An Improved Hybrid Cartesian/Immersed Boundary Method for Flow Simulation with Moving Boundaries

Guanting Su¹, Mengzong Zheng¹, * and Qiushi Li¹, ²

¹School of Energy and Power Engineering, Beihang University, Beijing, China
²Key Laboratory of Fluid and Power Machinery, Xihua University, Chengdu, China

*Corresponding author: zhengmengzong@buaa.edu.cn

Abstract. To simulate flow problems with moving fluid-solid interfaces, a boundary treatment technique based on a hybrid Cartesian/immersed boundary method is presented. In the vicinity of the immersed boundary, a forcing term is added into the governing flow equations to modify the discretization scheme for the viscous term during its spatial discretization. Addition of such a term enforces desired boundary condition on the immersed boundary. A one-dimensional linear interpolation scheme is used to obtained required information for the modified discretization scheme when dealing with a moving boundary. The present method is essentially second-order accurate and can obtain results that are in good agreement with benchmark references when simulating the flow induced by the in-line oscillation of a circular cylinder.

Keywords: immersed boundary method, moving boundaries, Cartesian grid, incompressible flow, finite difference method.

1. Introduction

Despite the continuous growth of computational power, numerical simulation of flow past a flexible or rigid moving solid object with complex shape still is a cumbersome task for traditional computational fluid dynamic approaches that employ body conformal grids. This led to the emergence of several alternatives, among which immersed boundary (IB) method stands out as a promising approach after introduced by Peskin [1]. By adding appropriate forcing terms to the governing equations, the IB methods can be implemented on a fixed nonbody-conformal (e.g. Cartesian) grid and reproduce the influence of an IB on the flow while avoiding complications as in the case of using a body conformal grid.

Classic IB methods derive the expression of the forcing term through appropriate constitutive laws such as Hook’s law based on the characteristic of the IBs and have been successfully used in a wide variety of problems [2]. However, major drawbacks exist when dealing with rigid bodies using these IB methods, like the inability to produce a sharp representation of the IB due to the employment of a smooth distribution function for the body-force terms and the inconvenience faced [3].

First introduced under the context of Cartesian grid method for inviscid flow simulations by Clarke et al. [4] and later applied to the simulation of viscous flows [5, 6], the cut-cell methods reshape the Cartesian grid cells cut by the IB according to the local geometry of the boundary and produces a locally body conforming unstructured grid. Implementing a finite volume approach, the fluxes across the cut-
cell faces are reconstructed from surrounding regular cells and adjacent boundaries. This ensures the inherent conservation of flow properties and the higher accuracy near the IBs, which are desiring and distinguish the cut-cell method from other IB methods. However, treating cut-cells with various topologies could be cumbersome even in two-dimensional problems, which make the application of cut-cell methods to three-dimensional problems with moving complex boundaries unstraightforward and nontrivial.

Although the implementation of the cut-cell method on three-dimensional flow problems with moving complex boundaries may sometimes lead to complication and inconvenience, its combination with the classic IB method give rise to a simpler and more straightforward sharp interface approach called hybrid Cartesian immersed boundary (HCIB) method [7]. The key idea is to view the body-force term as an implicit term whose addition results in the direct satisfaction of the velocity boundary conditions in the vicinity of the IB rather than a term that requires explicit calculation. Through this direct forcing approach, instead of modifying the shape of local grid cell as in the cut-cell methods, local velocity distribution is modified according to the IB using one-dimensional linear interpolation along an arbitrary coordinate direction [8], bilinear interpolation [9], quadratic interpolation along local boundary normal direction [10] and an interpolation scheme based second-order Taylor series expansion [11]. Moreover, HCIB method are expanded to the interactions of flexible structure with surrounding fluid [12] and the simulations of inviscid compressible flow in subsonic [13] and supersonic [14] regimes.

The essence of the HCIB methods lies in the direct imposition of the velocity boundary conditions on the IB. Rather than resorting to a certain interpolation scheme to determine the velocities on the grid points immediately adjacent to the IB, Leonardi and Orlandi [15] altered the discretization scheme of the governing equations on such points to impose the velocity boundary conditions in an equally direct fashion and successfully simulated flow problems with stationary complex IBs, such as turbulent flow in rough channels [16, 17]. In this paper, we expand this method to the simulation of flow problems with moving complex IBs and necessary amendments are elaborated in the following sections. In Section 2, the governing equations, the Navier-Stokes solver and other numerical details are presented. Results are presented in Section 3 and finally some concluding remarks are given in Section 4.

