

City Research Online

City St George's, University of London

Citation: Wang, Q., Liao, K., Ma, Q. & Khayyer, A. (2025). A Novel Overset Grid Assembly Strategy based on Immersed Boundary Method for Fluid-Structure Interaction. Journal of Computational Physics, 541, 114321. doi: 10.1016/j.jcp.2025.114321

This is the accepted version of the paper.

This version of the publication may differ from the final published version. To cite this item please consult the publisher's version.

Permanent repository link: https://openaccess.city.ac.uk/id/eprint/35778/

Link to published version: https://doi.org/10.1016/j.jcp.2025.114321

Copyright and Reuse: Copyright and Moral Rights remain with the author(s) and/or copyright holders. Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge, unless otherwise indicated, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way. For full details of reuse please refer to City Research Online policy.

City Research Online: http://openaccess.city.ac.uk/ publications@citystgeorges.ac.uk

A Novel Overset Grid Assembly Strategy based on Immersed Boundary Method for Fluid-Structure Interaction

Qiang Wang [a], Kangping Liao [a,*], Qingwei Ma [b], Abbas Khayyer [c]

[a] *College of Shipbuilding Engineering, Harbin Engineering University, Harbin, China*

[b] *School of Mathematics, Computer Science and Engineering, City, University of London, London, UK*

[c] *Department of Civil and Earth Resources Engineering, Kyoto University, Kyoto, Japan*

Abstract

The overset grid method and the immersed boundary (IB) method, which have significant advantages in dealing with moving boundaries involving large amplitude motion, are widely used for fluid-structure interaction (FSI) problems in engineering applications. However, these methods still have some limitations. In the overset grid method, the hole-cutting operator and iterative procedure, which are complex and time-consuming, are necessary for flow field information exchange between the sub-domains and the background domain. In the IB method, non-physical pressure oscillations, which can result in numerical instability, often exist in simulating FSI problems with moving boundaries. In order to integrate the advantages of these two methods and overcome their limitations, a novel overset grid assembly strategy based on the IB method is proposed. In the present method, two additional body force terms, obtained by the diffuse-interface IB method, are considered in the momentum equations for solving the flow field in the background domain. The first body force term implicitly imposes a no-slip boundary condition of the body surface in the background domain. The second body force term is used to further correct the background flow field, and to ensure the continuity of the flow field between the background domain and the sub-domains. In this way, the explicit flow field information interpolation process from the sub-domains to the background domain is not necessary. Therefore, the complicated hole-cutting operator can be avoided. In addition, a third-order polynomial time extrapolation technique is adopted to estimate the pressure at the sub-domain boundaries. The velocity at the sub-domain boundaries can be determined with the estimated pressure. Therefore, the velocity and pressure boundary conditions at the sub-domain boundaries can be ensured without an iterative process between the sub-domains and the background domain. It significantly enhances the computational efficiency since the iterative process is avoided. Several benchmark cases are carried out to verify and validate the novel overset grid assembly strategy. In all cases, the results are in good agreement with the published experimental or numerical results. Furthermore, the results demonstrate that the present method can effectively suppress non-physical pressure oscillations, which often exist in the IB method when considering moving boundaries.

Keywords: Immersed boundary method; overset grid method; body force; pressure interpolation

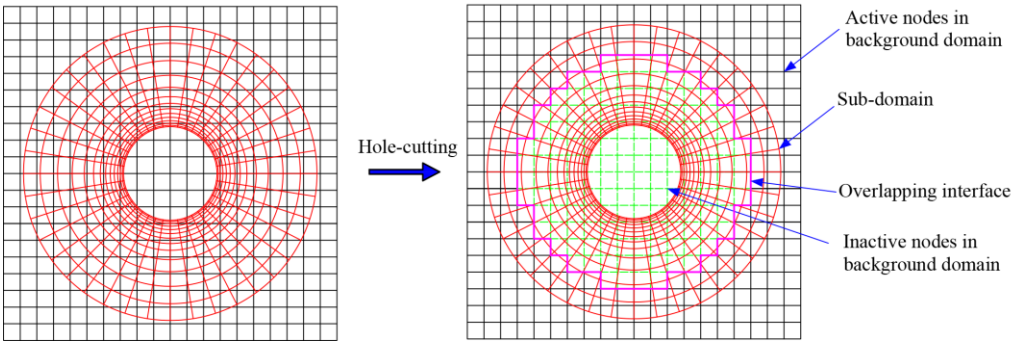
1. Introduction

FSI problems with moving boundaries and complex geometries are frequently encountered in engineering applications, such as marine propellers [1], floating offshore wind turbines [2], or aquatic swimmers [3]. To handle FSI problems involving moving boundaries, several numerical methods have been developed, including grid deformation methods [4], IB methods [5], and overset

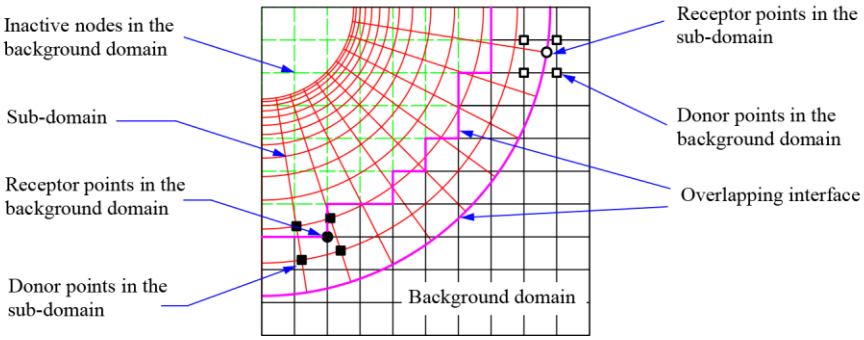
grid methods [6], etc. In grid deformation methods, the grid topology around a moving body is modified at each time step. Therefore, large amplitude motions are usually not allowed. In contrast, IB methods and overset grid methods are suitable for addressing FSI problems involving large amplitude motions, since they do not need to modify the grid topology.

The overset grid method was firstly proposed by Steger et al. [6], and has been implemented in many commercial software (Star-CCM+ [7], ANSYS Fluent [8]), open-source software (OpenFOAM [9]) and in-house codes [10]. In the overset grid method, the entire computational domain consists of a background domain and sub-domains. In general, the background domain is fixed, and the sub-domains can move with bodies. One advantage of the overset grid method is that the background domain can be discretized with Cartesian grids. More complex body-fitted or unstructured grids can be applied in the sub-domains to resolve small flow scales. Another advantage is that the overset grid method does not require modifying the grid topology at each time step. Therefore, the overset grid method is particularly suitable for addressing dynamic boundary problems involving large amplitude motions and high Reynolds number.

In overset grids, the grid topology of the sub-domains and the background domain is typically different and non-conformal. Therefore, special treatments are required for exchanging flow field information between the background domain and the sub-domains. In order to achieve the exchange of flow field information, the overset grid assembly process is necessary, as shown in Fig. 1.



(a) Hole-cutting step



(b) Interpolation searching step

Fig. 1. Schematic demonstration of the traditional overset grid assembly

The overset grid assembly process includes two steps: hole-cutting step and interpolation searching step [11]. The hole-cutting step excludes grid nodes located outside of computational domains and defines the overlapping interface of the background domain. After hole-cutting, the background domain grid nodes are divided into active and inactive nodes, as shown in Fig. 1(a).

