-conditioned Bi-Conjugated Gradient), TDMA (Tri-Diagonal Matrix Algorithm).

Property /	Model Grid	NFF	NWF	UMI	NnA	UMIST
Co-located	Main	Yes	Yes	Yes	Yes	No
	Sub	Yes	Yes	Yes	Yes	Yes
Algebraic solver	Main	BCG	BCG	BCG	BCG	TDMA
	Sub	BCG	BCG	BCG	BCG	TDMA
Relax.	U,V,W	Main	0.9	0.9	0.9	0.35
		Sub	0.9	0.9	0.9	1.0
T	Main	0.99	0.99	0.99	0.99	0.6
	Sub	1.0	1.0	1.0	1.0	1.0
k, ε	Main	0.8	0.8	0.8	0.8	0.45
	Sub	0.8	0.8	0.8	0.8	0.85

Table 2

Overview of model differences between the different wall treatments. Acronyms used in table; Cnt (Entity evaluated from continuity), Slv (Entity is solved for), FF (Face Flux), WSS (Wall Shear Stress), SST (Source and Sink Terms of the turbulent equations), DIV (Dirichlet using Interpolated Values from main-grid).

Treatment of		NFF	NWF	UMI	NnA	UMIST	DIF
Solved dimensions		3	3	3	3	3*	1
Accounts for advection		Yes	Yes	Yes	No	Yes	No
Mass flux		Cnt	Cnt	Cnt	N/A	Cnt	N/A
Wall-normal velocity		Slv	Slv	Cnt	Slv	Cnt	Cnt
To main	U, T	FF	WSS	WSS	WSS	WSS	WSS
	k, ε	FF	FF	FF	FF	SST	SST
To sub	U, T	FF	FF	FF	FF	DIV	DIV
	k, ε	FF	FF	FF	FF	DIV	DIV

*The UMIST model solve its equations in 3D but on a number of one-dimensional grids.

2.5.1. Channel flow

The channel flow is a simple test case with neither a wall-normal advection nor a non-zero pressure gradient. Fully developed flow is ascertained by using a periodic boundary condition in the streamwise direction, together with a pressure gradient as the driving source.

2.5.2. Axisymmetric impinging jet

The test case chosen for most verifications and validations of the numerical wall models in the literature is the axisymmetric semi-confined impinging jet on a plane surface (Baughn et al., 1991; Cooper et al., 1993), see Fig. 8. The distance between the nozzle and the impinging wall is set to two nozzle diameters, the outer radius of the impinging wall is set to 10 nozzle diameters, and the half angle of the sector is set to 1°. The boundary opposite of the impinging wall is set to a symmetric boundary condition.⁸ Fully developed flow in the nozzle is ascertained by feeding the inlet from a pipe with periodic streamwise boundary conditions. The impinging jet involves both advection and diffusion, and has a non-uniform pressure gradient. Hence, we can here verify whether these features are implemented correctly.

For comparisons the skin friction, C_f , and the Nusselt number, Nu , are used. The skin friction is defined as $C_f = 2\tau_w / (\nu U_{in}^2)$, where τ_w is the wall shear stress, ν is the viscosity, and U_{in} is the velocity at the inlet. The Nusselt number is defined as $Nu = \nabla_{Nw} T / (T_w - T_{in}) / D$, where $\nabla_{Nw} T$ is the wall-normal gradient of the temperature, T_w and T_{in} are the temperatures at the wall and the inlet, respectively, and D is the inlet diameter.

⁸ Use of a symmetry condition is a common simplification of the more correct use of an ambient pressure condition. The symmetry condition adds stability to the solution but may have an impact on comparisons with experiments. As no comparison with experiments is made, we allow ourselves to use this simplification.

2.6. Verification and validation

In the concept of systems engineering the definition of verification means that the problem has been solved “right”, and validation should ensure that the “right” problem is being worked on. Schwer (2009) presents a commonly accepted definition of verification and validation of mathematical models, where the second part focuses on asserting the predictive capability of the model. However, this research aims to reduce the computational cost, or improve the speed-up, of the predictive capability of a turbulence model family. The aspect of computational cost of the predictive model is silently ignored by Schwer (2009) but may be mapped to the more general definition of validation in systems engineering. Hence, the definitions used here are a mixture of both definitions above. Verification is used to affirm that certain simulations using the robust numerical wall functions give more or less identical results as a simulation using the default wall treatment, i.e. a LRN model. This is close to the special definition of code verification by Schwer (2009). Several aspects are tested during validation, accuracy, robustness and speed-up, which maps closest to the counterpart in systems engineering.

2.6.1. Verification

During the verification, only grids of cMesh type, see Fig. 3a, are used to mimic LRN as closely as possible. For the same reason, the CD advection scheme would also have been used for turbulent set-ups if it were not a question of robustness problems. Instead, the LUD scheme is used for all turbulent set-ups. Recall that the advective fluxes over the interface are computed using the CD scheme. Except for differences in the advection scheme, a correct implementation would give identical results for the sub-grid model and a default wall treatment of any low-Reynolds-number turbulence model under any of the following premises:

- the pressure gradient is constant across the domain covered by the sub-grid. The test case is the turbulent channel flow.
- the sub-grid has only one cell layer. The test case is the laminar set-up of the impinging jet.

The exchange of fluxes to and from the sub-grid for all entities needs to be verified. First, the diffusion is tested using a turbulent channel flow, $Re_\tau = 590$, using the Launder–Sharma model. However, as a turbulence model is included, the correction of the velocity gradients used in the production term at the interface is also verified. Recall that the turbulent production is proportional to the square of the velocity gradient. To test for a combination of advection and diffusion, the laminar axisymmetric impinging jet on a plane surface is used, $Re_D = 450$. This test case also includes a non-uniform pressure gradient. Hence, all of these features are tested simultaneously but must be tested with a single layer in the sub-grid. The verification is presented in Section 3.3.

2.6.2. Validation

The purpose of the validation is to investigate the usefulness of the different wall treatments in terms of accuracy and speed-up. In addition, the difference in robustness, i.e. the easiness of finding a converged solution, is highlighted. The HRN approach represents the fast, low precision prediction, and the LRN approach represents the slow, high precision prediction. In the LRN approach, many more cells are used as compared to what is used in HRN. It thus has a higher computational cost. The AMS mesh together with all the numerical wall functions represent different strategies of finding a sweet-spot between accuracy and speed-up.