2. Numerical methods

2.1. Governing equations and basic solver

In the present paper, unsteady incompressible Navier-Stokes equations formulated in primitive variable form are solved. Taking the influence of immersed solid objects into consideration, the IB method introduces a forcing term $f$, which is non-zero except on the IBs, into the momentum equation and transforms the equations that govern the encompassing flow as follows:

$$\frac{\partial u}{\partial t} + \nabla \cdot (uu) = -\nabla p + \frac{1}{Re}\nabla^2 u + f,$$

$$\nabla \cdot u = 0$$

(1a)

(1b)

Note that the governing equations are nondimensionalized using a reference velocity $U_0$, a reference length $L_0$ and a reference time scale $L_0/U_0$, which will all be specified afterwards for each test case; pressure $p$ is nondimensionalized by $\rho U_0^2$, where $\rho$ is the density of the fluid. $Re = U_0L_0/\nu$ is the Reynolds number, where $\nu$ is the kinematic viscosity of the fluid.

The governing equations are spatially discretized using second-order central differences on a staggered grid and advanced in time employing a fractional-step method [18, 19]. Second-order-explicit Adams-Bashforth scheme is used for the convective terms and second-order-implicit Crank-Nicolson scheme is used for the viscous terms. The treatment of the forcing term $f$ will be elaborated in the next section.

Inside the flow field, fast implicit solution of the discretized form of Eq. (1a) requires the usage of alternating direction implicit (ADI) method [20], which simultaneously ensures consistency and
unconditional stability, especially for three-dimensional problems. Eq. (1a) is first transformed into the following delta form:

\[
\left(1 - \frac{\Delta t}{2Re} \nabla^2 \right) \delta u = \Delta t \cdot \text{RHS}^{n+1/2},
\]

where \( \delta u = \hat{u} - u^n \) and \( \text{RHS}^{n+1/2} = -\nabla p^n - \left[ 3 \cdot \nabla (u u^n) / 2 - \nabla (u u^n) / 2 \right] + \nabla^2 u^n / Re \). Then consider a three-dimensional problem under Cartesian coordinate system as an example, Eq. (2) for this problem can be consistently split as follows:

\[
\left\{ \begin{array}{l}
1 - \frac{\Delta t}{2Re} \frac{\partial^2}{\partial x^2} \delta u' = \Delta t \cdot \text{RHS}^{n+1/2} \\
1 - \frac{\Delta t}{2Re} \frac{\partial^2}{\partial y^2} \delta u'' = \Delta t \cdot \text{RHS}^{n+1/2} \\
1 - \frac{\Delta t}{2Re} \frac{\partial^2}{\partial z^2} \delta u = \Delta t \cdot \text{RHS}^{n+1/2}
\end{array} \right.
\]

where \( \delta u' = u' - u^n \), \( \delta u'' = u'' - u^n \), and \( u' \) and \( u'' \) are intermediate velocities. Obviously, \( \text{RHS}^{n+1/2} \) can be evaluated using quantities known from previous steps. Therefore, Eq. (3) can be spatially discretized into a tridiagonal system of linear algebraic equations which can be efficiently solved.

2.2. Discretization of the forcing term

Under the context of the hybrid Cartesian immersed boundary method, the forcing term \( f \) is viewed as an implicit term whose addition results in the direct satisfaction of the velocity boundary conditions on the IB. When the momentum equation is advanced in time in the aforementioned fashion, the temporally discretized equation on the IBs is as follows:

\[
\hat{u} - u^n = -\text{RHS}^{n+1/2} + f^{n+1/2}.
\]

To satisfy the velocity boundary condition on the IB, yielding \( \hat{u} = U_{\text{IB}}^{n+1} \), where \( U_{\text{IB}}^{n+1} \) is the known local velocity of the IB at time layer \( n+1 \), the corresponding explicit expression for \( f \) on the IBs is simply as follows:

\[
f^{n+1/2} = -\text{RHS}^{n+1/2} + U_{\text{IB}}^{n+1} - u^n / \Delta t.
\]

