Turbulent Flow Simulation for a Sharp Immersed Boundary Method

Congcong Chen  
chencongcong@buaa.edu.cn  
Beijing, China

Xiaofeng Sun  
sunxf@buaa.edu.cn  
Beijing, China

Zhuo Wang  
wzhuo@buaa.edu.cn  
Beijing, China

Lin Du  
lindu@buaa.edu.cn  
Beijing, China

ABSTRACT

In this paper, a wall function is combined with the sharp immersed boundary method (IBM) for turbulent flow simulation on Cartesian grid. The wall function derived directly from Spalart-Allmaras (S-A) turbulence model or its modified version is employed to determine proper tangential velocity boundary condition. Several two-dimensional (2-D) cases, such as non-inclined / inclined flat plate, zero attack NACA 0012 airfoil, are investigated to check the accuracy of the present IBM. Although the skin friction coefficients are very difficult to predict, the surface pressure coefficient of the airfoil can be calculated accurately even with a coarse grid, in which the \( y^+ \) of the reference point defined at leading edge is about 501.9. Compared to the original wall function, the modified wall function achieves a better grid convergence. The validation cases of plate and NACA 0012 have shown the primary ability of the present IBM to treat attached flow.

INTRODUCTION

Although body-conformal grids, structured or unstructured, have been widely used in Computational Fluid Dynamics (CFD) to simulate flows around arbitrary bodies, the automatic grid generation or reconstruction is still a big challenge when dealing with complex or moving geometry. Therefore, the Cartesian grid methods, which can generate the grid automatically and rapidly around complex solid boundaries, have come back to the lime-light. In the simplest Cartesian grid method, boundaries are generally approximated as a series of staircase-like steps, which would introduce deviations into the solution (Ferziger and Peric, 2012). In order to reproduce the wall boundary smoothly and accurately in Cartesian grid solvers, three primary methods have been applied: the hybrid grid method (Charlton, 1997, Wang, 1998), the cut-cell method (Lee and Ruffin, 2007, Harada et al., 2017) and the immersed boundary methods (Mittal and Iaccarino, 2005, Du et al., 2014, Du and Sun, 2015). When the layer grids are used along with the Cartesian grids in a hybrid grid method, the boundary layer is easily treated as commonly used body-conformal grids. However, the laborious grid generation and complicated coordinate transformation and interpolation are required again (Lee and Ruffin, 2007). The cut-cell methods cut the cells that intersect with wall surfaces. The cut-cell method can track the boundary as exact as body-conformal grids, whereas the calculation of the intersection is complicated, and special treatments such as cell-merged techniques are required to handle the small cells in the vicinity of the geometry (Udaykumar et al., 2001). On the other hand, the IBM uses only orthogonal cells, and sharp interface can also be achieved straightforward (Du et al., 2014, Majumdar et al., 2001), it is a simple and robust method.

Adaptive grid refinement technique has been proved to be a promising approach to enhance the local grid resolution only at interested regions, such as near the boundary or with high gradient, and have been validated in incompressible / compressible laminar flow. With the help of local grid refinement, Tullio et al. (de Tullio et al., 2007) have also proved that simulation of a turbulent boundary layer is possible if the minimum cell size is smaller than the height of the viscous sublayer. However, the isotropic property of Cartesian grid will significantly increase near wall grid density, which will become fatal when the Reynolds number becomes higher. Therefore, two distinct wall treatments have been proposed to alleviate the grid resolution constraint of High-Reynolds simulation.

The first treatment is to take advantage of the wall function methods to bridge the near wall region to core region of the turbulence with analytical expressions, which are derived from turbulent boundary layer function. Ghosh et al. (Ghosh et al., 2010) used a power-law expression along with a discrete continuity equation to reconstruct the velocity vector at near wall grid. Lee and Ruffin (Lee and Ruffin, 2007) used Spalding’s formulation, which yields a unified form valid for the log law layer and the viscous sublayer as well as the buffer layer, to determine the tangential velocity at the ghost cell. Kalitzin et al. (Kalitzin et al., 2005) suggested to build adaptive wall functions based on well-resolved numerical look-up tables for each individual turbulent model to ensure consistency between the eddy-viscosity and the velocity gradient. Recent years, in the researches team of University of Tokyo (Tamaki et al., 2017, Harada et al., 2017), following the advice of Kalitzin, a new SA wall model (Allmaras and Johnson, 2012) is served as a
specific wall function for the SA turbulence model and a linear relation between local modified eddy-viscosity $\tilde{\nu}$ and non-dimensional length $y'$ is used to define the boundary condition for the turbulent simulation.