The active nodes will be used for the solution of the flow field, and the inactive nodes will be removed from the computation [12]. In the overset grid assembly process, the main two challenges are the degree of automation and computational efficiency of the hole-cutting process [11]. To overcome these challenges, various hole-cutting methods have been proposed. These methods can be generally classified as explicit hole-cutting methods and implicit hole-cutting methods [13]. The explicit hole-cutting methods include the hole-map method [14], the vector method [15] and the x-ray method [16], etc. They use specified curves or surfaces to remove the grid nodes inside the bodies. However, the limitation of explicit hole-cutting methods is that they require a certain level of user input and expertise, which significantly affects the performance of hole-cutting [17].The implicit hole-cutting methods can automatically complete the hole-cutting process based on certain mesh criteria, such as distance-to-the-wall [18], cell volume [19], cell aspect ratio and cell angle [20], etc. The implicit hole-cutting methods have a high degree of automation, but they require very large computational costs and memory [11]. To enhance the computational efficiency of the implicit hole-cutting method, several approaches have been developed, including those proposed by Tian et al.[12], Fu et al. [21] and Oh et al. [22], etc.

The interpolation searching step consists of two parts: the first part is identifying all receptor points, and the second part is finding donor points for all receptor points, as shown in Fig. 1(b). After the interpolation searching step is completed, flow field information is transferred from the donor points to the receptor points. In general, this transmission is explicit, meaning that the flow field information at the receptor points is directly obtained through interpolation from the donor points. To improve the numerical stability of the overset grid method, the simultaneous approximation term (SAT) penalty method was proposed, which is an implicit method for transferring the flow field information [23,24]. In the SAT penalty method, penalty terms are added to the receptor points of both the background domain and the sub-domains. Additionally, depending on how the flow field is solved in the background domain and the sub-domains, the overset grid methods can be further divided into weak coupling methods [25] and strong coupling methods [26]. In weak coupling methods, the flow fields in different computational domains are solved separately using identical or different methods. It is easy to code, but it usually requires several iterations between the background domain and the sub-domains to obtain accurate velocity and pressure at the receptor points. Therefore, the computational cost increase rapidly with the number of grids. In strong coupling methods, the flow fields in the background domain and the sub-domains are solved simultaneously in a unified framework. In each time step, the iterative process is not required, and accurate velocity and pressure at receptor points can be obtained. However, the construction of the global coefficient matrix is quite complex.

The IB method was initially proposed by Peskin [5] and applied to the simulation of heart valves. In general, there are two types of IB methods [27]: the diffuse-interface IB methods [5,28] and the sharp-interface IB methods [29,30]. A detailed introduction can be found in the work of Verzicco [31]. In the IB methods, the governing equations can be solved on a fixed Cartesian grid, and the body force term is added to the momentum equation to consider the effect of the body boundary on the fluid flow. It has several advantages: (i) It is easy to generate grids, since the grids do not need to conform to the body [32]; (ii) It is easy to deal with a moving body, since the body can move freely in a fixed Cartesian grid [33]. Therefore, the IB methods can be used to deal with large amplitude/deformation motion. However, the IB methods have some limitations. One of them is the non-physical pressure oscillations in simulating moving boundaries. In order to suppress non-

physical pressure oscillations, various approaches have been developed, such as those by Yang and Balaras [34], Yang et al. [35], Liu et al. [36], Chiu [37], Yu and Pantano [38], and Xu et al. [39], etc. Another limitation is that mesh refinement in all directions is necessary to accurately resolve near-wall boundary layers, resulting in a large number of grids. As the Reynolds number increases, this problem worsens due to the thinning of the boundary layer. Therefore, the IB methods become impractical for high-Reynolds number flows. To address this problem, some work has been devoted to combining the IB methods with wall-layer models [40], local mesh refinement [41], or adaptive mesh refinement [42].

In order to integrate the advantages of the overset grid methods and the IB methods while mitigating their drawbacks, a coupling method based on these methods has been proposed. Specific improvements in the coupling method include: (i) reducing the total number of grid cells and enhancing the computational efficiency of the IB method [43]; (ii) increasing the grid resolution near body boundaries and improving the accuracy of the IB methods [33,44,45,46]; (iii) simplifying grid generation in the overset grid method [32]. However, to the best knowledge of the authors, except for the half-domain body-fitted mesh immersed boundary method (HBM-IBM) proposed by Wang at al. [46], the reported weak coupling overset grid methods still require hole-cutting operations, and the iterative process between the background domain and the sub-domains cannot be avoided.

To further address the limitations of the coupling method mentioned above, a novel overset grid assembly strategy is proposed in this study. In the present method, the diffuse-interface IB method is introduced into the assembly process of weak coupling overset grid methods, which significantly differs from the HBM-IBM, as shown in Table 1. During the transfer of flow field information from the background domain to the sub-domains, to accurately impose the boundary conditions of the sub-domains, the third-order polynomial time extrapolation technique is used to estimate the pressure at the sub-domain receptor points. The velocity at the sub-domain receptor points can be obtained using the estimated pressure and the intermediate velocity, which is directly interpolated from the background domain. When the flow field information is transferred from the sub-domains to the background domain, the two additional body force terms, obtained by the diffuse-interface IB method, are added to the momentum equations for solving the background domain. The novelties of the present method include: (i) The complex hole-cutting operation is avoided, compared to the traditional overset grid methods; (ii) The iterative process between the background domain and the sub-domains is not necessary, in contrast to the traditional weak coupling overset grid methods; (iii) Compared to the HBM-IBM, there is no need to search for the background domain grid covered by the sub-domains, and only the flow field on the background domain grid near the body boundary and the artificial boundary (see Fig. 8) needs to be corrected by the diffuse-interface IB method; (iv) The novel overset grid assembly strategy can be easily combined with the IB methods to effectively suppress non-physical pressure oscillations in moving body simulations.

In the present method, the background domain, discretized with Cartesian grids, is fixed. The sub-domains, discretized with body-fitted or Cartesian grids, are moving with the body. This study primarily focuses on the information exchange between the background domain and the sub-domains. For simplicity, uniform Cartesian grids are used for both the background domain and the sub-domains. The IB method is used to impose the no-slip condition at the body boundary in the sub-domains. In addition, the motions in FSI are also simplified from induced motions to specified

motions. The rest of the paper is organized as follows. In Section 2, the numerical method is described in detail, including the governing equations, the novel overset grid assembly strategy, and the treatment of body boundaries. The results of a series of benchmark 2D cases are discussed in Section 3, including the results of Taylor-Green vortex flow (Section 3.1), lid-driven cavity flow (Section 3.2), flow past a fixed circular cylinder (Section 3.3), in-line oscillation of a circular cylinder (Section 3.4), transversely oscillating cylinder in a free-stream (Section 3.5) and flow past two tandem cylinders (Section 3.6). Conclusions are drawn in Section 4.