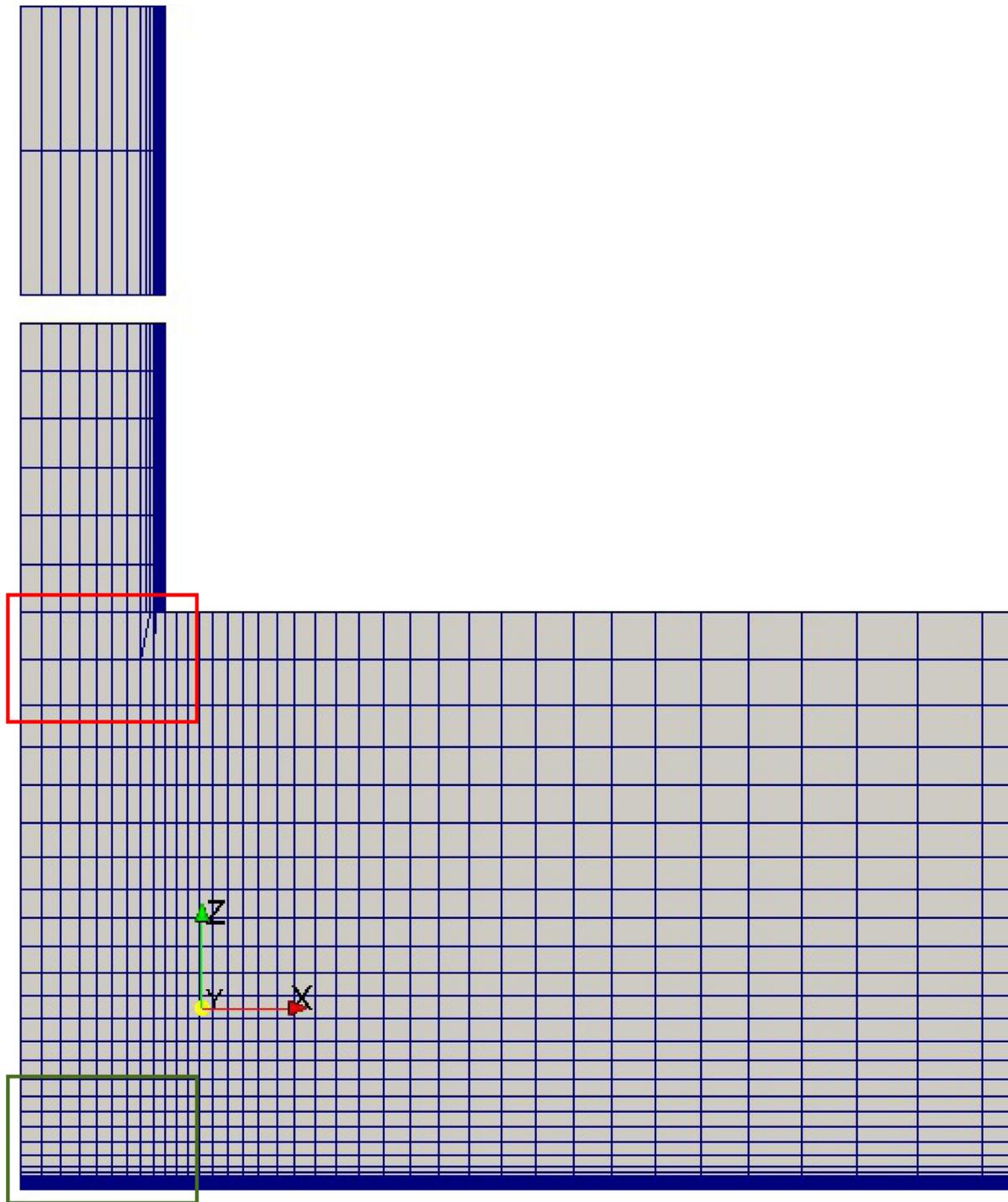


Fig. 8. Left part of an AMS mesh for a turbulent axisymmetric impinging jet set-up at $Re_D = 23,000$ where the symmetry line is at $x = 0$ and the impinging wall at $z = 0$. The inlet of the impinging jet is fed with flow conditions from a periodic pipe, i.e. fully turbulent flow. The coloured frames are locations of zoom in, see Figs.9 and 10.