Since the locations of the grid points where the flow variables are stored do not always coincide with the IBs, the explicit expression of the body-force term \( f \) after spatial discretization requires corrections in the vicinity of the IBs, where \( f \) should equal to zero formerly since it is inside the flow region. The points on which the correction of \( f \) takes place are called IB points [10, 12, 13]. The classification of the IB points usually is the initial step of a HCIB method, the process or the criterion of which may vary slightly according to the different grid formation or spatial discretization scheme employed. The general gist is to label the points that are in the immediate vicinity of the IBs as the IB points [21].

In [7-14], after the introduction of the body-force term, velocities on the IB points are determined according to a prescribed local velocity profile. This is where the method employed by the present paper parts its way with the above HCIB methods, putting the body-force to effect by discretizing the governing equation distinctively on the IB points. In this sense, \( f \) can be explicitly expressed as follows:

\[
f = -D_n \left( \text{RHS} \right) + D_{n,\text{IB}} \left( \text{RHS} \right).
\]

where \( D_n \) and \( D_{n,\text{IB}} \) indicate the normal second-order central differential scheme and a distinctive spatial discretization scheme on the IB points respectively. Consequently, when it comes to the correction for the forcing term \( f \) on the IB points, the specific form of \( D_{n,\text{IB}} \) is discussed. In the vicinity
of the IBs, where the IB points are located, the presence of the IBs poses influence on the flow mainly through viscous effect while convective effect and pressure gradients plays a relative minor part [8]. Therefore, $D_{s,\text{IB}}$ only contains a distinctive discretization of the viscous term, which is illustrated in Fig. 1, while other terms are discretized normally.

**Figure 1.** Discretization stencil for $D_{s,\text{IB}}$ at an IB point A. The shaded area represents the solid domain. An open triangle represents an intersection point on the IB with the grid line and a filled circle represents a grid point.

During the spatial discretization at Point A in Fig. 1, to enforce the boundary condition at the intersection point B, the discretization scheme of the viscous term employs the following second order differential form:

$$
\frac{\partial^2 u}{\partial x^2} = \frac{2u_B}{\Delta b \cdot (\Delta b + \Delta e)} - \frac{2u_A}{\Delta b \cdot \Delta e} + \frac{2u_E}{\Delta e \cdot (\Delta b + \Delta e)}.
$$

(7)

When the delta form ADI method is employed as described in Section 2.1, the viscous term in delta form is also similarly discretized as follows:

$$
\frac{\partial^2 \delta u}{\partial x^2} = \frac{2\delta u_B}{\Delta b \cdot (\Delta b + \Delta e)} - \frac{2\delta u_A}{\Delta b \cdot \Delta e} + \frac{2\delta u_E}{\Delta e \cdot (\Delta b + \Delta e)}.
$$

(8)

### 2.3. Treatment of moving IBs

For a moving boundary, location of IB points and related boundary points may vary as time advances. Therefore, the required information of $u^n_B$ may not be available at time layer $n+1$, impeding the direct employment of $D_{s,\text{IB}}$ described by Eq. (2). For convenience, we refer to the boundary point that relates to Point A in Fig. 1 at time layer $n$ as Point $B^n$ and its counterpart at time layer $n+1$ as Point $B^{n+1}$.

A straightforward way to extract information at an arbitrary location from a discretized field is resorting to an interpolation process. It should be noted that although multi-dimensional problems are dealt here, bilinear or trilinear interpolation is not required since Point $B^{n+1}$, $B^n$ and A always locate on the same straight line. As a result, simple linear interpolation will suffice. Four possible spatial arrangements of the moving IBs and gridlines exist and the evaluation procedure differs from each other under different arrangements, which are enumerated hereinbelow and illustrated in Fig. 2:

Situation 1 as in Fig. 2(a): from time layer $n$ to $n+1$, the local IB moves roughly towards its local exterior normal direction and the movement is not rapid enough for the IB to cross a grid line in a single time-step. As a result, located between Points $B^n$ and $A$, Point $B^{n+1}$ lies in the flow region at time layer $n$. Therefore $u^{n+1}_{B^{n+1}}$ can be obtained as $u^{n+1}_{B^{n+1}} = D\left(u^n_{B^n}, u^n_A\right)$, where $D$ indicates a one-dimensional linear interpolation operator.
Situation 2 as in Fig. 2(b): from time layer \(n\) to \(n+1\), the local IB moves roughly towards its local exterior normal direction. Different from Situation 1, the movement is rapid enough that the IB crosses a grid line in a single time-step, shifting the corresponding IB node from Point I to Point A. In this case, Point \(B^{n+1}\) locates between Points I and A and lies within the fluid domain at time layer \(n\). Therefore, \(u_{B^{n+1}}^n\) can be obtained as

\[
\left(\mathbf{u}_I^n, \mathbf{u}_A^n\right) = D(\mathbf{u}_I^n, \mathbf{u}_A^n).
\]

Figure 2. Grid-IB layout for four different types of boundary movement relative to an IB point.

Situation 3 as in Fig. 2(c) or 2(d): from time layer \(n\) to \(n+1\), the local IB moves roughly towards its local interior normal direction. Whether the IB crossed a grid line in a single time-step as in Fig. 2(c) or not as in Fig. 2(d), Point \(B^{n+1}\) lies in the solid region at time layer \(n\). Therefore, similar to the approach utilized by [8, 15], \(u_{B^{n+1}}^n\) is imposed as equal to the local velocity of the solid body itself, i.e.

\[
u_{B^{n+1}}^n = U_{B^{n+1}}^n.
\]

It should be noted that apart from the three situations enumerated above, there are other situations where the IB coincides with a grid point at time layer \(n\) or \(n+1\). For the latter case, since Point \(B^{n+1}\) coincides with a grid node, \(u_{B^{n+1}}^n\) can be directly obtained from the descended information of the time layer \(n\). While in the former case, the classification of the situation and its processing is not affected with Point \(B^{n+1}\) coinciding with a grid node.

3. Results and discussions

3.1. Accuracy study: flow induced by a circular cylinder oscillating in a cavity

The numerical scheme introduced in Section 2 is theoretically second-order accurate. To demonstrate the formal accuracy of the method, a grid refinement study is carried out by simulating the flow due to a sinusoidally oscillating rigid circular cylinder inside a cavity similar to common practice [6, 10]. This two-dimensional test case is nondimensionalized using the diameter of the circular cylinder and the maximum translational velocity of the oscillation, on which the Reynolds number is based and set equal to 100. The oscillation takes place inside a square cavity ranged \(-1 \leq x \leq 1\) and \(-1 \leq y \leq 1\) and the location of the circular cylinder center is prescribed as \(x(t) = 0, y(t) = 0.125\sin(8\pi t)\). Non-slip wall boundary conditions are applied to all four boundaries of the computational domain.

To perform the error analysis, the following sequence of uniformly spaced mesh is used: \(20 \times 20\), \(40 \times 40\), \(80 \times 80\) and \(160 \times 160\). Due to absence of the exact analytical solution to this flow problem, the solution on the finest mesh is considered to be the exact solution for the following analysis. For all four simulations, the same sufficiently-small time step of \(\Delta t = 7 \times 10^{-4}\) is used to minimize the effect of temporal...
error, where $T$ is the nondimensionalized oscillation period and equals to $\pi/4$. The simulations were initiated at $t=0$ and last for 1.25 oscillation cycle. At the end of the simulation, when the translational velocity of the cylinder reaches its maximum, the $L_1$ and $L_2$ norms of the error are calculated as follows:

$$\varepsilon_{N}^{q} = \left[ \frac{1}{N^2} \sum_{i=1}^{N^2} \left| u_i^{(N)} - u_i^{e} \right|^q \right]^{1/q},$$  

(9)

where $\varepsilon_{N}^{q}$ indicates the $q$th norm of error of the solution on the $N \times N$ grid, $u_i^{(N)}$ is the $x$-component of velocity on the $i$th node obtained from the simulation with the $N \times N$ grid and $u_i^{e}$ is the corresponding exact value on the finest grid. The result is summarized by a log-log scaled plot as in Fig. 3. It is clearly shown that as the grid spacing size decreases, the two norms of error both decrease at a rate which is close to second order. A small deviation from the second order slope can be observed but this should be caused by the lack of a true exact solution to this flow problem. Therefore, we can draw the conclusion that the present method is essentially second-order accurate when dealing with moving boundaries.