2. Numerical method

The overset grid assembly process typically involves the exchange of flow field information between the background domain and the sub-domains. In the novel overset grid assembly strategy, a third-order polynomial time extrapolation technique is used to accurately impose the boundary conditions of the sub-domains when flow field data is transferred from the background domain to the sub-domains. Conversely, for the transfer of flow field information from the sub-domains to the background domain, two additional body force terms, obtained by the diffuse-interface IB method, are considered in the momentum equations for solving the background domain. The first body force term implicitly imposes the no-slip condition on the body boundary in the background domain. The second body force term utilizes the information from the sub-domains to further correct the background flow field, thereby ensuring the continuity of the flow field between the background domain and the sub-domains. In the following sections, the proposed method is described in detail.

2.1 Governing equations and numerical solver

The non-inertial incompressible Navier-Stokes (N-S) equations with a body force are

$$\nabla\cdot\mathbf{u} = 0 \tag{1}$$

$$\frac{\partial\mathbf{u}}{\partial t} + [(\mathbf{u}-\mathbf{V})\cdot\nabla]\mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u} + \mathbf{f} - \boldsymbol{\Omega}\times\mathbf{u} \tag{2}$$

where $\mathbf{u}$ is the velocity vector, p is the pressure. Re denotes the Reynolds number defined by $Re = UL/\nu$, U, L and ν are the reference velocity, the reference length and the kinematic viscosity, respectively. $\mathbf{V}$ ($\mathbf{V} = \mathbf{V}' + \boldsymbol{\Omega}\times\mathbf{r}$) denotes the moving velocity of the computational domain. Here, $\mathbf{V}'$ and $\boldsymbol{\Omega}$ denote the translational velocity and rotational angular velocity of the computational domain, respectively. $\mathbf{r}$ is the position vector from the arbitrary grid node to the rotational center. $\mathbf{f}$ is the body force term. In this work, the background domain is fixed, and the sub-domains can move with the bodies. Hence, $\mathbf{V}$ and $\boldsymbol{\Omega}$ are set to zero in the background domain.

The dynamics of a body are described by Newton's second law and Euler's equation [47].

$$m_s\frac{\mathrm{d}\mathbf{V}'}{\mathrm{d}t} = \mathbf{F} + (m_s - m_f)\mathbf{g} \tag{3}$$

$$I_s\frac{\mathrm{d}\boldsymbol{\Omega}}{\mathrm{d}t} = \mathbf{T} \tag{4}$$

where m_s is the mass of the body, I_s is the moment of inertia of the body, $\mathbf{g}$ is the gravitational acceleration, m_f is the mass of the fluid displaced by the body. $\mathbf{V}'$ and $\boldsymbol{\Omega}$ represent the translational velocity and rotational angular velocity of the body, respectively. $\mathbf{F}$ and $\mathbf{T}$ represent the fluid force and fluid moment, respectively.

For a rigid body motion, the non-slip velocity boundary condition has to be satisfied on the body surface. The velocity at an arbitrary point on the body surface or inside the body is given by

$$\mathbf{U} = \mathbf{V}' + \boldsymbol{\Omega} \times \mathbf{r}' \tag{5}$$

where $\mathbf{r}'$ is the position vector from the body reference center to the arbitrary point.

In this study, the incompressible N-S equations are numerically solved on staggered uniform Cartesian grids with finite difference method. A fractional step method is adopted for the velocity-pressure coupling, in which the continuity equation is enforced by solving a pressure Poisson equation [48]. The third-order upwind scheme and second order central difference scheme are used for spatial discretization of the convective and viscous term, respectively. The second-order Adams-Bashforth scheme is adopted for temporal discretization of the convection term and viscous term [49]. In the present method, a unified set of equations is used to solve the background domain and the sub-domains. The procedures of the flow field time-advancement for each time step are summarized as in Eqs. (6) ~ (10).

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\frac{1}{2}(3\mathrm{C}^n - \mathrm{C}^{n-1}) + \frac{1}{2}(3\mathrm{D}^n - \mathrm{D}^{n-1}) - \nabla p^n - \boldsymbol{\Omega} \times \mathbf{u}^n \tag{6}$$

$$\mathbf{u}^{**} = \mathbf{u}^* + \Delta t \nabla p^n \tag{7}$$

$$\frac{\hat{\mathbf{u}} - \mathbf{u}^{**}}{\Delta t} = \mathbf{f}^{n+1} \tag{8}$$

$$\nabla^2 p^{n+1} = \frac{\nabla \cdot \hat{\mathbf{u}}}{\Delta t} \tag{9}$$

$$\mathbf{u}^{n+1} = \hat{\mathbf{u}} - \Delta t \nabla p^{n+1} \tag{10}$$

where, the superscript n denotes the current time step. $\mathbf{u}^*$, $\mathbf{u}^{**}$ and $\hat{\mathbf{u}}$ are three intermediate velocity fields. C and D represent the convective term and viscous term, $[(\mathbf{u} - \mathbf{V}) \cdot \nabla]\mathbf{u}$ and $\nabla^2 \mathbf{u}/Re$, respectively. The pressure Poisson equation (Eq. (9)) is solved by successive over relaxation (SOR) method. The calculations for the body force terms $\mathbf{f}$ in the background domain and the sub-domains are presented in Sections 2.2 and 2.3, respectively. In the present work, the gravity is neglected. The body motion is specified for simplicity. Therefore, $\mathbf{V}$ and $\boldsymbol{\Omega}$ are specified as constant.

In this study, the information exchange between the background domain and the sub-domains involves the following key aspects: (i) the treatment of boundary conditions for the sub-domains during the transfer of flow field information from the background domain to the sub-domains; (ii) the treatment of body boundaries in the background domain and the ensuring of flow field continuity in the whole computational domain when transmitting flow field data from the sub-domains to the background domain. The treatment of these key aspects is presented in Section 2.2.

2.2 Novel overset grid assembly strategy

The novel overset grid assembly strategy is described in this section. In the present method, the exchange of flow field information between the background domain and the sub-domains includes two steps: (i) the transfer of flow field data from the background domain to the sub-domains; (ii) the transfer of flow field information from the sub-domains to the background domain. In the first step, a third-order polynomial time extrapolation technique is used to accurately estimate the pressure at the sub-domain receptor points. The intermediate velocities at the receptor points are obtained from the intermediate velocity field of the background domain through interpolation, as shown in Fig. 2(a). Subsequently, the velocities at the receptor points can be computed using the

estimated pressure and the intermediate velocities. Thereby, the boundary conditions of the sub-domains are accurately imposed without iterative calculations between the background domain and the sub-domains. In the second step, two additional body force terms are added to the momentum equations of the background domain. The first body force term implicitly imposes the no-slip condition on the body boundary in the background domain. The second body force term uses flow field data from the sub-domains to further correct the background domain flow field, thereby ensuring the continuity of the flow field in the whole computational domain. In this way, the explicit flow field information interpolation process from the sub-domains to the background domain is not necessary. Therefore, the complicated hole-cutting process can be avoided. In the present method, the velocities on the body boundary and the artificial boundary are used to compute the first and second body force terms, respectively, as illustrated in Fig. 2(b).