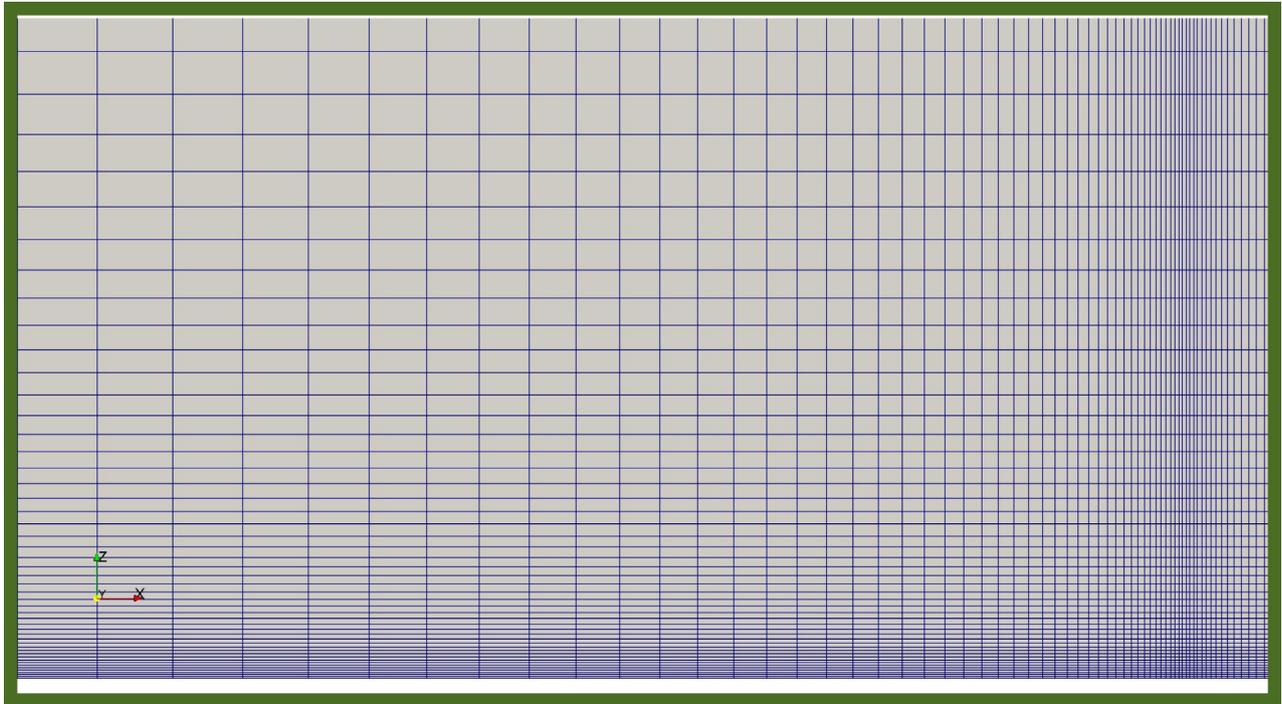
3. Results

3.1. Evaluation of meshing strategies

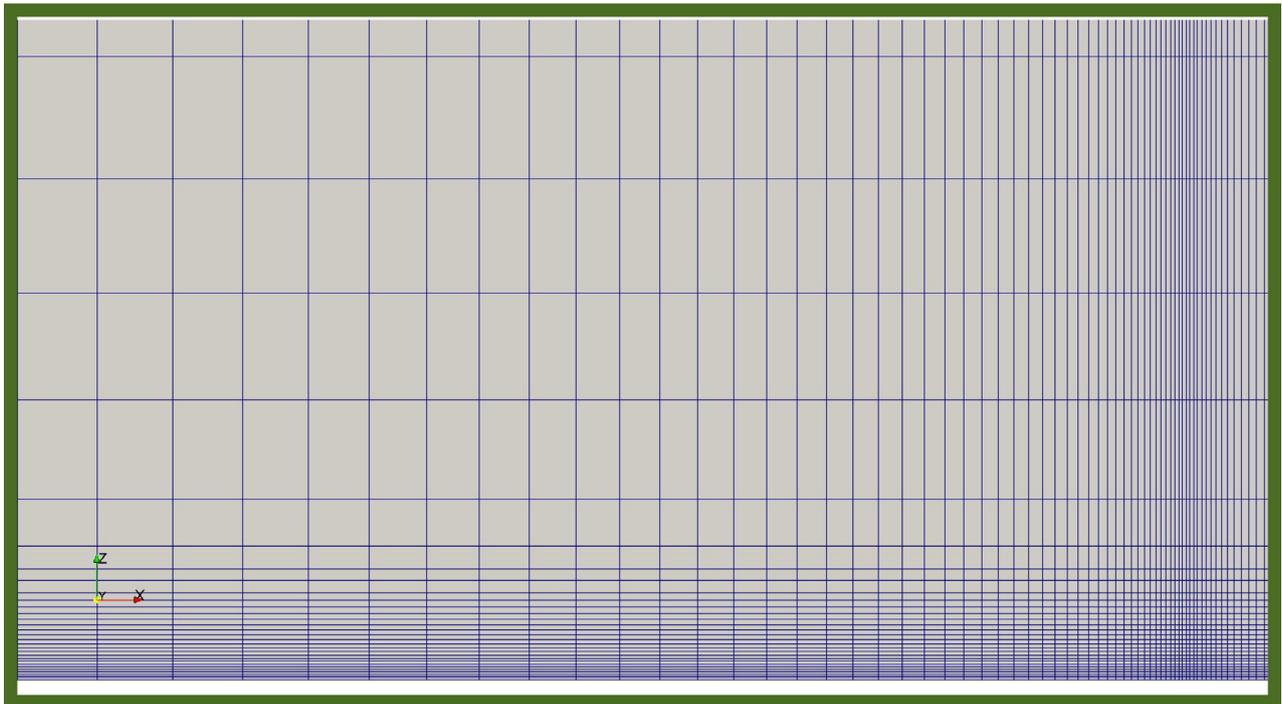
Fig. 8 displays the part of the mesh containing the inlet for a set-up of the turbulent impinging jet where AMS is applied to both the impinging wall and the pipe wall. The thick blue line at the walls is in fact many thin lines representing the borders of many thin cells with a high aspect ratio. The coloured boxes, in green

and red, indicate the regions where zoom-ins are available, see Figs.9 and 10.

In Table 3, the number of cells, required iterations, and lead time are collected for a number of set-ups of the turbulent impinging jet, at three different Reynolds numbers, using different mesh strategies at the impinging wall and the pipe wall. Applying the AMS reduces the number of cells compared to a standard LRN set-up to a fraction of around 0.6 for AMS at the impinging wall, 0.4 for AMS at the pipe wall, and 0.25 when AMS is applied to both walls. The number of iterations are also reduced, but only for set-



(a) for a typical LRN mesh.



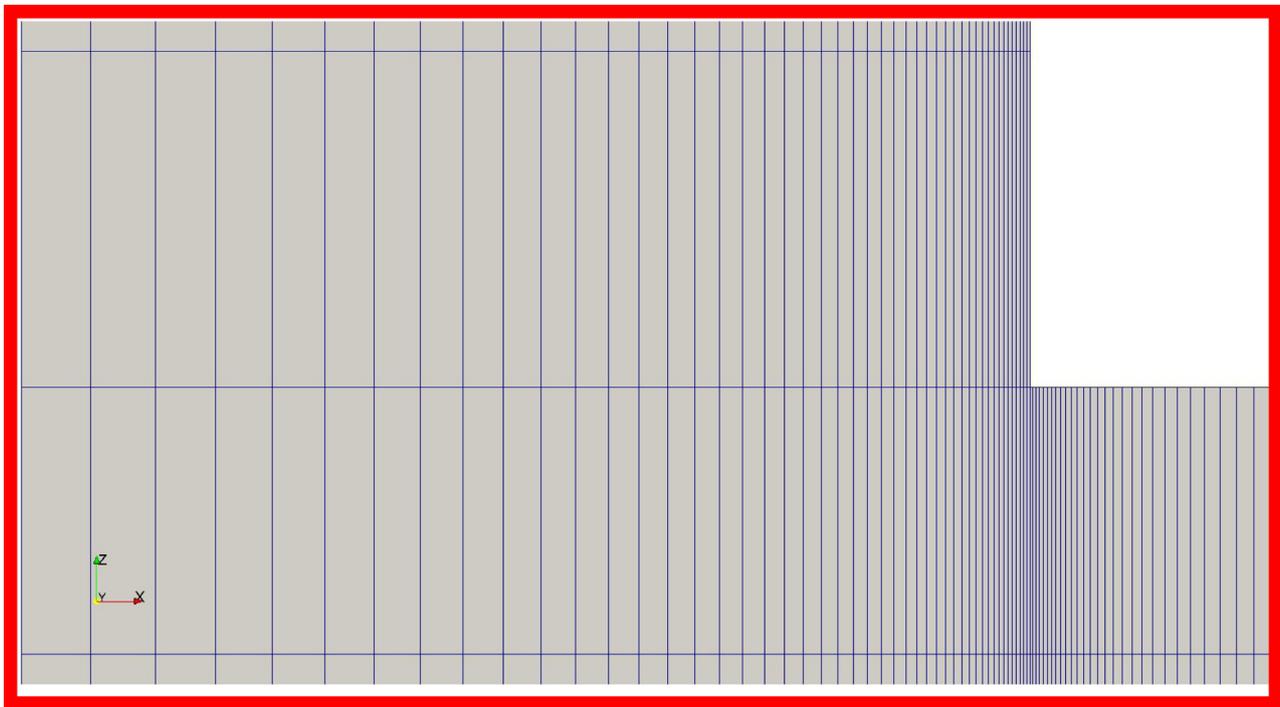
(b) for an AMS mesh for the impinging wall (z-direction).

Fig. 9. Meshing details at the impinging wall for a mesh for turbulent axisymmetric impinging jet at $Re_D = 23,000$. For location see coloured frames in Fig. 8.

