Contents lists available at ScienceDirect



International Journal of Heat and Mass Transfer

journal homepage: www.elsevier.com/locate/ijhmt

A ghost-cell immersed boundary method for the simulations of heat transfer in compressible flows under different boundary conditions Part-II: Complex geometries



Kun Luo^a, Chaoli Mao^a, Zhenya Zhuang^a, Jianren Fan^{a,*}, Nils Erland L. Haugen^b

^a State Key Laboratory of Clean Energy Utilization, Zhejiang University, Hangzhou 310027, PR China ^b SINTEF Energy Research, N-7465 Trondheim, Norway

ARTICLE INFO

Article history: Received 12 May 2016 Received in revised form 6 July 2016 Accepted 8 August 2016

Keywords: Immersed boundary method Compressible flow Irregular boundary Heat transfer

ABSTRACT

In this paper, our previous ghost-cell compressible immersed boundary method (Luo et al., 2016) is further implemented to solve heat transfer problems in flows with complex solid geometries. Arbitrary 2Dimmersed boundaries are presented by many micro line segments. Each line segment is identified by two vertices. An extension to 3D situation is straightforward, in which arbitrary surfaces can be divided into many triangular surface elements. Two different interpolation schemes for the mirror points, namely inverse distance weighting and bilinear interpolations, are compared. An accurate capture of the secondary vortex street far behind an elliptical cylinder indicates a successful combination of current IB method with the fluid solver. Then, forced convective flow over an inclined non-circle cylinder is used to further validate present method. Finally, *Mach* > 0.3 cases are studied to demonstrate the essentiality of taking compressibility into consideration in high-speed thermal flow problems.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

The CFD (computational fluid dynamics) method has been a well-developed academic discipline and gradually become an effective instrument for engineering problems. When it comes to flows over complex geometries, in a traditional way, a bodyfitted mesh is generated to describe the boundary of the immersed body. Under this situation, the implementation of boundary conditions is simple and straightforward because grid line and body surface align with each other. However, for arbitrary complex geometries, the generation process of a high quality bodyconformal grid and its re-meshing process can be very resourceconsuming.

In contrast, a totally different idea had been introduced by Peskin [2] in 1972. That is "immersed boundary method (IB method)", in which a Cartesian grid was used to resolve blood flow regardless of the complex geometry of human heart valve. The effect of elastic heart valve wall on surrounding fluid flow was taken into consideration through a force term on the right-hand side of momentum equation. This idea successfully avoids the generation of an unstructured body-fitted grid to conform complex geometries and thereby makes simulation of complex structure-

* Corresponding author. E-mail address: fanjr@zju.edu.cn (J. Fan).

http://dx.doi.org/10.1016/j.ijheatmasstransfer.2016.08.010 0017-9310/© 2016 Elsevier Ltd. All rights reserved. fluid interaction more efficient. Moreover, as a result of using mesh of simple topology structure, the parallelization of code is straightforward and also more efficient.

Since then, IB methods have attracted many researchers' attention. Many efforts have been made to improve the accuracy and broaden the application. Generally, IB methods fall into two different categories, i.e., continuous force approach and discrete force approach [3]. A detailed discussion of IB methods can be found in [4–7]. IB method was originated to mimic the effect of elastic boundary on fluid flow and it made a sense that a force term was to represent the effect since the elastic force model in [1] had a physic basis. In a similar way, a PID (portion-integral-derivation) force model was presented for rigid boundary [8]. However, the free parameters included in this model may degrade numerical accuracy as well as stability. To overcome this downside, Fadlun et al. [9] proposed a direct-forcing scheme for rigid immersed body. They also showed that solving the interpolation formulas together with discrete momentum equations was equivalent to applying force term and then the explicit addition of force term was not required. This is where the original idea of ghost-cell based immersed boundary (GCIB) method comes from. Compared with original IB method [1] and direct-forcing method [10,11], no Dirac delta function is used to distribute the force term from Lagrange point to underlying Euler grid in GCIB method. Therefore, the boundary is sharply represented. This is a desirable feature to resolve boundary layer in high Reynolds number flows. Besides, since in GCIB method there is no need to modify the fluid solver (i.e., the implementation of boundary conditions through this method can be summarized into a separate module), its combination with existing solver is very easy. Another point worth of noting is that a high-order interpolation scheme can be constructed for GCIB method [12,13] to further save computational resource.

The difficulty in GCIB method's extension to situations with irregular geometries lies in how to track the boundaries correctly. Conceptually, two ideas exist to overcome this. An unstructured triangle surface mesh was used in [14–16]. This mesh can be used to represent arbitrary geometries and has gained its popularity in biology fluid mechanics where a very complex body, such as a bluegill sunfish pectoral fin and a false vocal fold, interacts with surrounding flows in a two-way coupled manner [17,18]. Another choice is a standard level-set signed distance function [19,20]. This strategy also applies to both rigid and deformable structures. We refer the reader to [14–20] and references therein for detailed descriptions of these two methods.

Application of IB methods to heat transfer problems was reported by Kim et al. [21], in which a heat source/sink was introduced into energy conservation equation to account for the effect of hot/cool boundary wall. Wang et al. [22] proposed a multidirect heat source scheme to improve the accuracy of boundary condition enforcement. Since then, many researchers have made their efforts to improve IB methods' capability to various heat transfer problems [23-29]. In our previous work [1], a secondorder accurate GCIB method was designed for the implementation of Dirichlet, Neumann and Robin boundary conditions. It also should be noted that our GCIB method was combined with a compressible fluid solver [30] and study on the effect of compressibility on heat transfer process was carried out. In this paper, we combine our previous method [1] with an unstructured surface mesh to devise a GCIB method for the simulation of heat transfer process between compressible fluid and irregular boundaries. To our best knowledge, no such report has ever been presented.