The second treatment is the two-layer wall models, which are originally developed to replace the near-wall region with a wall model (a set of boundary-layer equations) to provide approximate wall boundary conditions for outer large eddy simulation (LES) in body-conformal grid. Tessicini et al. (Tessicini et al., 2002) have validated the applicability of a simple near-wall model, based on the local equilibrium hypothesis, in the framework of IB method for LES. Capizzano (Capizzano, 2011) have solved the Reynolds-averaged Navier–Stokes (RANS) equations on external region and simplified layer equations (Wang and Moin, 2002) on embedded virtual subgrid that originates from the forcing point down to the solid surface near the wall. A slip velocity has been assigned to ensure the correct wall-shear. Indeed, the two-layer wall model is expected to be more accurate in the application of separated flow and unsteady flow, on account of the capability of boundary-layer equations in considering the effect of unsteady and convection.

In order to simulate turbulent flows in Cartesian grid, a sharp immersed boundary method using a wall function to describe the boundary layer has been developed. Firstly, the numerical method employed will be discussed, including the governing equations, grid generation and wall function implementation. Then, three 2-D cases will be studied to validate the sharp IBM, including the non-inclined plate, inclined plate and the NACA 0012 airfoil with zero attack. Both the distribution of skin friction coefficient and the profile of tangential velocity are investigated by using different wall functions. Finally, a comprehensive conclusion for the sharp IBM is given.

NUMERICAL METHODS

In the past few years, our group (Du et al., 2014, Du and Sun, 2015) has validated our sharp interface IBM thoroughly for simulating both stationary and moving boundary problems (i.e. vortex-induced vibration) in compressible flow using stretched grid. Nowadays, in order to extend our inhouse code’s ability in handling complex geometry and high-Reynolds flow, wall function method have been investigated.

Governing equations

The 2-D unsteady compressible Reynolds-averaged Navier–Stokes equations are written in terms of conservative variables:

$$\begin{align*}
\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_j} &= 0, \\
\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial (\rho u_i u_j)}{\partial x_j} + \frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} &= 0, \\
\frac{\partial E}{\partial t} + \frac{\partial (E u_i + p u_i)}{\partial x_j} - \frac{\partial (\tau_{ij} u_j - q_i)}{\partial x_j} &= 0,
\end{align*}$$

where $\rho$ represents the density, $p$ represents the pressure, $u_i$ represents the velocity component in the $i$-th coordinate direction ($i = 1, 2$), $E$ represents the total energy per unit volume. The total energy is given by:

$$E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho u_i u_i,$$

where $\gamma$ denotes the specific heat capacity ratio. The total stress tensor $\tau_{ij}$ is the sum of the viscous and the Reynolds stress tensor:

$$\tau_{ij} = 2\rho(\nu + \nu_S)\delta_{ij}, \quad \nu_S = \frac{1}{3} \nu_{ib} \delta_{ij},$$

where the Boussinesq hypothesis is applied through the introduction of the eddy viscosity $\nu_S$ , and where $\nu_{ib} = (u_{ib} + u_{ib}) / 2$ is the strain-rate tensor, $\nu$ the molecular viscosity, depending on temperature $T$ through Sutherland’s law. Similarly, the total heat flux $q_i$ is the sum of a molecular and a turbulent contribution:

$$q_i = -\rho c_p \left( \frac{\nu + \nu_S}{\nu} \right) \frac{\partial T}{\partial x_i},$$

where the production term, destruction term and diffusion term are defined as:

$$\text{(Production)} - \text{(Destruction)} + \text{(Diffusion)},$$

and

$$f_{\text{x}} = c_{f_1} \left( 1 - f_{\text{d}} \right) \rho \tilde{S} \tilde{\nu},$$

$$f_{\text{d}} = c_{f_3} \text{exp} \left( - c_{f_4} \tilde{\chi}^2 \right)$$

with $d$ refer to the wall distance of the grid. The other constants in the model are as follow:
Grid becomes finer when it gets closer to the wall boundary without increasing grid density at other area. Detail of the related algorithms and validations can be found in the work of Wang (Wang et al., 2019). Here, a brief instruction of the grid generation will be given.