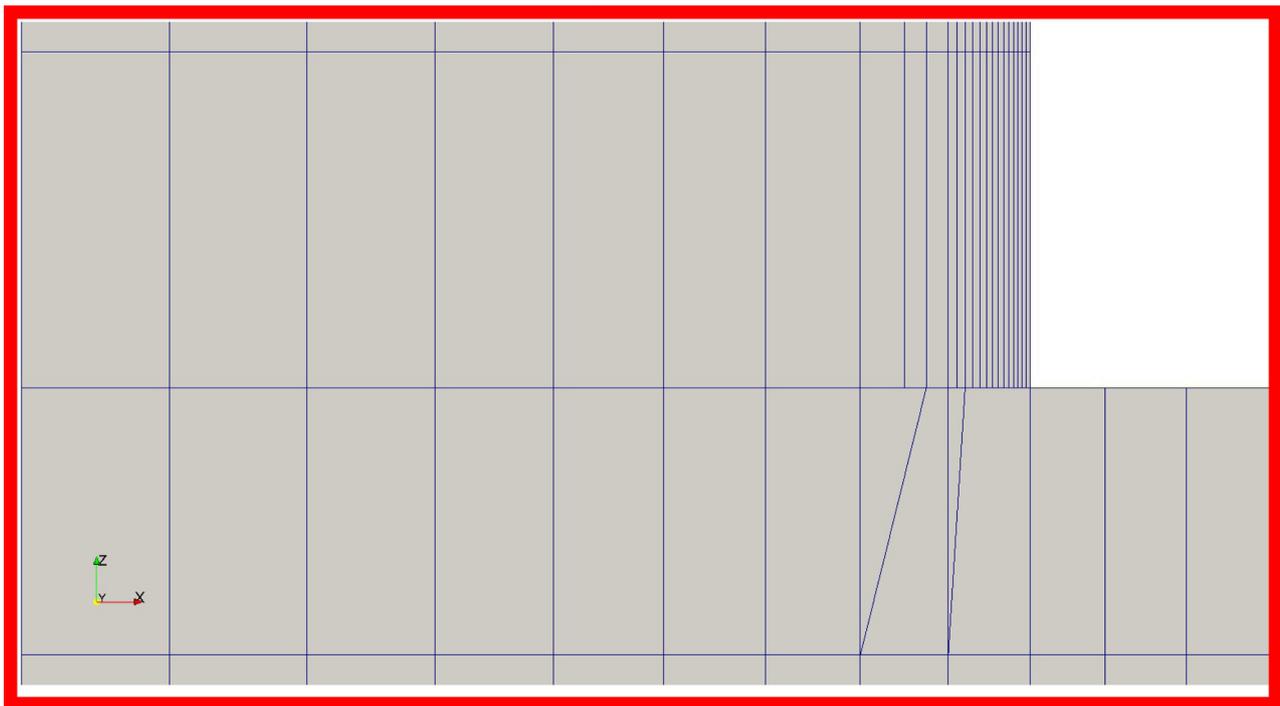
ups where AMS is applied to the pipe wall. The combined effect of a fewer number of cells and iterations leads to a substantial reduction in lead time, a factor between 6 and 12 when AMS is applied to the pipe wall and between 9 and 16 when applied to both walls.

The turbulent dissipation is often the entity that displays the highest gradients in a RANS simulation. Thus, this is a good candidate to use when asserting that the mesh is fine enough to re-

solve the shear and boundary layers. In Fig. 11, the magnitude of the gradient of the turbulent dissipation is compared for a set-up using AMS with a typical LRN set-up at a Reynolds number of 71,000. Only a minor difference can be spotted around the jet's shear layer close to the pipe exit, where the typical LRN mesh, using four times as many cells, resolves it only slightly better.



(a) for a typical LRN mesh.



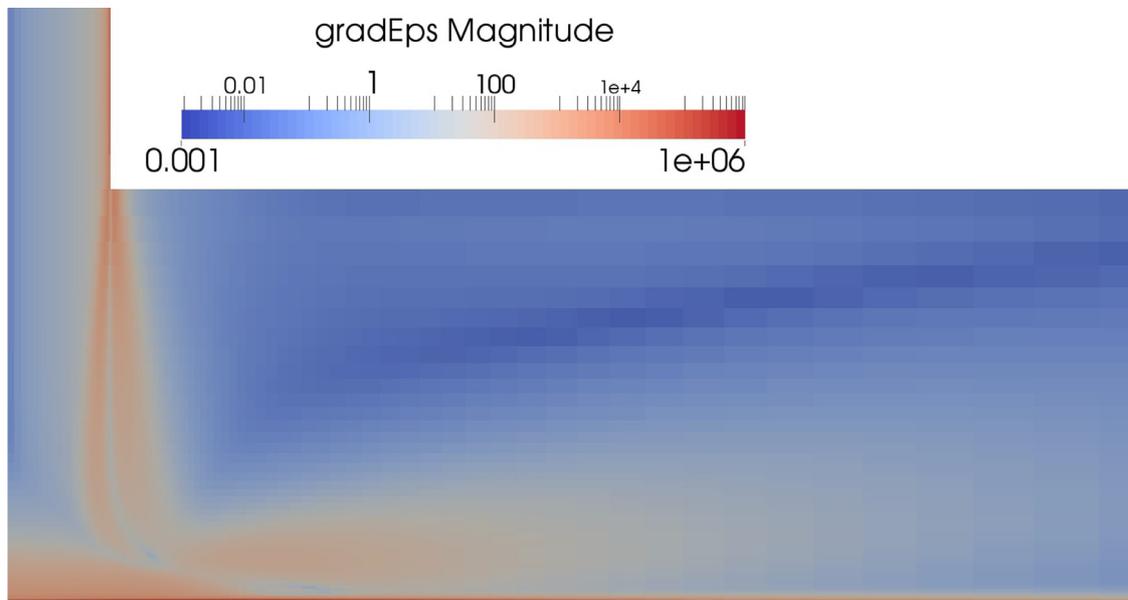
(b) for an AMS mesh for the pipe wall (x-direction).

Fig. 10. Meshing details at the pipe for a mesh for turbulent axisymmetric impinging jet at $Re_D = 23,000$. For location see coloured frames in Fig. 8.

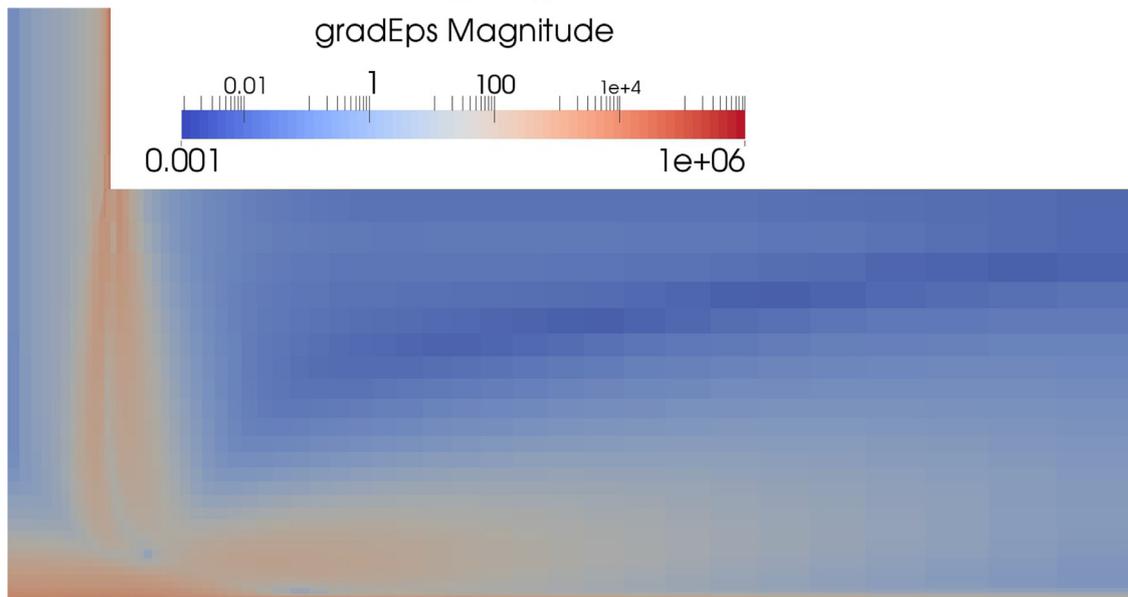
To examine the features at the wall, we examine the wall shear stress and the heat transfer at the wall, which are both plotted in Fig. 12 for all mesh strategies at a Reynolds number of 71,000. The wall shear stress in Fig. 12a is evaluated to exactly the same values for all mesh strategies, as all lines are plotted on top of each other. The only possibility to separate the results from each other is by recognising that the number of cells along the impinging wall

differ if AMS is applied to the pipe wall or not. For the heat transfer, a similar agreement is shown in Fig. 12b except for the cell at the very centre of the stagnation point, where a small difference is noticeable.