The rest part of the current paper is organized as follows. Section 2 gives the numerical methodology, including compressible governing equations, introduction to GCIB method, construction strategy for irregular geometry and two different interpolation procedures for mirror point. Section 3 starts with a test case where effect of the relative resolution between boundary and background grid is investigated. Following this is a spatial convergence examination to check if the present GCIB method still remains a second-order accuracy. After these, several benchmark cases are studied to validate our GCIB method's capability to handle irregular fluid–solid interface. Furthermore, compressibility effect in high speed forced convective flow is revealed. Finally, we draw a conclusion in Section 4.

2. Numerical methodology

2.1. Governing equations

Mass, momentum and energy conservation equations together with the equation of state are used to describe the compressible flows in present paper. The continuity equation reads as follows,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \, \vec{u}) = 0 \tag{1}$$

where ρ is fluid density, \vec{u} is the vector of fluid velocity and t is time.

In our research, the temperature ratio between solid and fluid is small. Therefore, we can assume properties of fluid such as dynamic viscosity μ , specific heat c_p at constant pressure and heat conductivity λ to be constant. Thus, the momentum and energy conservation equations can be simplified as,

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla(\vec{u}) = \frac{1}{\rho} \left(-\nabla p + \vec{f}_{vs} \right)$$
(2)

$$\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T = \frac{1}{\rho c_p} \left(\nabla \cdot (\lambda \nabla T) + 2\rho v S \otimes S \right)$$
(3)

In Eq. (2), *p* is pressure, $\vec{f}_{vs} = \nabla \cdot (2\rho vS)$ is viscous force, and $v = \mu/\rho$ is kinematic viscosity. The symbol *S* represents trace-less strain rate tensor with $S_{ij} = (\partial u_i/\partial x_j + \partial u_j/\partial x_i)/2 - \delta_{ij}\nabla \cdot \vec{u}/3$. The last term on the right hand side of energy equation is viscous dissipation source.

The state equation for an ideal gas to close Eqs. (1)-(3), is given by,

$$p = \rho RT \tag{4}$$

where, *R* is the specific gas constant and can be calculated from $R = R_u/M$. R_u is universal gas constant and *M* is molar mass.

The above governing equations are solved by a sixth-order centered finite-difference scheme on a non-staged mesh and an explicit three-stage Runge–Kutta scheme for spatial derivative and time advancement, respectively. The time step is limited by CFL number criterion. The sixth-order centered finite-difference scheme can be expressed as follows,

The first-order derivative:

$$f'_{i} = (-f_{i-3} + 9f_{i-2} - 45f_{i-1} + 45f_{i+1} - 9f_{i+2} + f_{i+3})/60\delta x$$
(5)
The second-order derivative:

$$f_i'' = (2f_{i-3} - 27f_{i-2} + 270f_{i-1} - 490f_i + 270f_{i+1} - 27f_{i+2} + 2f_{i+3})/180\delta x^2$$
(6)

where, δx is the local grid size.

To avoid "wiggles", the advection term in Eqs. (1)–(3) is discretized by a fifth-order upwinding scheme in which the point furthest downwind is excluded from the centered finite-difference stencil. The fifth-order upwinding stencil can be written as,

$$-uf'_{(up,5th)} = -uf'_{(centr,6th)} + \frac{|u|\Delta x^5}{60}f^{(6)}$$
(7)

where Δx is local grid size.

In order to construct the above upwinding scheme, the sixthorder derivative is needed. And it is straightforward to approximate such a derivative by Taylor expansion on a uniform mesh. However, when a stretched grid is used, things become complicated. In this paper, the following chain procedure is proposed to calculate the sixth-order derivative and thus to construct a fifthorder upwinding scheme on a stretched grid.

$$f'(\mathbf{X}) = f'(\zeta) / \mathbf{X}' \tag{8}$$

$$f''(\mathbf{x}) = f''(\zeta) / (\mathbf{x}')^2 - \mathbf{x}'' f'(\zeta) / (\mathbf{x}')^2$$
(9)

$$f'''(\mathbf{x}) = f'''(\zeta) / (\mathbf{x}')^3 - 3\mathbf{x}'' f''(\zeta) / (\mathbf{x}')^2 - \mathbf{x}''' f'(\zeta) / (\mathbf{x}')^3$$
(10)

$$f^{(4)}(\mathbf{x}) = \frac{f^{(4)}(\zeta)}{(\mathbf{x}')^4} - \frac{6\mathbf{x}''}{(\mathbf{x}')^2} f^{\prime\prime\prime}(\zeta) - \left[\frac{4\mathbf{x}^{\prime\prime\prime}}{(\mathbf{x}')^3} + \frac{3(\mathbf{x}'')^2}{(\mathbf{x}')^4}\right] f^{\prime\prime}(\zeta) - \frac{\mathbf{x}^{(4)}}{(\mathbf{x}')^4} f^{\prime\prime}(\zeta)$$
(11)

$$f^{(5)}(x) = \frac{f^{(5)}(\zeta)}{(x')^5} - \frac{10x''}{(x')^2} f^{(4)}(\zeta) - \left[\frac{10x'''}{(x')^3} + \frac{15(x'')^2}{(x')^4}\right] f'''(\zeta) - \left[\frac{5x^{(4)}}{(x')^4} + \frac{10x''x'''}{(x')^5}\right] f''(\zeta) - \frac{x^{(5)}}{(x')^5} f'(\zeta)$$
(12)

$$\begin{split} f^{(6)}(x) &= \frac{f^{(6)}(\zeta)}{(x')^6} - \frac{15x''}{(x')^2} f^{(5)}(\zeta) - \left[\frac{45(x'')^2}{(x')^4} + \frac{20x'''}{(x')^3} \right] f^{(4)}(\zeta) \\ &- \left[\frac{15(x'')^3}{(x')^6} + \frac{60x''x'''}{(x')^5} + \frac{15x^{(4)}}{(x')^4} \right] f^{'''}(\zeta) \\ &- \left[\frac{15x''x^{(4)}}{(x')^6} + \frac{6x^{(5)}}{(x')^5} + \frac{10(x''')^2}{(x')^6} \right] f^{''}(\zeta) - \frac{x^{(6)}}{(x')^6} f^{'}(\zeta) \end{split}$$
(13)

where, ζ is the index of grid point and *x* is a given function of ζ to stretch/compress the uniform mesh.