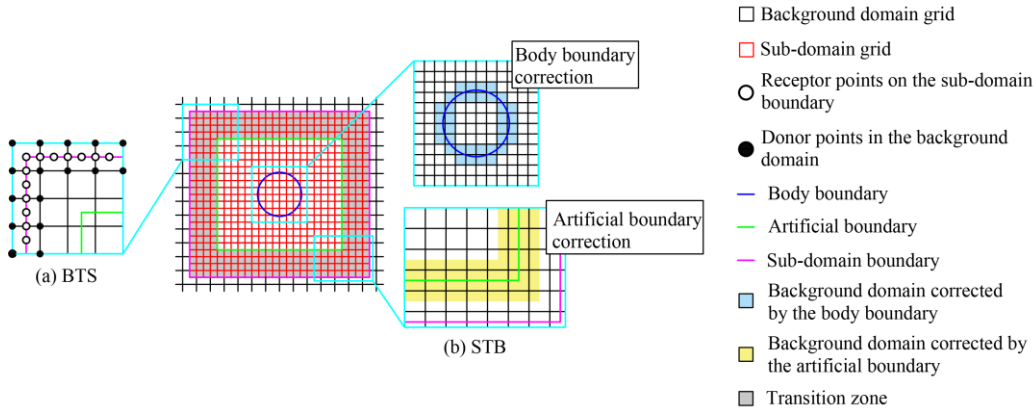


Fig. 2. Schematic demonstration of the novel overset grid assembly strategy. (BTS: the flow field information is transferred from the background domain to the sub-domains; STB: the flow field information is transferred from the sub-domains to the background domain)

2.2.1 Flow field information transfer from background domain to sub-domains

For overset grid methods, transferring flow field information from the background domain to the sub-domains is essential. In general, this transfer is achieved by imposing boundary conditions on the sub-domains. Here, for the convenience of description, the grid nodes that impose sub-domain boundary conditions are designated as receptor points, as illustrated in Fig. 3. At these receptor points, velocity and pressure are usually obtained by interpolation from the background domain.

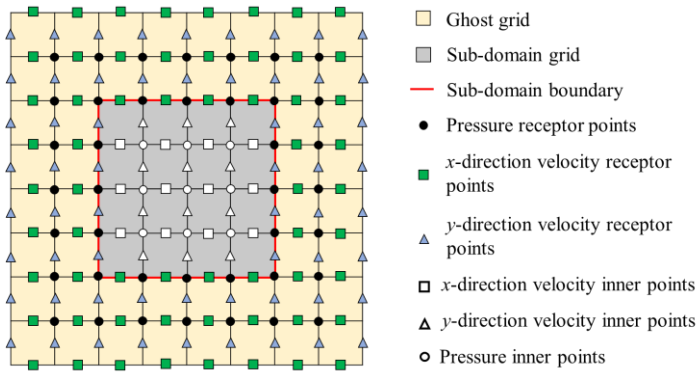


Fig. 3. Sketch of the sub-domain receptor points on staggered Cartesian grids

In overset grid methods, the accurate imposition of sub-domain boundary conditions requires that the velocity and pressure at the receptor points satisfy the following conditions:

$$p_s^{n+1}(\mathbf{x}_{receptor}^{n+1}) = P_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) \tag{11}$$

$$\mathbf{u}_s^{n+1}(\mathbf{x}_{receptor}^{n+1}) = \mathbf{U}_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) \tag{12}$$

where,

$$P_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) = I[p_b^{n+1}(\mathbf{x})] \tag{13}$$

$$\mathbf{U}_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) = I[\mathbf{u}_b^{n+1}(\mathbf{x})] \tag{14}$$

represents the interpolated pressure and velocity at the receptor points, respectively. I is the interpolation operation, the superscript $n+1$ denotes the next time step, the subscript s denotes the sub-domain, the subscript b denotes the background domain, *receptor* denotes the receptor point, $\mathbf{x}$ is the position vector, p is the pressure, $\mathbf{u}$ is velocity vector.

In traditional overset grid methods, several iterations between the background domain and the sub-domains need to be performed to satisfy Eqs. (11) and (12), as shown in Fig. 4. From Eqs. (11) and (12), it can be observed that if the right-hand side terms can be determined during the sub-domain solution, the velocity and pressure at the receptor points can be directly obtained via these equations. In this way, the boundary conditions of the sub-domains can be accurately imposed without iterative processes between the background domain and the sub-domains. In the present method, the velocity and pressure fields at the $n+1$ time step in the background domain are unknown when solving the sub-domains (see Fig. 7). Therefore, the right-hand side terms of Eqs. (11) and (12) cannot be directly obtained by interpolation from the background domain. To obtain the right-hand side terms of Eqs. (11) and (12), special treatment of the velocity and pressure at the receptor points is required.

In this study, to determine the right-hand side term of Eq. (11), a third-order polynomial time extrapolation technique is adopted, which has been used by Rajamuni et al. [50] to accurately estimate the acceleration of the body at the next time step. The third-order polynomial time extrapolation technique is applied to determine the right-hand side term of Eq. (11) as follows:

$$P_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) = 3P_b^{n}(\mathbf{x}_{receptor}^{n}) - 3P_b^{n-1}(\mathbf{x}_{receptor}^{n-1}) + P_b^{n-2}(\mathbf{x}_{receptor}^{n-2}) \tag{15}$$

where, the superscripts n, $n-1$ and $n-2$ denote the current, first preceding and second preceding time steps, respectively.

In general, receptor points are non-conformal with the grid nodes of the background domain. This leads to the requirement that the pressures P_b^n, P_b^{n-1}, and P_b^{n-2} at the receptor points be interpolated from the background domain. Here, a distance-based bilinear interpolation is used [51].

$$\Phi(\mathbf{x}_{receptor}) = I[\Phi(\mathbf{x})] = \sum_{k=1}^{nk} c_k \Phi(\mathbf{x}_k) \tag{16}$$

where, $\mathbf{x}_{receptor}$ is the position vector of the receptor point, $\mathbf{x}$ is the position vector of grid nodes in the background domain, $\mathbf{x}_k$ denotes the coordinates of a grid node in the interpolation stencil corresponding to the receptor point $\mathbf{x}_{receptor}$. I represents the interpolation operation. nk denotes the

number of grid nodes in the interpolation stencil, Φ denotes either the velocity or the pressure. c is the coefficient of the interpolation function and its specific value are detailed in Reference [51].

Substituting Eq. (16) into Eq. (15), it becomes

$$P_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) = 3I[p_b^n(\mathbf{x})] - 3I[p_b^{n-1}(\mathbf{x})] + I[p_b^{n-2}(\mathbf{x})] \tag{17}$$

Since the pressure fields in the background domain at the time steps n, $n-1$ and $n-2$ are known, the right-hand side term P_b^{n+1} of Eq. (11) can be directly determined via Eq. (17). The pressure errors at receptor points are analyzed in Sections 3.1. The results indicate that the third-order polynomial time extrapolation technique can accurately estimate the pressure at receptor points.

For the velocity field, the intermediate velocity $\mathbf{u}_b^{**}$ in the background domain has been obtained when the sub-domain is being computed (see Fig. 7). In addition, the pressure P_b^{n+1} at the receptor points has been obtained through Eq. (17). In the sub-domain, the effects of the body boundary on receptor points are negligible because these points are sufficiently far from the body boundary. Thus, the right-hand side term of Eq. (12) can be determined by replacing $\hat{\mathbf{u}}_b$ in Eq. (10) with $\mathbf{u}_b^{**}$, where $\mathbf{u}_b^{**}$ is the intermediate velocity field not affected by the body boundary.

$$\mathbf{U}_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) = \mathbf{U}_b^{**}(\mathbf{x}_{receptor}^{n+1}) - \Delta t \nabla P_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) \tag{18}$$

In Eq. (18), the intermediate velocity field $\mathbf{U}_b^{**}$ at the receptor points is obtained from the background domain via interpolation. Here, Eq. (16) is used to perform the interpolation operation. Substituting Eq. (16) into Eq. (18), it becomes

$$\mathbf{U}_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) = I[\mathbf{u}_b^{**}(\mathbf{x})] - \Delta t \nabla P_b^{n+1}(\mathbf{x}_{receptor}^{n+1}) \tag{19}$$

The desired values at the receptor points in the sub-domains can be obtained by substituting Eqs. (17) and (19) into Eqs. (11) and (12), respectively. Thereby, the boundary conditions of the sub-domains can be accurately imposed, without iterative processes between the background domain and the sub-domains.