Hence, from a naive meshing strategy, the number of cells can be reduced by a factor of four and the lead time a factor of ten without sacrificing any accuracy using the AMS near the walls.



(a) for a typical LRN mesh.



(b) for an AMS mesh.

Fig. 11. The magnitude of the gradient of the turbulent dissipation, $\bar{\epsilon}$, at $Re_D = 71,000$.

3.2. Evaluation of mesh stretching

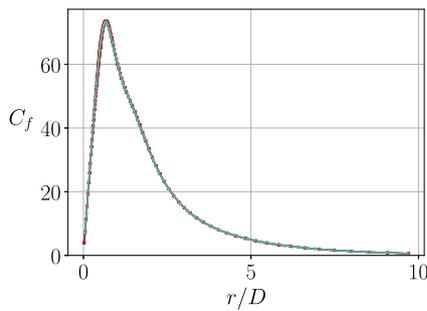
Similar data are collected in Table 4, as were earlier displayed in Table 3, from set-ups with different stretchings, i.e. geometric cell expansion. The same stretching is applied in the wall normal direction for both the impinging wall and the pipe wall, including the cells below the nozzle exit. Both the standard LRN, with a uniform stretching, and the AMS meshes have been considered, where the given stretching refers to the near-wall stretching, r_w . For the AMS mesh the stretching in the ‘interface’ region, r_i , and in the outer region, r_o , also have to be set, see Figs. 1 and 8. Here the outer stretching, r_o , is set to give the same maximal cell sizes in the wall normal direction as the corresponding LRN set-up. The ratio used in the region interfacing the near-wall and the bulk region, r_i , is set to a value of two to quickly increase the cell size going from the wall to the bulk. In order not to introduce large discretization er-

rors using a high stretching, it is important to locate the interface region to where there are not any high gradients. Specifically, this region should be at a sufficient distance from the wall where the gradient of the turbulent dissipation has decreased considerably from its near-wall peak value. Three different stretchings, near-wall for AMS, are considered, $r_w \in [1.07, 1.09, 1.11]$, which were all demonstrated to be robust in terms of reaching a converged solution. As is shown in Table 4, an increased stretching ratio results in fewer cells and iterations and hence also shorter computational lead-times. Going from $r = 1.07$ to $r = 1.11$ gives less than half the lead-time for both LRN and AMS. The reference used for the relative lead-time is the corresponding AMS set-up from Table 3 with stretching ratios $r_w = r_o = 1.07$ and $r_i = 2$. This difference in outer stretching, r_o , between the reference and the corresponding AMS set-up in Table 4 gives a difference in relative lead-time of 30%.

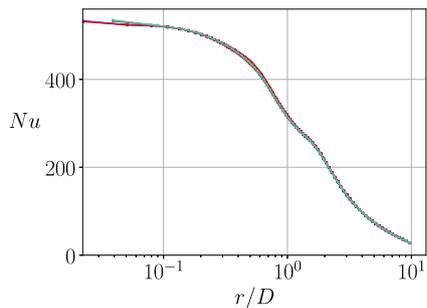
Table 3

Number of cells and computing times, i.e. wall clock time, for the axisymmetric impinging jet using the Launder–Sharma model for different mesh strategies/wall treatments at the impinging wall. y_i^+ denotes location of the interface.

Re_D	Meshing		Wall treat.	#Cells	Iter.	Time	
	$y_{i,pipe}^+$	$y_{i,imp.}^+$				[s]	(rel.)
23000	–	–	LRN	10,925	1384	217	10.3
	–	100	AMS	6815	1305	126	6.0
	50	–	AMS	4196	743	42	2.0
71000	50	100	AMS	2636	521	21	1.0
	–	–	LRN	15,503	2548	590	8.6
	–	200	AMS	10,058	2755	407	5.9
220000	100	–	AMS	5657	1540	130	1.9
	100	200	AMS	3710	1164	69	1.0
	–	–	LRN	21,278	6738	2010	15.8
	–	300	AMS	14,648	7472	1647	13.0
	150	–	AMS	8158	2028	279	2.2
	150	300	AMS	5749	1645	127	1.0



(a) Skin friction, C_f , plotted along the impinging wall.



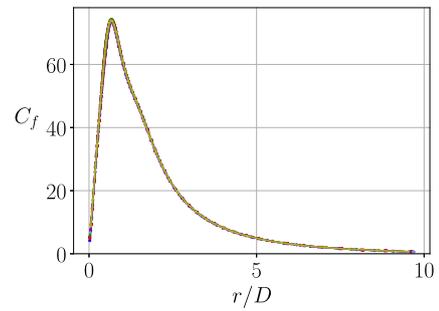
(b) Heat transfer, Nu , plotted along the impinging wall.

Fig. 12. Turbulent axisymmetric impinging jet at $Re_D = 71,000$ comparing; —: standard LRN, —: AMS ($y_i^+ = 100$) on pipe wall, —: AMS ($y_i^+ = 200$) on impinging wall, and —: AMS on both walls, y_i^+ as earlier.

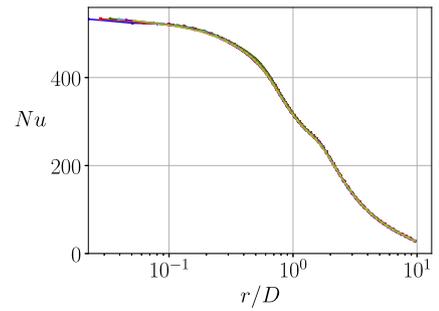
Table 4

Number of cells and computing times, i.e. wall clock time, for the axisymmetric impinging jet using the Launder–Sharma model for different mesh strategies. r_w denotes near-wall stretching.

Re_D	Meshing			Wall treat.	#Cells	Iter.	Time	
	$y_{i,pipe}^+$	$y_{i,imp.}^+$	r_w				[s]	(rel.)
71,000	–	–	1.07	LRN	15,503	2548	590	8.6
	–	–	1.09	LRN	10,608	2342	397	5.8
	–	–	1.11	LRN	7660	2180	277	4.0
	100	200	1.07	AMS	3946	1160	87	1.3
	100	200	1.09	AMS	2951	809	51	0.7
	100	200	1.11	AMS	2339	655	34	0.5



(a) Skin friction, C_f , plotted along the impinging wall.



(b) Heat transfer, Nu , plotted along the impinging wall.

Fig. 13. Turbulent impinging jet at $Re_D = 71,000$ with different mesh stretching, r_w , for standard LRN and AMS ($y_{i,pipe}^+ = 100$ and $y_{i,imp.}^+ = 200$); —: LRN ($r_w = 1.07$), —: LRN ($r_w = 1.09$), —: LRN ($r_w = 1.11$), —: AMS ($r_w = 1.07$), —: AMS ($r_w = 1.09$), and —: AMS ($r_w = 1.11$).