2.2. Ghost-cell immersed boundary method

In GCIB method, at first all grid nodes are flagged as fluid points, solid points or ghost points. Then variable value on ghost point is interpolated from boundary conditions and fluid variables near the boundary. Boundary conditions are implicitly incorporated into flow field through ghost points. Most often, ghost points are solid points which have at least one neighbor fluid point in the south/ north, east/west or top/bottom direction. To make sure that a discretization stencil of four or six-order is constructed, two or three layers of solid points near the immersed boundary should be used as ghost points [31-34]. However, in accordance with specific condition, ghost points can also be outside solid domain [14,35]. In the preliminary calculation, a simple linear extrapolation is used to obtain flow-variables at ghost points, which obviously lies outside the interpolation stencils. The major drawback of such extrapolation is that large and negative weighting coefficients are often encountered. Although algebraically correct, they can lead to severe numerical instability when coupled with flow solver. Introducing a mirror point inside the flow domain for each ghost point is a better procedure to ensure suitable weighting coefficients of the neighboring nodes [36]. The mirror point usually locates some distance away from the boundary in the boundary-normal direction. The distance can be either equal to that from ghost point to the boundary [37] or designated specifically [25,38]. The variable values on mirror point is interpolated from surrounding four (2D) or eight (3D) grid points, using a bilinear interpolation scheme (2D case) or a trilinear scheme (3D case) [15,33,39]. Inverse distance weighting (IDW) interpolation [40] can also be used. Once the variable values on mirror points are calculated, values of corresponding variable at ghost point can be computed by using a linear approximation along the boundary normal direction into which the prescribed boundary conditions are incorporated.

2.3. Boundary construction strategy

In the current method, the immersed boundary is represented by line segments (2D) or an unstructured surface mesh (3D). Surface mesh with triangle elements is a most popular choice due to the well simplicity and efficiency balance and also the seamless integration into GCIB method. This kind of triangulated surface can be used for an accurate representation of a wide variety of flow configuration which are of engineering interest.

In the first step, grid points that lie inside solid boundaries are found out and flagged as solid points. The rest are marked as fluid points. A straightforward method for the identification is schematically shown in 2D in Fig. 1. For a given grid point, the closest surface element is determined and then a dot product between the closest facet's normal vector and the direction vector extending from centroid of the closest facet to grid point under consideration is taken. In this step, the normal vector of every surface facet should be controlled to point outward when surface mesh is generated and in a reverse way for an inner flow problem. A negative dot product then indicates a solid points. To avoid the situation as



Fig. 1. Identification of solid points for 2D geometry, (\bullet) solid points, (\Box) fluid points, (\bigcirc) centroid of line elements, (\triangle) vertex of line elements.

shown in Fig. 2, after the assignment of all the grid nodes, for every grid points marked as "solid point", a dot product between the direction vector connecting it and the centroid of any micro line segments and the normal vector of the corresponding line segment is taken. Only those points always giving a negative result are actually identified as a "solid point". However, this strategy only works for a convex boundary such as those in present paper. For an arbitrary boundary, a more efficient but more sophisticated method, namely ray-method [41], is a better choice.

For stationary boundaries, this detection needs to be done only at the beginning of the simulation. While for a moving or deforming boundaries, the work needs to be done at every time step. The moving boundary is limited to travel within one grid size during one time step. With this constraint, an algorithm can be adopted to help save computational time and we only need to search over a square region as shown in Fig. 3 to repeat the classification of the grid nodes. The length and width if the rectangle search region can be expressed as,

$$Lx = x_{max} - x_{min} + 2\Delta x_{loc}$$

$$Ly = y_{max} - y_{min} + 2\Delta y_{loc}$$
(14)



Fig. 2. A confused situation for the classification of solid and fluid points.



Fig. 3. Sketch for the repeating search process with the moving interface, the black or red dash line rectangle indicating the search region. Solid line denotes the IB at n time-step and dash line is that at time-step n + 1.

in which, x_{max} and x_{min} indicate the *x* coordinate of the rightest and the most left point on the boundary, respectively. y_{max} and y_{min} indicate the *y* coordinate of the topside and the lowest point on the boundary, respectively. Δx_{loc} and Δy_{loc} mean the local grid size. Therefore, in both of the static and moving boundary cases, there is little contribution from the detection to the total CPU time.

Secondly, part of solid points are marked as ghost points. Ghost points are those solid points which are not far away from the boundary (when the background grid is fine enough) and have a neighbor fluid point. To construct a sixth-order central finite difference stencil, three layer of ghost points are assigned here, as shown in Fig. 4. Since the proportion of ghost points is small, more layer of ghost points will contribute very little to the computation time. For example, in the square cylinder cases below, the number of grid nodes being used is 1440000 for a 20D * 20D computational domain and the number of ghost points, the incremental percentage



Fig. 4. 2D schematic diagram for ghost points and the method to assign mirror points, (\blacksquare) ghost points, (\bullet) BI points, (\oplus) mirror points.

of computational time by using three layer ghost points is only 492 * 2/3/1440000 * 100% = 0.0228%.

After the assignment of ghost points, a corresponding boundary intercept (BI) point is determined for every one of them. While conceptually simple, the implementation can be very complicated and special attention is needed to find an exactly correct BI point. Here, we adopt a method building on the robust procedure proposed in [15].