![Wall Boundary](image)

**Figure 2** Adaptive grid refinement around arbitrary wall. The solid line in red represents the wall boundary.

The grid system consists of various Cartesian grid level with varying resolution. Assuming that $L^i$ and $\Delta x^i$, $i = 0, 1, 2, \ldots$, represent the individual grid level and their corresponding grid size, respectively:

$$
\Delta x^i = \frac{\Delta x_0}{2^i},
$$

(9)

Where the subscript index 0 represent the initial background Cartesian grid, with $\Delta x_0$ being the corresponding grid size. $G$ is the set of whole grid system, it can be defined as:

$$
G = \sum_{i=0}^{L^i}.
$$

(10)

Firstly, a coarsest structured Cartesian grid level, $L^0$, is initialed to cover the whole computational domain, and the total grid number of $L^0$ is $N^0$. Then, the off-wall distance of each grid node $d_i^0$ ($i = 1, 2, \ldots N^0$) is calculated and a portion of grids which fulfill the given criterion, $d_i^0 < D^0$, is marked. The new refined level, $L^1$, is generated by bisection of the marked $L^0$ grids at each direction. This refinement process is done iteratively, until an assigned target value is reached at the immersed boundary. Fully unstructured arrays are employed to store the locations, connection and refinement relations and physical variables of the whole refined grids. A time refinement is also proposed to get a constant CFL number in different grid refined layer:

$$
\frac{\Delta t^i}{\Delta x^i} = \text{constant},
$$

(11)

Where $\Delta t^i$ is the time step used on $L^i$.

**Immersed Boundary Method**

Present sharp IB method is originally developed for low-Reynolds flow (Du et al., 2014), in which the velocity boundary condition is imposed implicitly by reconstructing the velocity vector at the boundary points $P_h$ as shown in Figure 3.

**Grid Generation**

The adaptive grid refinement method based on the given distance criterion, as shown in the Figure 2, has been developed to replace the original stretched grid by our group.
The line $P_l P_i$ is across point $P_0$ and perpendicular to the solid surface at point $P_i$. Point $P_i$ is surrounded by four fluid grids $A_{1,2,3,4}$, from which the bilinear interpolation can be used to obtain the velocity vector of $P_i$:

$$V_{\iota} = \sum_{i} \delta(i) \left( \frac{d_{\iota}}{\Delta x_{\iota}} \right) \left( \frac{d_{\iota}}{\Delta y_{\iota}} \right) V_{\iota},$$

where the delta function is defined as:

$$\delta(x) = \begin{cases} 1 - |x|, & |x| < 1 \\ 0, & |x| \geq 1 \end{cases},$$

here, $\Delta x_{\iota}$ denotes the grid size near the wall boundary, $d_{\iota}$ and $d_{\iota}$ represent distances between point $P_i$ and $i$-th adjacent grid in the $x$- and $y$- direction, respectively. Then, the velocity at $P_i$ can be calculated through the linear interpolation:

$$\overline{V_{\iota}} = \frac{l_1}{l} \overline{V_{\iota}} + \frac{l_2}{l} \overline{V_{\iota}},$$

where $l_1$ is the distance between points $P_0$ and $P_i$, and $l_2$ is the distance between points $P_i$ and $P_n$, $l = l_1 + l_2$, which is equal to $3 \Delta x_{\iota}$.