Fig. 13 plots the skin-friction and the heat transfer for all setups in Table 4. As can be seen, the results are plotted on top of each other for both skin-friction, see Fig. 13a, and heat transfer, see Fig. 13b. Thus, minor changes of the stretching ratio do not affect the accuracy but can substantially improve the lead-time.

3.3. Verification of wall function implementations

Fig. 14 compares the results for the turbulent channel flow of the default LRN, the face flux and the wall flux models on typical LRN meshes, see Figs. 1a and 3a. By observing the results in Fig. 14a we can already tell that the diffusion scheme has been implemented correctly, as the plots match perfectly. That the turbulent entities are also plotted on top of each other in Fig. 14b demonstrates that the velocity gradients at the interface have also been corrected properly before the turbulent production is evaluated in the cells next to the interface in the main-grid.⁹

Fig. 15 presents simulation results of the laminar impinging jet, where the verification of the CD advection scheme shows the same flawless agreement for both skin-friction and heat transfer as was earlier presented for the diffusion scheme.

3.4. Evaluation of new wall functions

It is important for a wall meshing strategy to be insensitive to the location of the interface as the boundary layer thickness is not known *a priori*. In Fig. 16, the wall shear stress and the heat transfer at the impinging wall are compared for different meshing strategies for two different locations of the interface near the impinging wall. At the peak of the wall shear stress, see Fig. 16a,

⁹ Recall that the turbulent production is proportional to the square of the velocity gradient.

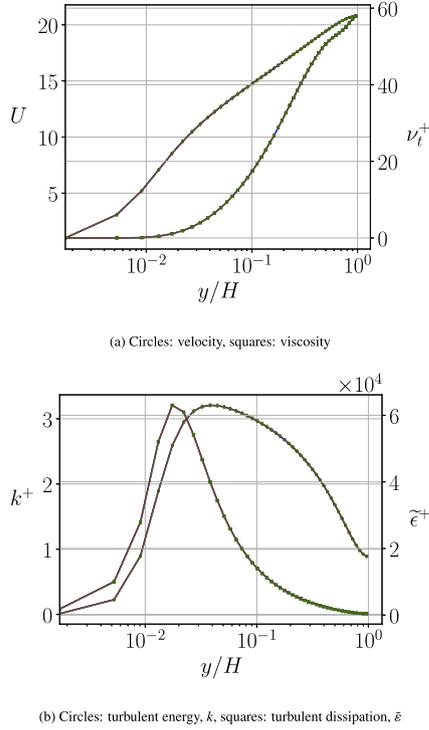


Fig. 14. Turbulent channel flow at $Re_\tau = 590$ comparing —: default wall treatment with the use of the sub-grid models, having 24 cell layers in the sub-grid; —: face flux and —: wall flux. Markers represent cell centres.

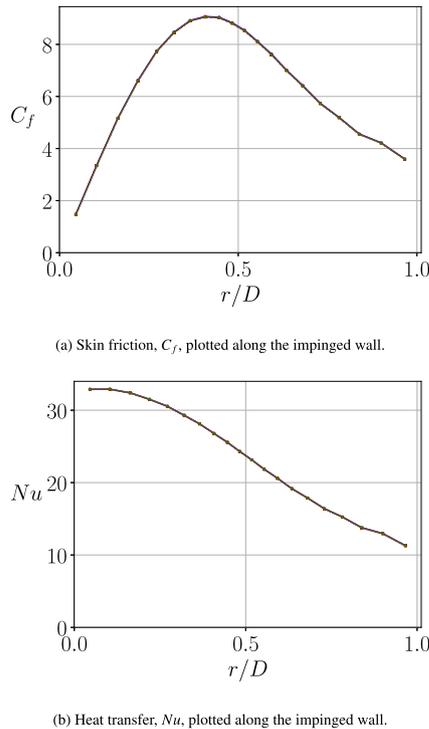


Fig. 15. Laminar semi-confined axisymmetric impinging jet at $Re_D = 450$ comparing —: default wall treatment with the use of the sub-grid models, having one single cell layer; —: face flux and —: wall flux. Markers represent face centres at the wall.

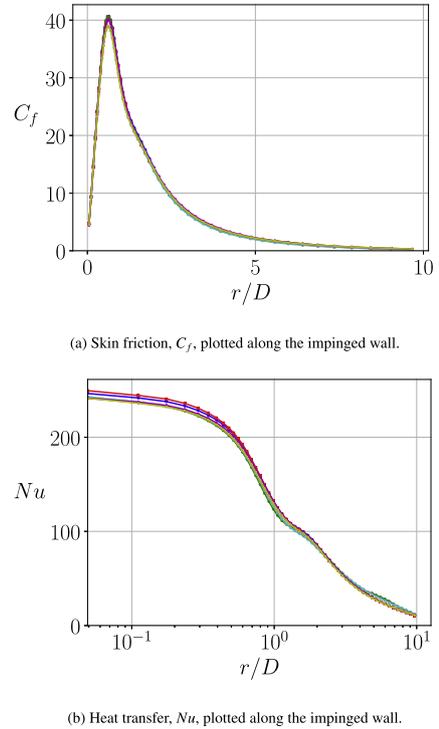


Fig. 16. $Re_D = 23,000$, AMS is applied to the pipe wall ($y_i^+ = 50$) and different near-wall strategies are applied to the impinging wall, comparing: —: AMS $y_i^+ = 100$, —: AMS $y_i^+ = 300$, —: NFF $y_i^+ = 100$, —: NFF $y_i^+ = 300$, —: NWF $y_i^+ = 100$, and —: NWF $y_i^+ = 300$.

a small difference is noticeable for the wall flux model, but overall, all mesh strategies are very insensitive to the location of the interface, $y_i^+ \in [100, 300]$. For the heat transfer, see Fig. 16b, the insensitivity is even more remarkable as there is hardly any at all. However, smaller differences between the different mesh strategies are visible in this plot. At the very centre of the stagnation point, both the face flux and the wall flux model predict the heat transfer slightly lower than with the implicit wall strategy. Another small difference is that the face flux model predicts one extra bump at $r = 5R$. It is believed that this has to do with the combination of using face flux coupling with the Yap correction, as this bump is not visible for set-ups without the Yap correction (not shown).

3.5. Evaluation of wall function approximations

The number of cells, required number of iterations, and lead time are collected in Table 5 for all investigated wall strategies. SLW denotes standard log wall functions applied to both the impinging and the pipe wall at a fixed wall distance, y^+ , indicated within parentheses in the columns for y_i^+ values.

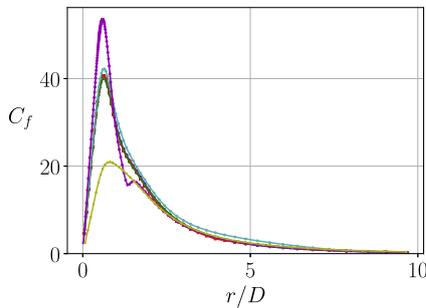
For all other set-ups, AMS has been utilized at the pipe wall and the indicated wall treatment at the impinging wall. The number of cells is governed by the y_i^+ parameter for the respective wall, and the aim has been to use one value for each Reynolds number. Unfortunately, some wall treatments have not been robust enough and, for these cases, the $y_{i,pipe}^+$ parameter has been reduced enough to reach a convergent solution. In this way, a wall treatment that is difficult to converge will result in a larger lead-time value, which is the case for the NnA treatment and occasionally for the NWF and the UMI treatments. The UMI treatment has been removed from the test at the highest Reynolds number as no convergent solution could be found.