First, the vertex closest to a given ghost point is determined. Then the set of line/surface elements sharing that vertex can be identified and a search is carried out among these elements to find the projective point (which should lies within the line/surface elements) as BI point (see Fig. 4). For a 2D geometry, there are two line elements surrounding one vertex. This number is case dependent for a triangulated 3D surface mesh. The above logic can be utilized to make the program more efficient and also easy to debug it. Several degenerate cases may arise in the search of BI point. For a convex body, the situation with multiple projective points is frequently encountered. Fig. 5(a) shows this for a 2D boundary and for a 3D case two more projective points may exist. In such a situation, the projection which has the shortest intercept is chosen as the BI point. We utilize two strategies in view of the situation where the multiple projection points are at comparable distances. For a square corner like that in Fig. 6(a), the strategy is that the variable value on ghost point GP is extrapolated from mirror point MPx and the corresponding boundary condition for the calculation of the x direction finite difference stencil at fluid point FP1 while for the approximation of the y direction finite difference stencil at fluid point FP2, mirror point MPy and the corresponding BI point are used to compute the variable value on ghost point GP. For a triangular sharp corner like that in Fig. 6(b), the extrapolated values from mirror point MP1 and MP2 are averaged to obtain that on ghost point GP.

For an indent immersed boundary, situation where there doesn't exist a projective point within the element often appears, as described in Fig. 5(b). When this happens, the search is first expanded to a larger region into which line/surface elements that share a common vertex with the original set of elements are included (see Fig. 5(b)). If it fails again, then the closest point on the original set of elements is marked as the BI point. Coordinates of the projective point with the shortest intercept is first calculated and then the region it falls into determines where we can find the closest point. For a 2D boundary, the nearest vertex determined at the beginning is the closest point.

Once BI point is determined for every ghost point, a corresponding mirror point can be obtained uniquely through a symmetric way or setting a constant distance δ away from BI point, as shown in Fig. 4. The value of δ can be assigned as $\sqrt{2}\Delta x_{loc}$ (2D case) or $\sqrt{3}\Delta x_{loc}$ (3D case) to make sure that even the first layer of mirror points are surrounded by four (2D) or eight (3D) fluid points. The drawback of the constant distance method is that a big Δ is chosen, and thus some error may be introduced, especially for a turbulent boundary layer. Therefore, the first method is adopted in this paper. The calculation of variable values on mirror points is illustrated in the following section.

2.4. Interpolation procedure

Two interpolation approaches are presented here. The first one is called "inverse distance weighting interpolation (IDW)" [40]. In this method, a generalized flow quantity on mirror point is represented by surrounding fluid points in the following way (see Fig. 7),

$$\phi_m = \sum_{i=1}^{N} \frac{\phi_i (1/d_{i,m})^2}{\sum_{i=1}^{N} (1/d_{i,m})^2}$$
(15)



Fig. 5. Two degenerate situations, (a) multiple projective points, (b) no projective point within the element.



Fig. 6. Special treatment for situations where multiple projection points are at comparable distances.



Fig. 7. IDW interpolation procedure.

When a ghost point is close to the immersed boundary, the corresponding mirror point may not have four/eight surrounding fluid points. In such a situation, only contribution from fluid points is taken into consideration. In Eq. (12), if any d tends to zero which also means that the mirror point is very close to a fluid point, the corresponding grid point is used to replace that mirror point to avoid numerical issue. It is observed in present paper that this treatment leads to a first-order local spatial accuracy.



Fig. 8. Local pressure coefficient around the cylinder.

Another interpolation scheme of mirror point is known as bilinear interpolation (2D case) or trilinear interpolation (3D case). Taking the 2D one as an example, variable values on mirror points can be written as,

$$\phi_m = C_1 x y + C_2 x + C_3 y + C_4 \tag{16}$$

The four unknown coefficients can be evaluated by solving the following equation using Gaussian elimination method,

$$A\mathbf{C} = \Phi \tag{17}$$

where,



Fig. 9. Variation of *L*₂ with grid of different resolution levels.

$$\mathbf{C} = \{C_1, C_2, C_3, C_4\} \tag{18}$$

is the vector of unknown coefficients and,

$$A = \begin{bmatrix} x_1y_1 & x_1 & y_1 & 1 \\ x_2y_2 & x_2 & y_2 & 1 \\ x_3y_3 & x_3 & y_3 & 1 \\ x_4y_4 & x_4 & y_4 & 1 \end{bmatrix}$$
(19)

is the coordinates-related matrix. The vector Φ includes the known generic variable values on surrounding fluid points, that is,

$$\Phi = \{\phi_1, \phi_2, \phi_3, \phi_4\}$$
(20)

If there is a non-fluid point surrounding the mirror point, its BI point is used to replace it, with matrix A and vector Φ modified accordingly.

In bilinear interpolation scheme, when mirror point's surrounding points are all fluid ones, the treatment of Neumann and Robin boundary conditions are the same with that for a Dirichlet one as shown above. However, situation becomes complicated when one or more of them are not fluid points. For simplicity, we assume that the last node is a non-fluid one. Then, the matrix *A* becomes,



Fig. 10. Wake behavior far behind cylinder after the flow gets to a dynamical steady state, (a) Ar = 0.5, Re = 150, (b) Ar = 0.25, Re = 150, (c) Ar = 0.25, Re = 100.

$$A = \begin{bmatrix} x_1 y_1 & x_1 & y_1 & 1 \\ x_2 y_2 & x_2 & y_2 & 1 \\ x_3 y_3 & x_3 & y_3 & 1 \\ n_1 y_{BI} + n_2 x_{BI} & n_1 & n_2 & 0 \end{bmatrix}$$
(21)

where the vector $\vec{n} = (n_1, n_2)$ denotes the unit normal vector of the line segment. Accordingly, the vector Φ is as follows,

$$\Phi = \{\phi_1, \phi_2, \phi_3, \phi\}$$
(22)

The symbol φ indicates the pre-set constant heat flux through the line segment. For a Robin boundary condition in the following form,

$$\alpha \left(\frac{\partial \phi}{\partial n}\right)_{BI} + \beta \phi = \varphi \tag{23}$$

We have,

$$A = \begin{bmatrix} x_1y_1 & x_1 & y_1 & 1\\ x_2y_2 & x_2 & y_2 & 1\\ x_3y_3 & x_3 & y_3 & 1\\ X_1 & X_2 & X_3 & X_4 \end{bmatrix}$$
(24)

With

$$\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} = \begin{bmatrix} \alpha(n_1 y_{BI} + n_2 x_{BI}) + \beta x_{BI} y_{BI} \\ \alpha n_1 + \beta x_{BI} \\ \alpha n_2 + \beta y_{BI} \\ \beta \end{bmatrix}$$
(25)



At last, a linear interpolation in which a Dirichlet, Neumann or Robin boundary condition is implicitly involved, is used for the calculation of flow-variables at ghost points, as has been discussed in our previous work [1].