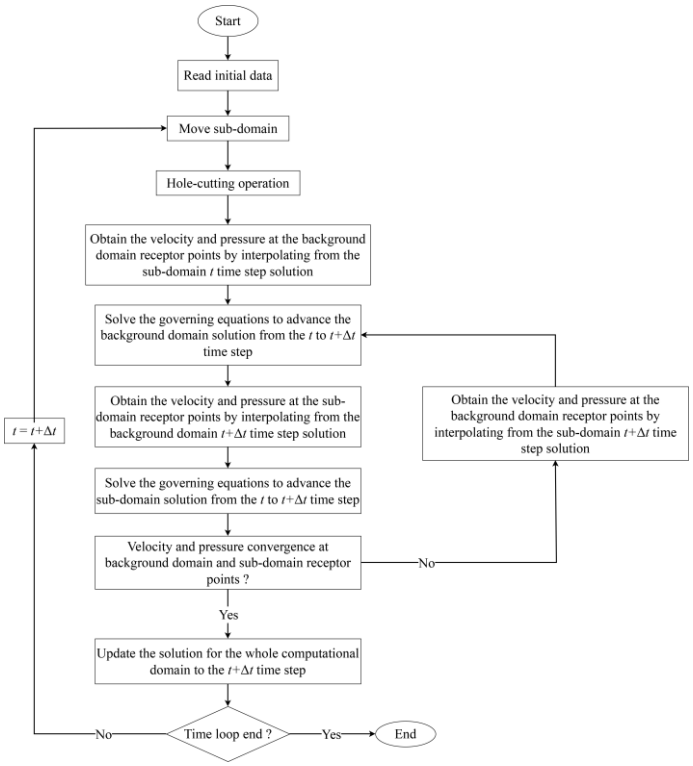


Fig. 4. Calculation process of the traditional weak coupling overset grid method.

2.2.2 Flow field information transfer from sub-domains to background domain

In the process of transferring flow field information from the sub-domains to the background domain, two additional body force terms are introduced into the background domain. The first body force term implicitly imposes the no-slip condition on the body boundary in the background domain, while the second body force term ensures the continuity of the flow field between the sub-domains and the background domain. In this way, the explicit flow field information interpolation process from the sub-domains to the background domain is not necessary. Thereby, the complicated hole-cutting process can be avoided.

The first body force term

In traditional overset grid methods, for the transfer of flow field information from the sub-domains to the background domain, it is essential to determine the overlapping interface (see Fig. 1) in the background domain. This process usually involves complex hole-cutting operations. After determining the overlapping interface, the flow field information of the sub-domains is transferred to the background domain with explicit interpolation.

In this study, to avoid complex hole-cutting operations, the first body force term is added to the momentum equations of the background domain. The first body force term implicitly imposes the no-slip boundary condition of the body in the background domain, primarily incorporating the influence of the body boundaries into the background domain flow field. In the present method, the velocities on the body boundaries are used to calculate the first body force terms based on the diffuse-interface IB method, as shown in Fig. 5.

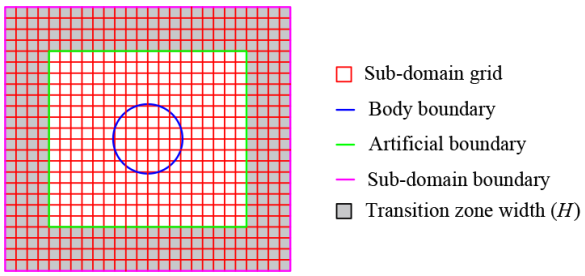


Fig. 5. Sketch of the sub-domain grid with body boundary and artificial boundary

In general, Eq. (20) is used to impose the no-slip conditions at the body boundary points in the flow field.

$$\mathbf{U}_b^{n+1}(\mathbf{X}_l^{n+1}) = \mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) \tag{20}$$

where,

$$\mathbf{U}_b^{n+1}(\mathbf{X}_l^{n+1}) = I[\mathbf{u}^{n+1}(\mathbf{x})] \tag{21}$$

represents the interpolated velocity at the body boundary points, I represents the interpolation operation, $n+1$ denotes the next time step, $\mathbf{u}$ is the velocity vector in the flow field, $\mathbf{U}$ is the body motion velocity, $\mathbf{X}_l$ is the coordinate of the body boundary points, $\mathbf{x}$ is the coordinate of the grid nodes in the flow field.

In the present method, to satisfy Eq. (20) in the background domain, the first body force term is added to the momentum equations of the background domain. The summation of Eqs. (6), (7), and (8) yields:

$$\frac{\tilde{\mathbf{u}}_b - \mathbf{u}_b^n}{\Delta t} = \mathbf{rhs}^n + \mathbf{f}_{bb}^{n+1} \tag{22}$$

where **rhs** regroups the convective and viscous terms, $\mathbf{u}_b$ is the velocity vector in the background domain, $\mathbf{f}_{bb}$ denotes the first body force term. In Eq. (8), $\mathbf{f}$ includes both the first and second body force terms. At the current step, only the first body force term is required. Thus, $\tilde{\mathbf{u}}_b$ is used instead of $\hat{\mathbf{u}}_b$ to distinguish it.

If we define $\mathbf{u}_b^{**}$ as the intermediate velocity of flow field, which is not affected by the body boundary. Eq. (22) can be rewritten in terms of it, as shown in Eq. (23).

$$\frac{\tilde{\mathbf{u}}_b - \mathbf{u}_b^{**}}{\Delta t} + \frac{\mathbf{u}_b^{**} - \mathbf{u}_b^n}{\Delta t} = \mathbf{rhs}^n + \mathbf{f}_{bb}^{n+1} \tag{23}$$

Eq. (23) is then further divided into two parts: one includes body force effect while the other does not. It can be seen that effect of the body boundary to the velocity field, which is governed by Eq. (24), is not considered. This part can be solved based on the flow field information from the previous time step.

$$\frac{\mathbf{u}_b^{**} - \mathbf{u}_b^n}{\Delta t} = \mathbf{rhs}^n \tag{24}$$

$$\frac{\tilde{\mathbf{u}}_b - \mathbf{u}_b^{**}}{\Delta t} = \mathbf{f}_{bb}^{n+1} \tag{25}$$

Based on the study of Kim et al. [52], $\mathbf{f}_{bb}^{n+1}$ is determined such that $\tilde{\mathbf{u}}_b$ satisfies the no-slip condition, with $\mathbf{u}_b^{n+1}$ satisfying the continuity equation because $\mathbf{u}_b^{n+1} = \tilde{\mathbf{u}}_b - \Delta t \nabla p^{n+1} = \tilde{\mathbf{u}}_b + o(\Delta t^2)$.