Although a similar interpolation procedure with more interpolation points has been validated by Tullio et al. (de Tullio et al., 2007) to simulate high-Reynolds flow, the linear interpolation can only be utilized in the viscous sublayer, this is a too strict confinement for the IBM, especially for the flow simulations involve complex geometries with curved and planar boundaries oblique to the grid. Therefore, wall function methods have been introduced to replace the original interpolation procedure for turbulent flow. Although the wall function always assumes incompressible wall, it can be extended to a compressible flow directly if the freestream Mach numbers is smaller than 1.4, which had been test numerically by Knopp et al. (Knopp et al., 2006).

**Original Wall Function**

Figure 4 shows the schematic of the wall function method. Here, only the fluid cells (marked with solid circle) are solved directly from Eq. (1) and Eq. (5) and the boundary condition is imposed implicitly onto the boundary points. For each boundary point $P_i$, a corresponding reference point $P_{r_i}$ is set in the flow region. The location and state vector of the reference point $P_i$ are calculated by the same procedure illustrated in the laminar case. After getting the velocity vector of $P_i$, a coordinate transformation is employed to obtain the tangential $u_{t,r_i}$ and normal velocity $u_{n,r_i}$. Then, substitute the $u_{t,r_i}$ and wall distance $l = l_1 + l_2$ into Eq. (8):

$$u_{t,r_i} = f_{sa} \left( y_{r_i}^+ \right), \quad u_{n,r_i} = \frac{u_{t,r_i}}{u_t}, \quad y_{r_i}^+ = \frac{u_{t,r_i}}{v},$$

where $f_{sa}$ represents the expression of Eq. (8). Then, $u_t$ can be solved using the Newton iterations. It is noted that a new expression of local Reynolds number $Re_y$ can be obtained by multiplying both sides of Eq. (8) with $y^+$:

$$Re_y = du_t / \nu = y^+ f_{sa} \left( y^+ \right) = F \left( y^+ \right),$$

inverting this function gives an explicit expression of $u_t$:

$$u_t = \frac{d}{\nu} F^{-1} \left( Re_y \right),$$

learning from the experience of Kalitzin et al. (Kalitzin et al., 2005), this inversion can be performed once and for all and store the inverse function in a look-up table generated with the Eq. (8). Then, the calculated $u_t$ is used to determine the tangential velocity at the boundary point $P_i$:

$$u_{t,r_i} = f_{sa} \left( y_{r_i}^+ \right) u_t, \quad y_{r_i}^+ = \frac{u_{t,r_i}}{v},$$

for the adiabatic boundary condition, the temperature of $P_i$ can computed from Crocco-Busemann formula:

$$T_{r_i} = T_{r_i} + \frac{u_t^2}{2 c_p} (u_{t,r_i}^2 - u_{t,r_i}^2).$$
the pressure of \( P_e \) is equal to that of \( P_i \):
\[
P_{p_0} = P_{p_i} .
\] (20)

Here, a linear interpolation is used to obtain the wall-normal velocity component of \( P_{p_i} \) for the non-penetrating boundary condition. Although different interpolation schemes had been adopted by researchers, such as: linear interpolation, bilinear interpolation and linear-quadratic (Majumdar et al., 2001), power-law (Ghosh et al., 2010), quadratic interpolation (Bernardini et al. 2016), none of the studies has shown that sophisticated interpolation schemes are significantly better than simple linear interpolation.
\[
\alpha_{0i} = \frac{1}{l} \alpha_{w,0i} (21)
\]

Finally, Eq. (7) is used to calculate the \( \sigma \) of \( P_{p_i} \):
\[
\sigma_{p_i} = \kappa \alpha_{p_i} l_i (22)
\]

**Modified Wall Function**

Tamaki et al. (Tamaki et al., 2017) has indicated that the unbalanced flux between the fluid cell and boundary cell is the primary factor to cause the oscillation in the surface pressure and fail to predict the proper friction coefficient. In order to resolve the near wall velocity correctly and maintain flux balance, the 'slip' boundary condition is introduced with a modified eddy viscosity profile. This method is now adopted to combine with our IBM directly. In present work, the wall distance \( l \) of reference point \( P_i \) is served as the cut-off distance, below which a linear velocity profile is assumed. Meanwhile, the eddy viscosity profile under cut-off distance is also modified to be \( \alpha_i = \text{const.} \), maintaining the balance of shear stress.