3. Numerical results

2.0

3.1. Effect of relative resolution

For a given mesh, the number of ghost points remains constant as the number of line segments (triangulated surface elements) reaches a critical value. However, the division of boundary/surface may change the normal direction of every line segment (or face element). This has an influence on the identification of mirror points and therefore affects the implementation of boundary conditions. Besides, a non-isotropic mesh is ordinarily generated to resolve a sharp corner or other finer features of the immersed boundary. Hence a test case is carried out here to investigate the effect of relative resolution between spatial grid and the surface mesh.

An isothermal flow in a domain of size $20D \times 20D$ with a fixed circle boundary immersed in is utilized for this purpose. Partially reflected Navier–Stokes characteristic boundary conditions (NSCBC) [42] are applied at both inlet and outlet. Periodic boundary conditions are used for spanwise direction. And on the solid surface, non-slip (i.e., fixed velocity) and non-impermeable boundary conditions (i.e., zero-gradient pressure) are enforced. Reynolds number is chosen to be Re = 40. Three cases, with the following



Fig. 11. Distribution of velocity component u, (a): along x axis at y = 0.0, (b): along y axis at different x positions (x = 2,4 and 6), Re = 150, Ar = 0.25, (c): Re = 100, Ar = 0.25, (b): Re = 150, Ar = 0.5.

104

relative resolution (the background grid resolution is fixed to be $\Delta_g = D/70$), $\Delta_b/\Delta_g = 0.5$, $\Delta_b/\Delta_g = 1.0$ and $\Delta_b/\Delta_g = 2.0$, where Δ_b means the boundary resolution (i.e., the length of the micro line segments) and Δ_g means the resolution of the underlying grid, are considered. Results are presented after the flow gets to a steady state.

The surface pressure coefficient, which is defined as $c_p = (p - p_{\infty})/(1/2\rho U^2 d)$, is shown in Fig. 8. Computational results with the three relative resolutions are not very different from one another and agree well with the reference data [43].

The reason for above minimal distinction may be that a simple flow is simulated here. In such a case, fluid variable varies in a linear way everywhere and so the relative resolution has a little effect as long as the spatial grid is fine enough. Even though, it's obvious that an around one relative resolution is appropriate for the following simulations.

3.2. Spatial convergence examination

An examination is conducted in this section to check if this new IB method remains a local second-order spatial accuracy. Two interpolation procedures for mirror points are compared. The immersed boundary is also a circle centered at the origin. The computational domain is of size $10D \times 10D$ and Reynolds number is set

to Re = 20. Boundary conditions are the same with that in Section 3.1. A series of uniform grids $(200 \times 200, 400 \times 400, 800 \times 800)$ are utilized to compute the same flow. The L_2 norm number expressed as follows is used to evaluate the rate of spatial convergence,

$$L_{2} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\phi_{i} - \phi_{d})^{2}}$$
(23)

Here *N* is the total number of line segments (sampling points are middle points of the line segments) and ϕ_d is the desirable variable value on immersed circular cylinder. For Dirichlet boundary condition, ϕ_d is prescribed. While for Neumann or Robin boundary condition, ϕ_d is calculated with the finest background grid.

Fig. 9 shows the logarithm result of L_2 norm. As can be seen, the bilinear interpolation procedure exhibits a second-order accuracy, while the IDW procedure is of only first-order accuracy. Reason may be that in IDW procedure contribution from non-fluid point to the calculation of variable value on mirror point is discarded. From Fig. 9, we can also know that for bilinear interpolation the temperature L2 error norm is about 0.007334 when $\Delta_g = D/60$, which means that the resolution $\Delta_g = D/60$ is enough in terms of L2 error norm. Therefore, in the following example cases, a $\Delta_g = D/60$ grid resolution is utilized for spatial discretization.



Fig. 12. Corresponding temperature contour, (a) Ar = 0.5, Re = 150, (b) Ar = 0.25, Re = 150, (c) Ar = 0.25, Re = 100.

3.3. Wake behavior behind an elliptical cylinder

The wake behind an ellipse cylinder in [44] is studied to validate current IB method in a qualitative way. In this case, the elliptical cylinder is located in a $60D \times 20D$ rectangle domain with a grid resolution $\Delta_g = D/60$. Notably, energy equation is coupled in this paper, which is a supplement to [44]. As shown in Fig. 10, far behind the cylinder with an aspect ratio (Ar) 0.5, two separating shear layers form, while the unstable shear layer develops into a secondary vortex street with Ar decreasing to 0.25. When Reynolds number decreases to 100, the amplitude of the secondary vortex street becomes narrow. More energy is fed into the wake by a strong recirculation due to a sharp geometry, leading to the secondary vortex street. Our results qualitatively accord well with that in [44]. Characteristics of this transition flow behavior is detailed in Fig. 11 in terms of a dimensionless velocity. Distinguish among the three situations is clearly displayed. Accordingly, temperature contour is presented in Fig. 12. Comparing Fig. 12 with Fig. 10, a synchronization between the vortex and high temperature region can be observed.