In the present method, to ensure the intermediate velocity $\tilde{\mathbf{u}}_b$ at the body boundary points satisfies Eq. (20), the body force $\mathbf{f}_{bb}^{n+1}$ is determined via Eq. (26) when the grid nodes in the background domain coincide with the body boundary points.

$$\mathbf{f}_{bb}^{n+1}(\mathbf{X}_l^{n+1}) = \frac{\mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) - \mathbf{u}_b^{**}(\mathbf{X}_l^{n+1})}{\Delta t} \tag{26}$$

Substitute Eq. (26) into Eq. (25), it becomes

$$\tilde{\mathbf{u}}_b(\mathbf{X}_l^{n+1}) = \mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) \tag{27}$$

Eq. (27) shows that no-slip condition on the body boundary is satisfied automatically by using Eq. (26), i.e. $\tilde{\mathbf{u}}_b(\mathbf{X}_l^{n+1}) = \mathbf{U}^{n+1}(\mathbf{X}_l^{n+1})$. However, the grid nodes in the background domain usually do not coincide with the body boundary points. Therefore, the body force $\mathbf{f}_{bb}^{n+1}$ in the background domain cannot be directly obtained from Eq. (26). To obtain this body force, a body force transfer mechanism between the background domain grid and the body boundary point is required. In the present study, the diffuse-interface IB method proposed by Uhlmann [53] is used to transfer body force between the background domain and the body boundary.

Firstly, the intermediate velocity at the body boundary point $\mathbf{X}_l^{n+1}$ is interpolated from the intermediate velocity field $\mathbf{u}_b^{**}$ on the background domain grid as:

$$\mathbf{U}_{bb}^{**}(\mathbf{X}_l^{n+1}) = \sum_{\mathbf{x}\in g_h} \mathbf{u}_b^{**}(\mathbf{x})\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_l^{n+1})\Delta_b{}^2 \tag{28}$$

where g_h denotes the collection of background domain grid points. $\mathbf{U}_{bb}^{**}$ is the intermediate velocity at the body boundary points, $\mathbf{X}_l^{n+1}$is the coordinate of the body boundary points at the $n+1$ time step. The variable $\mathbf{x}$ represents the coordinate of the background domain grid node, Δ_b is the grid size of the background domain. δ_{Δ_b} represents the regularized delta function. The general form of δ_{Δ_b} for the two-dimensional case is:

$$\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}) = \frac{1}{\Delta_b{}^2}\phi\left(\frac{x - X}{\Delta_b}\right)\phi\left(\frac{y - Y}{\Delta_b}\right) \tag{29}$$

with $\mathbf{x} = (x, y)$ is the position vector of the background grid node, $\mathbf{X} = (X, Y)$ are the position vectors of the body boundary points or the artificial boundary points. ϕ is a continuous 1D function depending on the specific regularized delta function.

Secondly, the body force $\mathbf{F}_{bb}$ is calculated based on the velocity difference between $\mathbf{U}^{n+1}$ and $\mathbf{U}_{bb}^{**}$ as:

$$\mathbf{F}_{bb}(\mathbf{X}_l^{n+1}) = \frac{\mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) - \mathbf{U}_{bb}^{**}(\mathbf{X}_l^{n+1})}{\Delta t} \tag{30}$$

where $\mathbf{U}^{n+1}$ is the body boundary velocity at the $n+1$ time step, $\mathbf{F}_{bb}$ represents the body force at the body boundary points, and Δt is the time step size.

Finally, the body force $\mathbf{f}_{bb}^{n+1}$ can be determined by spreading the body force $\mathbf{F}_{bb}$ to the background domain using regularized delta functions as:

$$\mathbf{f}_{bb}^{n+1}(\mathbf{x}) = \sum_{l=1}^{N_l} \mathbf{F}_{bb}(\mathbf{X}_l^{n+1})\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_l^{n+1})\Delta_b dh \tag{31}$$

where N_l is the total number of the body boundary points, and dh is the grid size of the body boundary. Substituting Eq. (31) into Eq. (25) gives the intermediate velocity $\tilde{\mathbf{u}}_b$ corrected by the

first body force term. The velocity error on the body boundary is analyzed in Sections 3.3. The results indicate that the intermediate velocity $\tilde{\mathbf{u}}_b$, corrected by the first body force term, satisfies the no-slip condition on the body boundary.

The second body force term

In overset grid methods, an overlapping zone is typically established between the background domain and the sub-domains. Continuity of the flow field between the background domain and the sub-domains is ensured by enforcing consistency of flow field information within the overlapping zone. In the overlapping zone, the velocity and pressure satisfy:

$$p_s^{n+1} = p_b^{n+1} \tag{32}$$

$$\mathbf{u}_s^{n+1} = \mathbf{u}_b^{n+1} \tag{33}$$

In traditional overset grid methods, to satisfy Eqs. (32) and (33) in the overlapping zone, several iterations need to be performed between the background domain and the sub-domains. However, the present method avoids the iterative process between the background domain and the sub-domains by using Eqs. (17) and (19). This may induce flow field discontinuities near the sub-domain boundaries, particularly when the background domain uses a coarse grid and the sub-domains use a fine grid, as shown in Figs. C1(a) and C2(a) of Appendix C. To address this issue, a transition zone is constructed near the sub-domain boundaries. In this study, the sub-domain grids at a distance of H from the sub-domain boundaries are chosen as the artificial boundaries. The transition zone is composed of the enclosed region between the artificial boundary and the sub-domain boundary, as shown in Fig. 5. The study reveals that selecting an appropriate transition zone width is crucial for ensuring numerical stability. A detailed analysis of how the transition zone width affects numerical stability is provided in Appendix B. In the present method, to achieve smooth flow field transition in the transition zone, the second body force term is added to the momentum equations of the background domain to further correct the background domain flow field. The velocities on the artificial boundaries are used to calculate the second body force terms based on the diffuse-interface IB method [53].

The intermediate velocity $\hat{\mathbf{u}}_b$ in the background domain, corrected by the second body force term, is expressed as:

$$\hat{\mathbf{u}}_b(\mathbf{x}) = \tilde{\mathbf{u}}_b(\mathbf{x}) + \Delta t \mathbf{f}_{ab}^{n+1}(\mathbf{x}) \tag{34}$$

where $\tilde{\mathbf{u}}_b$ is the intermediate velocity in the background domain corrected by the first body force term. $\mathbf{f}_{ab}^{n+1}$ represents the second body force term. The calculation of the body force $\mathbf{f}_{ab}^{n+1}$ is as follows.

Initially, the intermediate velocity at the artificial boundary point $\mathbf{X}_k^{n+1}$ is interpolated from the intermediate velocity field $\tilde{\mathbf{u}}_b$ in the background domain as:

$$\tilde{\mathbf{U}}_{ab}(\mathbf{X}_k^{n+1}) = \sum_{\mathbf{x}\in g_h} \tilde{\mathbf{u}}_b(\mathbf{x}) \delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_k^{n+1}) {\Delta_b}^2 \tag{35}$$

where $\tilde{\mathbf{U}}_{ab}$ is the intermediate velocity at the artificial boundary points, and $\mathbf{X}_k^{n+1}$is the coordinate of the artificial boundary points at the $n+1$ time step.