The modified tangential velocity of point \( P_{p_i} \) is defined as:
\[
\alpha_{i,w} = \alpha_{w,0i} - \left\{ \frac{\partial f_{sw}}{\partial y^+} \right\} (y^+_i - y^+_e) \alpha_{i} ,
\] (23)

here, \( \partial f_{sw} / \partial y^+ \) represents the first derivative of the Eq. (8). The other variables of \( P_{p_i} \) is calculated from original formulae (18)-(21). But the shear stress at the grids below cut-off distance is obtained by:
\[
\tau = \rho u^+_i \alpha^+_i ,
\] (24)

to obtain the constant \( \alpha_i \) from a linear solution of \( \alpha_i \), the damping function \( f_{sw} \) in the S-A model is reconstructed as a new one \( f'_{sw} \):
\[
f'_{sw} = f_{swi} + f_{swi} (1 - f_{s}) ,
\]
\[
f_{swi} = \left( \frac{\chi}{\alpha_i} \right)^{1/2},
\] (25)
\[
f_{s} = 0.5 + 0.5 \tanh \left\{ c_r \left( \frac{1 - \frac{1}{r_s}}{r_s} \right) \right\}, \quad c_r = 5 ,
\]

where \( r_s = d_{p} / d \). The new damping function \( f'_{sw} \) is the product of \( r_s \) and original \( f_{sw} \) at \( P_i \).

**VALIDATION RESULT**

Several representative numerical results are presented here as the preliminary validation cases to illustrate the primary ability of the present IBM to treat attached flow.

**Flat Plate**

In this section, turbulent flow simulations are performed by solving the flows around a two-dimensional thin flat plate at Mach = 0.2 and Re = 1.0 x 10^6. To preliminarily investigate the accuracy of the results for the nonaligned grids, the flat plate is inclined by \( \theta \) (deg) relative to the grids. The orientation angle \( \theta \) is set to 0 and 15 degree, as shown in Figure 5. The chord length of the plate is \( c = 2.0 \), while the reference length for the Reynolds number is 1.0, and the thickness of the plate is 0.002 with a cylindrical leading and trailing edge. Both the inhouse IBM solver and a commercial CFD solver (i.e. Fluent) have been used to calculate this problem. The settings of the computational meshes are described in Table 1. In Table 1 (also the Table 2), the \( y^+ \) of IBM is calculated at the reference points, while for Fluent \( y^+ \) is defined at first off-wall cell. The refinement levels for the IBM is kept 5 for all cases. Both the original and modified wall functions have been used in the IBM.

<table>
<thead>
<tr>
<th>Table 1 Characteristics of the computational meshes for turbulent flat plate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fluent</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>IBM</td>
</tr>
<tr>
<td>( \alpha = 0^\circ ) (Case 1)</td>
</tr>
<tr>
<td>Coarse</td>
</tr>
<tr>
<td>Medium</td>
</tr>
<tr>
<td>Fine</td>
</tr>
<tr>
<td>( \alpha = 15^\circ ) (Case 2)</td>
</tr>
<tr>
<td>Coarse</td>
</tr>
<tr>
<td>Medium</td>
</tr>
<tr>
<td>Fine</td>
</tr>
</tbody>
</table>

**Case 1**

Figure 5a gives a sketch of this case. Figure 6 shows the distribution of skin friction coefficient \( c_f = \tau_{w} / (1/2 \rho \alpha^+_i) \) on the flat plate. The results obtained by modified wall function have shown the similar trend as that obtained by the original wall function. As shown in Figure 6a, an unexpectedly increase of \( c_f \) along the whole chord of plate is observed with the fine grid, \( y^+ = 110.7 \). The \( y^+ \) result of fine grid implies that the most of first off-wall grid is locating in the buffer sub-layer, in which a large gradient of velocity is existing and cannot be resolved correctly with a numerical scheme with second-order spatial accuracy (Tamaki et al., 2017).
Although small deviation exists as the grid resolution increasing, a relatively converged solution of $c_f$ can be obtained (Figure 6b) with the modified wall function in the whole calculated $y^+$ region. The grid convergence of nondimensional tangential velocity with different wall function are shown in Figure 7. The considerable agreement with the result of Fluent is achieved for both wall function with different grid resolution, but the performance of modified wall function is still better than that of the original one.