3.4. Forced convective flow over a non-circular cylinder

To validate present method's capacity to deal with complex boundaries, a forced convective problem is studied in this section. Dirichlet thermal boundary condition is imposed on the immersed boundary. The cylinder is kept inside a box of size $20D \times 20D$, having a uniform grid of resolution $\Delta_g = 1/60$ in the unit of the diameter of semi-circle or the edge of a square. The same computational domain boundary conditions as employed in Section 3.1 are used here. Prandtl number is kept to Pr = 0.70. In current calculation, the projected width of the cylinder in the spanwise direction is taken as a characteristic length, e.g., in the calculation of Reynolds number.

In the square cylinder case, three different attack angles with respect to stream-wise direction are considered at the same Reynolds number Re = 100. We neglect the effect of natural convection, so that *Ri* (Richardson number) is equal to zero here [45]. Fig. 13 describes the instantaneous vorticity and temperature contour, indicating that details of flow field resulting from sharp corner are well revealed. The computed time averaged drag and lift coefficient ($\overline{C_D}$ and $\overline{C_L}$), time averaged Nusselt number (\overline{Nu}) and *Strouhal* number (dimensionless vortex shedding frequency, Str) are listed in Table 1. They agree well with tabulated data. For a non-symmetric situation (30°) even the small non-zero average lift force is accurately computed.

For a semi-cylinder, three inclination angles relative to the span-wise direction, $\alpha = 30^{\circ}$, 90° , 120° , are considered, corresponding to straight separation zone (SSZ), combine separation zone (CSZ) and circular separation zone (CRSZ), respectively. Reynolds number is set to Re = 150 as in [46]. Fig. 14 shows the instantaneous vorticity and temperature contour. The irregular boundary is accurately captured and a vortex shedding process can be clearly observed. A hot blob can be seen in the right part of Fig. 14, detaching from the hot cylinder and gradually being advected downstream. In Table 2, current computed data is compared with corresponding results in literatures. An excellent agreement shows that present IB method is able to handle such geometries.

Table 1 Comparison of computed average data for flow over a square cylinder.

α		$\overline{C_D}$	$\overline{C_L}$	Str	Nu
0 °	Present	1.54	0.0	0.205	4.15
	Dulhani et al. [45]	1.56	0.0	0.213	4.10
30°	Present	1.71	0.069	0.189	5.37
	Dulhani et al. [45]	1.74	0.063	0.178	5.21
45°	Present	1.85	0.0	0.201	5.42
	Dulhani et al. [45]	1.83	0.0	0.211	5.37



Fig. 13. Instantaneous vorticity and temperature contour around a square cylinder, (a) 0°, (b) 30°, (c) 45°.



Fig. 14. Instantaneous vorticity and temperature contour around a semi-circle cylinder, (a) 30°, (b) 90°, (c) 120°.

Fable 2
Comparison of computed time average drag and lift coefficients, Nusselt number and Strouhal number for flow over a semi-circle cylinder.

α		$\overline{C_D}$	$\overline{C_L}$	Str	Nu
30°	Present	1.90	-0.025	0.211	7.81
	Bhinder et al. [47]	1.98	-0.077	0.213	7.95
	De [46]	1.938	-0.038	0.212	7.869
90°	Present	1.12	0.78	0.17	5.05
	Bhinder et al. [47]	1.13	0.85	0.178	4.92
	De [46]	1.127	0.837	0.18	4.985
120°	Present	1.42	-1.34	0.21	6.95
	Bhinder et al. [47]	1.47	-1.45	0.211	6.93
	De [46]	1.475	-1.471	0.216	6.955



Fig. 15. Mach contour around the NACA0012 airfoil (Mach = 0.5, Re = 5000), left: from [47], right: present result.

In the aforementioned simulations, a very small Mach number of 0.01 is set for the sake of code validation. In current section, attention is focused on the investigation of the effect of compressibility on heat transfer process. At first, the present method is further validated by a Mach = 0.5 flow over a NACA0012 airfoil. A stretched grid is used to maintain that even with a small number

^{3.5.} *High-speed flow over a bluff body with heat transfer*



Fig. 16. Pressure (right) and skin friction (left) coefficient distribution along the NACA0012 airfoil surface (Mach = 0.5, Re = 5000).



Fig. 17. Average Nusselt number variation with Mach number.

of grid points the grid resolution is around $\Delta x = \Delta y = 1/200$ (in the unit of chord length) in the vicinity of the boundary to resolve the thin trail. The comparison of current result with that based on a body-conformal method [48] is presented in is Figs. 15 and 16, where one can see a good agreement.

Then, a semi-circle cylinder is chosen because of its geometry similarities with an aerospace craft, with which high-speed thermal flow is prevailing. Results with two attack angles, namely, $\alpha = 90^{\circ}$ and $\alpha = 120^{\circ}$, are presented due to their representativeness. The same computational domain size and grid resolution as in Section 3.3 are employed. While a fixed temperature is prescribed on cylinder, NSCBC and periodic condition are applied at inlet, outlet and transverse direction, respectively. The Mach number in this study is 0.3, 0.4 and 0.5. For different Mach numbers, the Reynolds number remains constant and Prandtl number is kept to be 0.7.

The influence of Mach number on heat transfer parameters like averaged Nusselt number is shown in Fig. 17. The increase of Mach number results in a decrease of the average Nusselt number. To explore the underlying physical reasons for this phenomenon, the temperature contour is plotted in Fig. 18. As can be seen, when compressible fluid flows over a bluff body, it is compressed in the body's front side and thereby a high temperature region appears there. With the increase of Mach number, the high temperature region goes further upstream, which means a thicker thermal boundary layer. This thermal boundary layer highly prevents heat being transferred from the hot cylinder to surrounding cool fluid. Heat transfer characteristics are much more complicated at the rear. Like in $\alpha = 120^{\circ}$ case, other than that fluid is compressed so much in the strong recirculation zone giving rise to an even higher temperature than that of the cylinder, a much low temperature region and thus a very thin thermal boundary layer shows up (see right part in Fig. 18). The latter may be ascribed to a violent expansion of fluid after the vortex shedding.