Subsequently, the body force $\mathbf{F}_{ab}$ is calculated based on the velocity difference between $\mathbf{U}_{ab}^{n+1}$ and $\tilde{\mathbf{U}}_{ab}$ as:

$$\mathbf{F}_{ab}(\mathbf{X}_k^{n+1}) = \frac{\mathbf{U}_{ab}^{n+1}(\mathbf{X}_k^{n+1}) - \tilde{\mathbf{U}}_{ab}(\mathbf{X}_k^{n+1})}{\Delta t} \tag{36}$$

where $\mathbf{U}_{ab}^{n+1}$ is the velocity of the artificial boundary points at the $n+1$ time step, and $\mathbf{F}_{ab}$ represents the body force at the artificial boundary points. Since the artificial boundary points coincide with the sub-domain grid nodes, this implies that $\mathbf{U}_{ab}^{n+1} = \mathbf{u}_s^{n+1}$, where $\mathbf{u}_s^{n+1}$ denotes the velocity of the sub-domain at the $n+1$ time step.

Lastly, the second body force term $\mathbf{f}_{ab}^{n+1}$ can be determined by spreading the body force $\mathbf{F}_{ab}$ to the background domain using regularized delta functions as:

$$\mathbf{f}_{ab}^{n+1}(\mathbf{x}) = \sum_{k=1}^{N_k} \mathbf{F}_{ab}(\mathbf{X}_k^{n+1}) \delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_k^{n+1}) \Delta_b ds \tag{37}$$

where N_k is the total number of the artificial boundary points, and ds is the grid size of the artificial boundary.

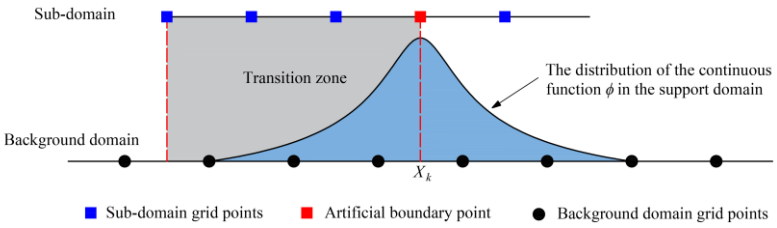


Fig. 6. Schematic diagram of one-dimensional overset grid

Substituting Eq. (37) into Eq. (34), the intermediate velocity $\hat{\mathbf{u}}_b$ in the background domain, corrected by the second body force term, is obtained. Here, a one-dimensional overset grid is applied to illustrate the specific calculation procedure of the intermediate velocity field $\hat{\mathbf{u}}_b$ corrected by the second body force term, as shown in Fig. 6. For an artificial boundary point X_k^{n+1}, substituting Eqs. (29), (35), (36), and (37) into Eq. (34) as:

$$\hat{\mathbf{u}}_b(x) = \tilde{\mathbf{u}}_b(x) + [\, \mathbf{U}_{ab}^{n+1}(X_k^{n+1}) - \tilde{\mathbf{U}}_{ab}(X_k^{n+1}) \,] \frac{1}{\Delta_b} \phi\left(\frac{x - X_k^{n+1}}{\Delta_b}\right) \Delta_b \tag{38}$$

where, X_k^{n+1} is the coordinate of the artificial boundary points at the $n+1$ time step. The variable x represents the coordinate of the background domain grid node.

Further simplifying Eq. (38), we can obtain:

$$\hat{\mathbf{u}}_b(x) = \mathbf{U}_{ab}^{n+1}(X_k^{n+1}) \phi\left(\frac{x - X_k^{n+1}}{\Delta_b}\right) + \tilde{\mathbf{u}}_b(x) - \tilde{\mathbf{U}}_{ab}(X_k^{n+1}) \phi\left(\frac{x - X_k^{n+1}}{\Delta_b}\right) \tag{39}$$

If we substitute $x = X_k^{n+1}$ into Eq. (39) with $\phi(0) = 1$, it becomes

$$\hat{\mathbf{u}}_b(X_k^{n+1}) = \mathbf{U}_{ab}^{n+1}(X_k^{n+1}) + \tilde{\mathbf{u}}_b(X_k^{n+1}) - \tilde{\mathbf{U}}_{ab}(X_k^{n+1}) \tag{40}$$

Since $\mathbf{U}_{ab}^{n+1} = \mathbf{u}_s^{n+1}$ and $\tilde{\mathbf{U}}_{ab}(X_k^{n+1}) = \tilde{\mathbf{u}}_b(X_k^{n+1})$ (see Eq. (35)), it follows that $\hat{\mathbf{u}}_b(X_k^{n+1}) = \mathbf{u}_s^{n+1}(X_k^{n+1})$. When x is outside the support domain of the regularized delta function, $\phi\left[(x - X_k^{n+1})/\Delta_b\right] = 0$. By substituting this into Eq. (39), we can obtain that $\hat{\mathbf{u}}_b(x) = \tilde{\mathbf{u}}_b(x)$.

As can be observed from Fig. 6, within the transition zone, the continuous function ϕ reaches its maximum value ($\phi(0)$) at the artificial boundary point X_k and equals zero outside the support domain. This indicates that within the transition zone, the intermediate velocity $\hat{\mathbf{u}}_b$ in the background domain, corrected by the second body force term, gradually changes from $\mathbf{u}_s^{n+1}$ to $\tilde{\mathbf{u}}_b$ as

the corresponding grid nodes in the background domain gradually transition from the artificial boundary to the outside of the support domain. After obtaining the intermediate velocity $\hat{\mathbf{u}}_b$ in the background domain, $\hat{\mathbf{u}}_b$ is used to update the pressure and velocity in the background domain. Therefore, the flow field corrected by the artificial boundary is continuous in the whole computational domain.

In the Appendix C, flow past a fixed circular cylinder and the in-line oscillation of a circular cylinder are used to illustrate the effect of the artificial boundary on the continuity of the flow field. Figs. C1 and C2 demonstrate that, for both stationary and oscillating circular cylinders, the flow field becomes discontinuous when the artificial boundary is absent. Conversely, the implementation of the artificial boundary effectively maintains flow field continuity across the entire computational domain.

In the present method, to accurately impose the no-slip boundary condition of the body in the background domain, the iterative method proposed by Kempe et al. [54] is adopted to calculate the first body force term $\mathbf{f}_{bb}^{n+1}$. In addition, for the commonly used regularized delta function, $\phi(0) \neq 1$. This leads to $\hat{\mathbf{u}}_b(X_k^{n+1}) \neq \mathbf{u}_s^{n+1}(X_k^{n+1})$. To ensure $\hat{\mathbf{u}}_b(X_k^{n+1}) = \mathbf{u}_s^{n+1}(X_k^{n+1})$, the second body force term $\mathbf{f}_{ab}^{n+1}$ is also computed using the same method [54]. The calculation of the intermediate velocity field in the background domain corrected by the first and second body force terms, which are calculated based on the iterative method, is summarized in **Algorithms 1** and **2**. Here, according to the suggestion of Kempe et al. [54], $n_f = 3$ is used.

In **Algorithms 1** and **2**, the regularized delta function δ_{Δ_b} is used. In this study, several regularized delta functions are tested when computing the first and second body force terms. These functions include the 3-point function [55], 4-point function [56], 6-point function [57], 4-point piecewise cubic function [58], smooth 3-point function [35], and smooth 4-point function [35]. The one-dimensional versions of these regularized delta functions are provided in Appendix A, with $r = (x - X)/\Delta_b$ or $r = (y - Y)/\Delta_b$.