The observation that can be made is that the SLW treatment is consistently five times faster than the reference AMS treatment.

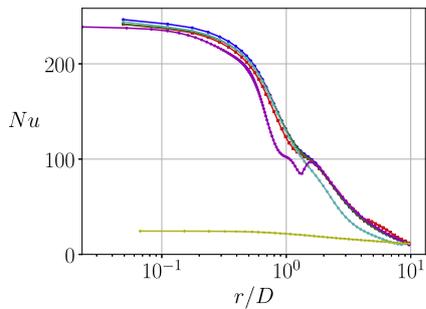
Table 5

Number of cells and computing times, i.e. wall clock time, for the axisymmetric impinging jet using the Launder–Sharma model for different mesh strategies/wall treatments at the impinging wall, UMI and NnA are defined in Section 2.4.

Re_D	Mesh interface		Wall treat.	#Cells (main+sub)	Iter.	Time	
	$y_{i,pipe}^+$	$y_{i,imp.}^+$				[s]	(rel.)
23,000	50	100	AMS	2636	521	23	1.0
	50	100	NFF	1544+1144	645	29	1.3
	50	100	NWF	1544+1144	548	24	1.0
	50	100	UMI	1544+1144	628	25	1.1
	2	100	NnA	3938+3014	1372	103	4.5
	(40)	(100)	SLW	678	318	4	0.2
71,000	100	200	AMS	3710	1164	78	1.0
	100	200	NFF	1999+1770	1295	79	1.0
	100	200	NWF	1999+1770	1133	73	0.9
	3	200	UMI	4877+4590	2141	239	3.1
	4	200	NnA	4613+4350	1754	192	2.5
	(40)	(100)	SLW	1742	389	12	0.2
220,000	150	300	AMS	5676	1886	173	1.0
	150	300	NFF	3121+2628	1951	156	0.9
	3	300	NWF	7331+6588	5149	818	4.7
	3	300	NnA	7331+6588	5144	781	4.5
	(40)	(100)	SLW	3663	728	43	0.2



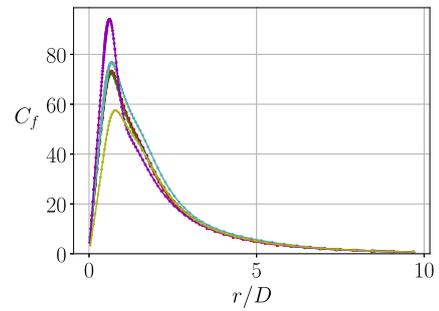
(a) Skin friction, C_f , plotted along the impinging wall.



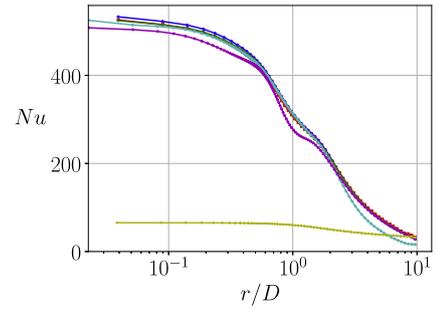
(b) Heat transfer, Nu , plotted along the impinging wall.

Fig. 17. $Re_D = 23,000$, AMS is applied to the pipe wall for all set-ups using an LRN model and different near-wall strategies are applied to the impinging wall, comparing: AMS, NFF, NWF, UMI, NnA, and HRN.

For set-ups that has the same value of the $y_{i,pipe}^+$ parameter as the AMS reference have also a lead-time within 10% of the reference. The exception is the NFF at the lowest Reynolds number which is 30% slower than the AMS. Work has been done to remove inefficiencies in the implementation of the different wall functions. And, as similar differences in lead-time between the numerical wall functions and the AMS are also seen in the number of iterations, one can rule out any major bottlenecks. Hence, taking into account computational lead-time no clear advantages can be seen for any of the numerical wall functions compared to “only” optimising the local mesh resolution (AMS).



(a) Skin friction, C_f , plotted along the impinging wall.



(b) Heat transfer, Nu , plotted along the impinging wall.

Fig. 18. $Re_D = 71,000$, AMS is applied to the pipe wall for all set-ups using an LRN model and different near-wall strategies are applied to the impinging wall, comparing: AMS, NFF, NWF, UMI, NnA, and HRN.

In Figs. 17–19, all wall strategies are compared in terms of shear stress and heat transfer at the impinging wall for different Reynolds numbers. The intention is to investigate the results from wall treatments using an LRN turbulence model and, hence, AMS is used as a reference. However, as the SLW treatment, employing an HRN turbulence model with a standard wall function, is used to compare computational lead-time, it is also included in these figures for completeness. For both the shear stress and heat transfer, all treatments display less difference from the reference with higher Reynolds number.

As the largest deviations, in general, are seen for the smallest Reynolds number, these results are used to point out a few differences for the different wall treatments, see Fig. 17. The NnA treatment displays the largest deviations around $r/D = 1$ for both shear stress and heat transfer. This is no surprise as, in this region, the advection is not negligible when the flow is changing direction and NnA does not contain any advection terms. It should be stated that the diffusion model from Bond and Blottner, which NnA tries to resemble, was not developed for advective flows. With this in mind, it is more astonishing that this model recovers and follows the reference closely for $r > 2r/D$. In contrast, the results of the UMI model agree with the reference very well from the stagnation point to $2r/D$. Further downstream, the results deviates from the reference case, especially for the heat transfer. As UMI only solves for the velocity component parallel to the wall, it is obvious that it is important to solve for all velocity components to ensure a well predicted direction of the mass flux further away from the stagnation zone.

Craft et al. (2004) demonstrated similar accuracy and better speed-up for the original UMIST-N model which differs from the UMI model in several aspects, see Tables 1 and 2. However, the most important factor for the difference in speed-up is the lack of robustness for the UMI model which resulted in a speed-up penalty when using a larger number of grid cells to reach convergence. Nevertheless, the combined accuracy and speed-up re-

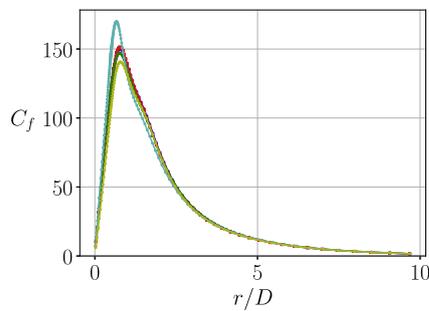
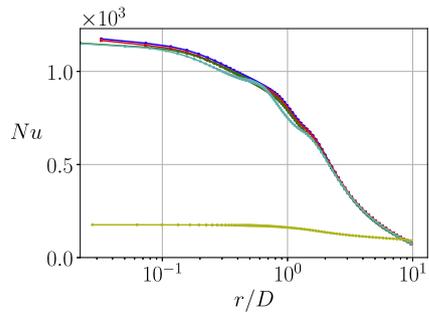
(a) Skin friction, C_f , plotted along the impinging wall.(b) Heat transfer, Nu , plotted along the impinging wall.