Fig. 19 depicts the distribution of local Nusselt number over the body's surface. With boundary layer attaching to the front side, Nusselt number there decreases with the increase of Mach number. Under $\alpha = 120^{\circ}$ situation, corresponding to the very high temperature region and a very thin thermal boundary layer, a negative local Nusselt number and an increase of local Nusselt number with the increase of Mach number can be observed. However, this increment is not enough to compensate the decrease of average Nusselt number at the front side.

According to analysis above, we can conclude that present method has the ability to accurately capture flow characteristics around a complex geometry. Additionally, through the compressible cases, we find that Reynolds number and Prandtl number are not sufficient to define heat transfer features in a high-speed thermal flow problem, and as supplement Mach number and parameters relevant to geometry should be included. Besides those, when dealing with body immersing in high temperature compressible fluid, special attention should be paid to avoid a more serious thermal inhomogeneity and thus heat stress caused by compressibility.

4. Conclusions

In this paper, we extend our previous compressible immersed boundary method to the simulation of heat transfer between fluid and irregular geometries, with a stretched underlying mesh and also the boundaries being sharply represented. An isothermal flow past a fixed circle cylinder case points out a little influence of relative resolution as long as the spatial grid is fine enough. Two interpolation procedures are incorporated into current IB method to calculate the variable value on mirror points. A spatial convergence test indicates that only the bilinear procedure can obtain a local second-order accuracy. The present method is validated in a qualitative way by an accurate capture of the second vortex street behind an elliptical cylinder. An investigation into the forced convective flow over a semi-cylinder having sharp corners with one straight edge and one circle edge and a square cylinder with sharp



Mach=0.5

Fig. 18. Instantaneous temperature contour in different Mach numbers, left: 90° , right: 120° .



Fig. 19. Distribution of local Nusselt number corresponding to Fig. 18.

corners consisting of two straight edges provides a further quantitative verification. Good agreements with reported data show that the present method successfully captures main characteristics of the interaction between fluid flow and irregular boundaries. Finally, the proposed method is employed to study compressibility effect on heat transfer process in high-speed flow, which again demonstrates the essentiality of taking compressibility into consideration. The extension of present method to moving interface is straightforward, so is the implementation of Robin thermal boundary conditions, both of which have been previously presented in our work [1].

To save considerable amount of computation time for a static IB, a stretched mesh method, has been developed and tested in a 2D NACA0012 airfoil case. As for a more complex problem (e.g., 3D deforming elastic member interacting with surrounding flows), a better choice may be the AMR (adaptive mesh refinement) technique as in [49]. In addition, since a high order polynomial [12] can be straightly constructed into ghost-cell IB method for the implementation of all types of BCs, which is a must to resolve the turbulent boundary layer where the generic variable varies in a quite non-linear way even in a very small space, the ghost-cell IB method has the capability to resolve a thin viscous boundary layer. This technique further makes the high Reynolds number turbulent 3D simulation be feasible.

Since a fixed Cartesian mesh is utilized to resolve the fluid flow and there is no need to re-mesh even for a moving irregular boundary, present method has a potential for the simulation of heat transfer process in multiphase flows laden with non-spherical solid particles. Besides, thanks to the combination with a compressible solver, the current method has the ability to study the burning process of a solid particle with the surface fully resolved. These works are in process.

Acknowledgement

Financial support from the National Natural Science Foundation of China (Nos. 51390491, 51390493 and 51276164) and the National Program for Support of Top-notch Young Professionals is acknowledged. This work is also partially supported by the Fundamental Research Funds for the Central Universities of China.

References

- K. Luo et al., A ghost-cell immersed boundary method for simulations of heat transfer in compressible flows under different boundary conditions, Int. J. Heat Mass Transfer 92 (2016) 708–717.
- [2] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (2) (1972) 252–271.
- [3] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (1) (2005) 239-261.
- [4] C.S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479–517.
 [5] G. laccarino, R. Verzicco, Immersed boundary technique for turbulent flow
- simulations, Appl. Mech. Rev. 56 (3) (2003) 331–347.
 [6] S. Haeri, J.S. Shrimpton, On the application of immersed boundary, fictitious domain and body-conformal mesh methods to many particle multiphase
- flows, Int. J. Multiph. Flow 40 (2012) 38–55. [7] F. Sotiropoulos, X. Yang, Immersed boundary methods for simulating fluid-
- structure interaction, Prog. Aerosp. Sci. 65 (2014) 1–21.
 [8] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip flow boundary with an external force field, J. Comput. Phys. 105 (1993) 354–366.
- [9] E.A. Fadlun et al., Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (1) (2000) 35–60.
- [10] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2) (2005) 448–476.
- [11] K. Luo et al., Full-scale solutions to particle-laden flows: multidirect forcing and immersed boundary method, Phys. Rev. E: Stat., Nonlin, Soft Matter Phys. 76 (6 Pt 2) (2007) 066709.
- [12] J. Xia, K. Luo, J. Fan, A ghost-cell based high-order immersed boundary method for inter-phase heat transfer simulation, Int. J. Heat Mass Transfer 75 (2014) 302–312.