![Figure 3. Variation of the $L_1$ and $L_2$ norms of the error for $u$ with spatial step size. The slope of the $L_1$ and $L_2$ lines are both around 1.7.](image)

3.2. Flow induced by the in-line oscillation of a circular cylinder in quiescent fluid

For the flow problem involving a circular cylinder oscillating in an initially static fluid, where the translational motion of the cylinder can be described as $x(t)=-A \sin(2\pi ft)$, $y(t)=0$, the governing dimensionless parameters are the Reynolds number, $Re=U_{\text{max}}D/\nu$, and the Keulegan-Carpenter number, $KC=U_{\text{max}}/fD=2\pi A/D$, where $A$ and $f$ is the oscillation amplitude and frequency of the cylinder, respectively, $U_{\text{max}}$ is the maximum translational velocity of the cylinder, and $D$ is the diameter of the cylinder. In the present study, the Reynolds and Keulegan-Carpenter number were set to $Re=100$ and $KC=5$, under which the flow remains two-dimensional with a periodic vortex shedding pattern. The computational domain size is set to $20D \times 20D$. And local grid spacing in the vicinity of the cylinder is set to be $0.0125D$ and the time step is set to be $T/10^4$. The simulation results could be regarded as insensitive to all these configurations with the chosen value. And periodic vortex shedding pattern is established after two or three oscillation cycles.

After comparing the total aerodynamic force coefficient, the pressure and vorticity contours at four different phase angles are presented in Fig. 4. These contours are obtained in the fifth oscillation cycle when the periodicity is already reached. Observed from Fig. 4(b), as the cylinder moves from right to left, the flow separates at the same upper and lower position on the leading edge of the cylinder, which forms two counter-rotating vortices with same vorticity. This vortex formation ceases to continue as the cylinder reaches its extreme left location and the two generated vortices detach from the cylinder surface when the cylinder starts to accelerate backwards as shown in Fig. 4(d). These results accord with the corresponding experimental results reported in [22] as well as numerical results [9], indicating that the proposed numerical method can produce reliable results concerning dynamic vortex structure.
Figure 4. Pressure (a and c) and vorticity (b and d) contour of in-line oscillating circular cylinder in static fluid \((Re=100 \text{ and } KC=5)\) at different oscillation phase: \(0^\circ\) for (a, b); \(96^\circ\) for (c, d). The interval of the pressure contours is 0.1, with the pressure ranging from -2 to 2. The interval of the vorticity contours is 0.5, with the vorticity ranging from -20 to 20. The negative vorticity isolines are dashed.

Figure 5. Velocity profile comparison of four \(x\)-direction cross-sections of in-line oscillating circular cylinder in static fluid \((Re=100 \text{ and } KC=5)\) at different oscillation phase: (a) \(180^\circ\); (b) \(330^\circ\). Lines indicate present numerical results. Closed symbols indicate reference results from [9] in (a) and reference results from [22] in (b), respectively.
To further demonstrate the accuracy of this method, a detailed quantitative comparison of velocity profile at four $x$ locations with the reference results [9, 22] is presented in Fig. 5, exhibiting good agreement with experimental and numerical benchmarks.

4. Conclusions
To expand the applicability of an established hybrid Cartesian/immersed boundary method, a boundary treatment technique is introduced in the present paper to simulate flow problems with moving boundaries using static, non-body conformal, Cartesian grids. In the vicinity of the moving boundary, a forcing term that is added into the governing flow equations modifies the discretization scheme for the viscous term during its spatial discretization. Such modifications ensure the enforcement of desired boundary conditions on the immersed boundary. And a one-dimensional linear interpolation scheme is used to complete discretized information that is otherwise unavailable due to the movement of the immersed boundary for the modified discretization scheme.