Fig. 19. $Re_D = 220,000$, AMS is applied to the pipe wall for all set-ups using an LRN model and different near-wall strategies are applied to the impinging wall, comparing; —: AMS, —: NFF, —: NWF, —: NnA, and —: HRN.

ported for the UMIST-N model is impressive. The results of the NFF and the NWF treatment are already commented in Section 3.4 but agree with the reference far better than the other treatments investigated.

4. Concluding remarks

Two new numerical wall functions, independent of the turbulence model, have been implemented in the open source CFD package OpenFOAM. They use a *sub-grid* which overlaps the wall-adjacent cells in an ordinary *main-grid*. The coupling between the two grids uses only face fluxes or wall fluxes, giving a stringent mathematical implementation from a finite volume perspective. Continuity on the sub-grid is obtained by ensuring the mass flux over the faces overlapping a main-grid face is equal to that of the main-grid face' before cell internal mass flux in the sub-grid is updated to give a divergence-free velocity field in each and every cell of the sub-grid. The results show excellent agreement with default low-Reynolds-number calculations, while the computing requirement is somewhere between the requirements of high-Reynolds-number models with wall functions and those of low-Reynolds-number models.

Compared to earlier numerical wall functions (Craft et al., 2004; Gant, 2002; Myers and Walters, 2005; Chedevergne, 2010; Bond and Blottner, 2011), presented in the introduction, the present numerical wall functions are the first ones not being restricted to a specific turbulence model due to the use of face fluxes for the coupling of the turbulent entities together with the implementation in the highly modular CFD package OpenFOAM. This independence makes it very general, and should work directly with any low-Reynolds-number model constituted by transport equations.¹⁰

¹⁰ It should also work with any extra transport equation of any scalar or vector. The present numerical wall functions could also be adapted to a low-Reynolds-number model, which includes other type of equations, e.g. elliptic ones.

Earlier methods have instead used Dirichlet boundary conditions with interpolated values from the main-grid, together with averaged volume sources of the turbulent production and dissipation terms from the sub-grid, for the coupling between the two meshes. The new wall functions are also the first ones to ensure continuity on the sub-grid, which is the most plausible reason for being very robust for a wide range of Reynolds number and insensitive to changes in size of the near-wall main-grid cells.

However, an even more robust set-up was found using a standard low-Reynolds-number model with an optimized mesh that relaxes the general recommendations for grid generation of the near-wall resolution. This finding, resulting in similar speed-up as the implemented numerical wall functions, proves that the major calculation cost of the boundary layer is the diffusion and not the continuity for steady-state flows. To summarize, a number of inferences can be drawn:

- for complex flow, HRN, especially for heat transfer
- new recommendations for grid design on near-wall resolution can drastically reduce the computing times required for LRN modelling for certain types of flows
- no advantages have been demonstrated with the face flux model compared to using the LRN turbulence model with an AMS mesh
- the numerical wall flux model, i.e. NWF, can give slightly larger speed-up but may suffer from robustness problems

The numerical wall functions, and the relaxed grid design recommendations, bring a sweet-spot between accuracy and computing time for the tested low-Reynolds-number model, which is beneficial for many turbulent flow problems, but especially for complex industrial cases. The results from the set-ups presented here are in excellent agreement with the investigated turbulence model, and will probably be for any low-Reynolds-number model. The computing requirement is up to an order of magnitude less than for a default set-up of a low-Reynolds-number model.

The most compelling strategy is of course the relaxed grid design recommendation, i.e. AMS, as this does not require any alteration of the CFD code. To increase the understanding of what speed-up can be expected using AMS for different flows, more testing is needed. Bäckar (2016) already tested the AMS approach on the backward-facing step, having an adverse pressure gradient and high wall-normal gradients relatively far from the wall across the separation bubble. Thus, it is difficult to divide this set-up into two distinct regions with high and low gradients, respectively. The resulting speed-up was only around 30%.

Acknowledgements

This study has been funded by Volvo Technology Development AB partly together with the Swedish Energy Agency [grant number 33973-1]. The discerning comments and the constructive suggestions from the anonymous reviewers resulted in significant improvement of the manuscript and are gratefully acknowledged.

References

- Baughn, J., Hechanova, A., Yan, X., 1991. An experimental-study of entrainment effects on the heat-transfer from a flat surface to a heated circular impinging jet. *J. Heat Transf. Trans. ASME* 113 (4), 1023–1025. doi:10.1115/1.2911197. WOS:A1991HP76100034
- Bond, R.B., Blottner, F.G., 2011. Derivation, implementation, and initial testing of a compressible wall-layer model. *Int. J. Numer. Meth. Fluids* 66 (9), 1183–1206. doi:10.1002/flid.2309.
- Bäckar, J.-A., 2016. Robust Numerical Wall Functions Implemented in OpenFOAM. Licentiate Thesis, Chalmers University of Technology, Göteborg.
- Chedevergne, F., 2010. Advanced wall model for aerothermodynamics. *Int. J. Heat Fluid Flow* 31 (5), 916–924. doi:10.1016/j.ijheatfluidflow.2010.05.002. WOS:000281918600017

- Cooper, D., Jackson, D.C., Launder, B.E., Liao, G.X., 1993. Impinging jet studies for turbulence model assessment I. flow-field experiments. *Int. J. Heat Mass Transf.* 36 (10), 2675–2684.
- Craft, T., Gant, S., Iacovides, H., Launder, B., 2004. A new wall function strategy for complex turbulent flows. *Numer. Heat Transf. Part B Fundam.* 45 (4), 301–318. doi:10.1080/10407790490277931.
- Gant, S.E., 2002. Ph.D. Thesis, *Development and Application of a New Wall Function for Complex Turbulent Flows*. Department of Mechanical, Aerospace and Manufacturing Engineering, UMIST.
- Iacovides, H., Launder, B.E., 1984. PSL—an economical approach to the numerical analysis of near-wall, elliptic flow. *J. Fluids Eng.* 106 (2), 241–242. doi:10.1115/1.3243109.
- Launder, B., Sharma, B., 1974. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc. *Lett. Heat Mass Transf.* 1 (2), 131–137. doi:10.1016/0094-4548(74)90150-7.
- Myers, S., Walters, D.K., 2005. A one-dimensional subgrid near wall treatment for turbulent flow CFD simulation. In: *Proceedings of the American Society of Mechanical Engineers, Fluids Engineering Division (Publication) FED*, 261, pp. 577–585. FED.
- OpenFOAM®, 2015. Version 2.3.x.
- Patankar, S.V., Spalding, D.B., 1972. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transf.* 15 (10), 1787–1806. doi:10.1016/0017-9310(72)90054-3.
- Schwer, L. E., 2009. *Guide for verification and validation in computational solid mechanics*. Tech. Rep., ASME.
- Spalart, P.R., 2000. Strategies for turbulence modelling and simulations. *Int. J. Heat Fluid Flow* 21 (3), 252–263. doi:10.1016/S0142-727X(00)00007-2.
- Wald, J.-F., 2016. *Lois de paroi adaptatives pour un modèle de fermeture du second ordre dans un contexte industriel*. Ph.D. Thesis, de l'Université de Pau et des Pays de l'Adour.
- Yap, C.R., 1987. *Turbulent Heat and Momentum Transfer in Recirculating and Impinging Flows*. University of Manchester, United Kingdom. Ph.D. Thesis.