- [13] J. Xia, K. Luo, J. Fan, Simulating heat transfer from moving rigid bodies using high-order ghost-cell based immersed-boundary method, Int. J. Heat Mass Transfer 89 (2015) 856–865.
- [14] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids, J. Comput. Phys. 191 (2) (2003) 660–669.
- [15] R. Mittal et al., A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, J. Comput. Phys. 227 (10) (2008) 4825–4852.
- [16] K. Nagendra, D.K. Tafti, K. Viswanath, A new approach for conjugate heat transfer problems using immersed boundary method for curvilinear grid based solvers, J. Comput. Phys. 267 (2014) 225–246.
- [17] X. Zheng et al., A computational study of the effect of false vocal folds on glottal flow and vocal fold vibration during phonation, Ann. Biomed. Eng. 37 (3) (2009) 625–642.
- [18] H. Dong et al., Computational modelling and analysis of the hydrodynamics of a highly deformable fish pectoral fin, J. Fluid Mech. 645 (2010) 345.
- [19] C. Liu, C. Hu, An efficient immersed boundary treatment for complex moving object, J. Comput. Phys. 274 (2014) 654–680.
- [20] H.K.R.M. Uddin, A Cartesian-based embedded geometry technique with adaptive high-order finite differences for compressible flow around complex geometries, J. Comput. Phys. 262 (2014) 379–407.
- [21] H.C. Jungwoo Kim, An immersed-boundary finite-volume method for simulation of heat transfer in complex geometries, KSME Int. J. 18 (6) (2004) 1026–1035.
- [22] Z. Wang et al., Immersed boundary method for the simulation of flows with heat transfer, Int. J. Heat Mass Transfer 52 (19–20) (2009) 4510–4518.
- [23] J.R. Pacheco et al., Numerical simulations of heat transfer and fluid flow problems using an immersed-boundary finite-volume method on nonstaggered grids, Numer. Heat Transfer, Part B: Fundam. 48 (1) (2005) 1–24.
- [24] N. Zhang, Z.C. Zheng, S. Eckels, Study of heat-transfer on the surface of a circular cylinder in flow using an immersed-boundary method, Int. J. Heat Fluid Flow 29 (6) (2008) 1558–1566.
- [25] D. Pan, A simple and accurate ghost cell method for the computation of incompressible flows over immersed bodies with heat transfer, Numer. Heat Transfer, Part B: Fundam. 58 (1) (2010) 17–39.
- [26] W.W. Ren et al., Boundary condition-enforced immersed boundary method for thermal flow problems with Dirichlet temperature condition and its applications, Comput. Fluids 57 (2012) 40–51.
- [27] W. Ren, C. Shu, W. Yang, An efficient immersed boundary method for thermal flow problems with heat flux boundary conditions, Int. J. Heat Mass Transfer 64 (2013) 694–705.
- [28] A. Mark, E. Svenning, F. Edelvik, An immersed boundary method for simulation of flow with heat transfer, Int. J. Heat Mass Transfer 56 (1–2) (2013) 424–435.
- [29] I. Paul, K. Arul Prakash, S. Vengadesan, Forced convective heat transfer from unconfined isothermal and isoflux elliptic cylinders, Numer. Heat Transfer, Part A: Appl. 64 (8) (2013) 648–675.
- [30] https://github.com/pencil-code/.
- [31] A. Dadone, B. Grossman, Ghost-cell method for inviscid two-dimensional flows on Cartesian grids, AIAA J. 42 (12) (2004) 2499–2507.
- [32] A. Dadone, B. Grossman, Ghost-cell method with far-field coarsening and mesh adaptation for Cartesian grids, Comput. Fluids 35 (7) (2006) 676–687.
- [33] N.E.L. Haugen, S. Kragset, Particle impaction on a cylinder in a crossflow as function of Stokes and Reynolds numbers, J. Fluid Mech. 661 (2010) 239–261.
- [34] J.W. Nam, F.S. Lien, A ghost-cell immersed boundary method for large-eddy simulations of compressible turbulent flows, Int. J. Comput. Fluid Dyn. 28 (1– 2) (2014) 41–55.
- [35] A. Gilmanov, F. Sotiropoulos, A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies, J. Comput. Phys. 207 (2) (2005) 457–492.
- [36] Y. Tseng, J.H. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comput. Phys. 192 (2) (2003) 593–623.
- [37] S. Majumdar, G. Iaccarino, P. Durbin, RANS solvers with adaptive structured boundary non-conforming grids, Ann. Res. Briefs (2001) 353–366.
- [38] D. Pan, A general boundary condition treatment in immersed boundary methods for incompressible Navier–Stokes equations with heat transfer, Numer. Heat Transfer, Part B: Fundam. 61 (4) (2012) 279–297.
- [39] C. Merlin, P. Domingo, L. Vervisch, Immersed boundaries in large eddy simulation of compressible flows, Flow Turbul. Combust. 90 (1) (2013) 29–68.
- [40] A. Chaudhuri, A. Hadjadj, A. Chinnayya, On the use of immersed boundary methods for shock/obstacle interactions, J. Comput. Phys. 230 (5) (2011) 1731-1748.
- [41] J.S.F. Yang, Robust and efficient setup procedure for complex triangulations in immersed boundary simulations, J. Fluids Eng., Trans. ASME (2013).
- [42] C.S. Yoo, H.G. Im, Characteristic boundary conditions for simulations of compressible reacting flows with multi-dimensional, viscous and reaction effects, Combust. Theor. Model. 11 (2) (2007) 259–286.
- [43] S.C.R. Dennis, G. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds numbers up to 100, J. Fluid Mech. 42 (03) (1970) 471– 489.
- [44] M.C. Thompson et al., Low-Reynolds-number wakes of elliptical cylinders: from the circular cylinder to the normal flat plate, J. Fluid Mech. 751 (2014) 570–600.

- [45] J.P. Dulhani, S. Sarkar, A. Dalal, Effect of angle of incidence on mixed convective wake dynamics and heat transfer past a square cylinder in cross flow at Re = 100, Int. J. Heat Mass Transfer 74 (2014) 319–332.
- [46] A.K. De, A diffuse interface immersed boundary method for convective heat and fluid flow, Int. J. Heat Mass Transfer 92 (2016) 957–969.
- [47] A. Pal Singh Bhinder, S. Sarkar, A. Dalal, Flow over and forced convection heat transfer around a semi-circular cylinder at incidence, Int. J. Heat Mass Transfer 55 (19–20) (2012) 5171–5184.
- [48] Y. Sun, Z.J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow, J. Comput. Phys. 215 (1) (2006) 41–58.
- [49] A.P.S. Bhalla et al., A unified mathematical framework and an adaptive numerical method for fluid–structure interaction with rigid, deforming, and elastic bodies, J. Comput. Phys. 250 (2013) 446–476.