And accuracy study simulating the flow induced by a circular cylinder oscillating in a cavity shows that the present method is essentially second-order accurate. And the present method is validated using the test problem of the flow induced by the in-line oscillation of a circular cylinder, simulation result of which are in good agreement with experimental and numerical benchmarks.

Acknowledgments
The authors acknowledge the support of Beijing Natural Science Foundation (No. 3214047) and National Natural Science Foundation of China (No. 52006002).

References
[1] C.S. Peskin, Flow patterns around heart valves: a numerical method, Journal of computational physics, 10 (1972) 252-271.
[2] C.S. Peskin, The fluid dynamics of heart valves: experimental, theoretical, and computational methods, Annual review of fluid mechanics, 14 (1982) 235-259.
[3] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech., 14 (2005) 239-261.
[4] D.K. Clarke, H.A. Hassan, M.D. Salas, Euler calculations for multielement airfoils using Cartesian grids, AIAA Journal, 24 (1986) 353-358.
[5] T. Ye, R. Mittal, H.S. Udaykumar, W. Shyy, An Accurate Cartesian Grid Method for Viscous Incompressible Flows with Complex Immersed Boundaries, Journal of Computational Physics, 156 (1999) 209-240.
[6] H.S. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface cartesian grid method for simulating flows with complex moving boundaries, Journal of Computational Physics, 174 (2001) 345-380.
[7] J. Mohd-Yusof, Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries, CTR Annual Research Briefs, (1997) 317-327.
[8] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, Journal of Computational Physics, 161 (2000) 35-60.
[9] J. Yang, E. Balaras, An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, Journal of Computational Physics, 215 (2006) 12-40.
[10] A. Gilmanov, F. Sotiropoulos, A hybrid Cartesian/immersed boundary method for simulating flows with 3D, geometrically complex, moving bodies, Journal of Computational Physics, 207 (2005) 457-492.
[11] T. Gao, Y.H. Tseng, X.Y. Lu, An improved hybrid Cartesian/immersed boundary method for fluid–solid flows, International Journal for Numerical Methods in Fluids, 55 (2007) 1189-1211.
[12] S. Shin, S.Y. Bae, I.C. Kim, Y.J. Kim, J.S. Goo, Computations of flow over a flexible plate using the hybrid Cartesian/immersed boundary method, International Journal for Numerical
[13] Y. Zhang, C. Zhou, An immersed boundary method for simulation of inviscid compressible flows, International Journal for Numerical Methods in Fluids, 74 (2014) 775-793.

[14] S. Brahmachary, G. Natarajan, V. Kulkarni, N. Sahoo, A sharp interface immersed boundary framework for simulations of high speed inviscid compressible flows, International Journal for Numerical Methods in Fluids.

[15] S. Leonardi, P. Orlandi, A numerical method for turbulent flows over complex geometries, ERCOFTAC bulletin, 62 (2004) 41-46.

[16] P. Orlandi, S. Leonardi, DNS of turbulent channel flows with two- and three-dimensional roughness, Journal of Turbulence, 7 (2006) N73.

[17] P. Orlandi, S. Leonardi, Direct numerical simulation of three-dimensional turbulent rough channels: parameterization and flow physics, Journal of Fluid Mechanics, 606 (2008) 399-415.

[18] J. Kim, P. Moin, Application of a fractional-step method to incompressible Navier-Stokes equations, Journal of Computational Physics, 59 (1985) 308-323.

[19] R. Verzicco, P. Orlandi, A Finite-Difference Scheme for Three-Dimensional Incompressible Flows in Cylindrical Coordinates, Journal of Computational Physics, 123 (1996) 402-414.

[20] R.M. Beam, R.F. Warming, An Implicit Factored Scheme for the Compressible Navier-Stokes Equations, AIAA Journal, 16 (1978) 393-402.

[21] F. Sotiropoulos, X. Yang, Immersed boundary methods for simulating fluid–structure interaction, Progress in Aerospace Sciences, 65 (2014) 1-21.

[22] H. DÜTscher, F. Durst, S. Becker, H. Lienhart, Low-Reynolds-number flow around an oscillating circular cylinder at low Keulegan–Carpenter numbers, Journal of Fluid Mechanics, 360 (1998) 249-271.