Figure 5 Illustration of flow simulation around a flat plate

Figure 6 Skin friction coefficient of plate with $\alpha = 0^\circ$

Case 2.

Figure 5b gives a sketch of this case. Figure 8a shows the results of skin friction coefficient obtained by original wall function. Compared to the result in Figure 6a, it is no surprised to find that a large deviation is emerged when plate is not aligned with the Cartesian grids. Similar results have also been reported by other researchers (Harada et al., 2017). What’s more, the grid convergence of the IBM with original wall function is destroyed completely when the plate is inclined. In contrast, modified wall function does not severely mispredict skin friction when the plate is inclined, and a notable converged result is obtained by the modified wall function, as shown in Figure 8b. The grid convergence of nondimensional tangential velocity for $\alpha = 15^\circ$ is presented in Figure 9. Similarly, a better grid convergence is achieved by using modified wall function.
In this section, turbulent flow simulations are performed by solving the flows around a 2-D low-subsonic NACA 0012 airfoil at Machₐ = 0.3 and Reₐ = 3×10⁶ with zero angle incidence, which has been calculated by Capizzano et al. (Capizzano, 2011). The body fitted results extracted from their work serve as the reference values in this section. In present work, a computational domain extending 40c away from the body is initially discretized using a uniform Cartesian with Δx₀ = 0.5c. Four different refinement levels are adopted, i.e. 8, 9, 10, 11 (represent as rf = 8, 9, 10, 11, respectively). The detail information of the calculated grid is summarized in Table 2.

Table 2 Characteristics of the computational meshes for NACA 0012 airfoil

<table>
<thead>
<tr>
<th>Refinement Level</th>
<th>Δx_{min}</th>
<th>y⁺</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1.95×10⁻³</td>
<td>501.9</td>
</tr>
<tr>
<td>9</td>
<td>9.76×10⁻⁴</td>
<td>288.7</td>
</tr>
<tr>
<td>10</td>
<td>4.88×10⁻⁴</td>
<td>166.1</td>
</tr>
<tr>
<td>11</td>
<td>2.44×10⁻⁴</td>
<td>86.4</td>
</tr>
</tbody>
</table>

Figure 8 Skin friction coefficient of plate with α = 15°

Figure 9 Tangential velocity of plate with α = 15°

NACA 0012

Figure 10 Distribution of Cₚ on NACA 0012

Figure 11 Distribution of C_f on NACA 0012
CONCLUSIONS
In order to simulate turbulent flow around complex geometries, a sharp immersed boundary method using finite difference algorithm has been developed. A new wall function directly derived from Spalart-Allmaras turbulence model or its modified version developed by Tokyo university research team is employed to avoid the sharp increment of the grid number required to resolve a turbulent boundary layer. The 2-D cases of the non-inclined / inclined flat plate and zero attack NACA 0012 prove the primary ability of the present IBM to treat attached flow. Whether flat plate is inclined or non-inclined, the IBM using a modified wall function can achieve converged results with the y’ defined at the reference point varying from 100 to 260. Although there still exist certain deviations in the results of modified wall function, it has demonstrated obvious advantages in the grid convergence and the accuracy of prediction, comparing to the original wall function. In the case of NACA 0012 with zero attack, the surface pressure coefficient of the airfoil can be calculated accurately even with a rather coarse grid, in which the y’ of reference point defined at leading edge is about 501.9. In order to further improve the calculation accuracy of IBM in turbulent simulations, more attention should be payed to maintain the conservation property at the interface around the immersed boundary, and the two-layer wall model should also be taken into consideration.

ACKNOWLEDGMENTS
Financial support from the National Science Foundation of China (Grants No. 51790514 and 51706006) is gratefully acknowledged.

REFERENCES