Algorithm 1: The calculation of the intermediate velocity $\tilde{\mathbf{u}}_b$ corrected by the first body term

$\mathbf{u}_b^0(\mathbf{x}) = \mathbf{u}_b^{**}(\mathbf{x})$

do $m = 1, n_f$

(a)Interpolation:

$$\mathbf{U}_{bb}^{**}(\mathbf{X}_l^{n+1}) = \sum_{\mathbf{x}\in g_h} \mathbf{u}_b^{m-1}(\mathbf{x})\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_l^{n+1})\Delta_b{}^2$$

(b)Forcing:

$$\mathbf{F}_{bb}(\mathbf{X}_l^{n+1}) = \frac{\mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) - \mathbf{U}_{bb}^{**}(\mathbf{X}_l^{n+1})}{\Delta t}$$

(c)Distribution:

$$\mathbf{f}_{bb}^{n+1}(\mathbf{x}) = \sum_{l=1}^{N_l} \mathbf{F}_{bb}(\mathbf{X}_l^{n+1})\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_l^{n+1})\Delta_b dh$$

(d) Velocity correction:

$$\mathbf{u}_b^m(\mathbf{x}) = \mathbf{u}_b^{m-1}(\mathbf{x}) + \Delta t\mathbf{f}_{bb}^{n+1}(\mathbf{x})$$

end do

$\tilde{\mathbf{u}}_b(\mathbf{x}) = \mathbf{u}_b^{n_f}(\mathbf{x})$

Algorithm 2: The calculation of the intermediate velocity $\hat{\mathbf{u}}_b$ corrected by the second body term

$\mathbf{u}_b^0(\mathbf{x}) = \tilde{\mathbf{u}}_b(\mathbf{x})$

do $m = 1, n_f$

(a)Interpolation:

$$\widetilde{\mathbf{U}}_{ab}(\mathbf{X}_k^{n+1}) = \sum_{\mathbf{x}\in g_h} \mathbf{u}_b^{m-1}(\mathbf{x})\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_k^{n+1})\Delta_b{}^2$$

(b)Forcing:

$$\mathbf{F}_{ab}(\mathbf{X}_k^{n+1}) = \frac{\mathbf{U}_{ab}^{n+1}(\mathbf{X}_k^{n+1}) - \widetilde{\mathbf{U}}_{ab}(\mathbf{X}_k^{n+1})}{\Delta t}$$

(c)Distribution:

$$\mathbf{f}_{ab}^{n+1}(\mathbf{x}) = \sum_{k=1}^{N_k} \mathbf{F}_{ab}(\mathbf{X}_k^{n+1})\delta_{\Delta_b}(\mathbf{x} - \mathbf{X}_k^{n+1})\Delta_b ds$$

(d) Velocity correction:

$$\mathbf{u}_b^m(\mathbf{x}) = \mathbf{u}_b^{m-1}(\mathbf{x}) + \Delta t \mathbf{f}_{ab}^{n+1}(\mathbf{x})$$

end do

$\hat{\mathbf{u}}_b(\mathbf{x}) = \mathbf{u}_b^{n_f}(\mathbf{x})$

2.3 Treatment of body boundaries in sub-domains

In the present method, body-fitted or Cartesian grids can be used for the sub-domains. This study primarily focuses on the information exchange between the background domain and the sub-domains. For simplicity, uniform Cartesian grids are used for each sub-domain. The IB method is used to impose the no-slip condition on the body boundary in the sub-domains. For the IB method, the no-slip condition on the body boundary is imposed through the body force term **f** (see Eq. 8). The detailed description of imposing no-slip boundary condition via body force term **f** is presented in Section 2.2.2. Here, the calculation of the body force terms in the sub-domains is briefly introduced. In this study, the body force terms **f** are calculated by the standard diffuse-interface IB method [53], with the sub-domains discretized using an Eulerian grid and the body boundaries represented through a Lagrangian grid.

Firstly, the intermediate velocity at the Lagrangian point $\mathbf{X}_l^{n+1}$ is interpolated from the intermediate velocity field $\mathbf{u}^{**}$ (Eq. 7) on the Eulerian grid as:

$$\mathbf{U}^{**}(\mathbf{X}_l^{n+1}) = \sum_{\boldsymbol{x}\in g_h} \mathbf{u}^{**}(\mathbf{x})\delta_h(\mathbf{x} - \mathbf{X}_l^{n+1})h^2 \tag{41}$$

where $\mathbf{x}$ is the position of an Eulerian grid point, g_h is the uniform Eulerian grid with grid size of h, and δ_h is the regularized delta function.

Then, the Lagrangian force is calculated based on the velocity difference at the Lagrangian point $\mathbf{X}_l^{n+1}$ as:

$$\mathbf{F}(\mathbf{X}_l^{n+1}) = \frac{\mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) - \mathbf{U}^{**}(\mathbf{X}_l^{n+1})}{\Delta t} \tag{42}$$

with $\mathbf{U}^{n+1}$ is the Lagrangian point velocity at the $n+1$ time step.

Finally, the Eulerian body force can be determined by spreading the Lagrangian force to the Eulerian grid using regularized delta functions as:

$$\mathbf{f}^{n+1}(\mathbf{x}) = \sum_{l=1}^{N_l} \mathbf{F}(\mathbf{X}_l^{n+1})\delta_h(\mathbf{x} - \mathbf{X}_l^{n+1})h^2 \tag{43}$$

where N_l is the total number of Lagrangian points.

The general form of δ_h for the two-dimensional case is:

$$\delta_h(\mathbf{x}-\mathbf{X}) = \frac{1}{h^2}\phi\left(\frac{x-X}{h}\right)\phi\left(\frac{y-Y}{h}\right) \tag{44}$$

with $\mathbf{x} = (x, y)$ and $\mathbf{X} = (X, Y)$ are the position vectors of an Eulerian grid and a Lagrangian grid, respectively. ϕ is a continuous 1D function depending on the specific regularized delta function. Here, the 4-point regularized delta function proposed by Peskin [56] is used to calculate and distribute the body force at the Lagrangian points. The one-dimensional versions of the 4-point regularized delta function are provided in Appendix A, with $r = (x - X)/h$ or $r = (y - Y)/h$.

The calculation of the intermediate velocity field on the Eulerian grid corrected by the body force $\mathbf{f}^{n+1}$, which are calculated based on the iterative method [54], is summarized in **Algorithm 3**. Following Kempe et al. [54] recommendation, three iteration steps ($n_f = 3$) are used in this study.

Algorithm 3: The calculation of the intermediate velocity $\hat{\mathbf{u}}$ corrected by the body term $\mathbf{f}^{n+1}$

$\mathbf{u}^0(\mathbf{x}) = \mathbf{u}^{**}(\mathbf{x})$

do $m = 1, n_f$

(a)Interpolation:

$$\mathbf{U}^{**}(\mathbf{X}_l^{n+1}) = \sum_{\boldsymbol{x}\in g_h} \mathbf{u}^{m-1}(\mathbf{x})\delta_h(\mathbf{x}-\mathbf{X}_l^{n+1})h^2$$

(b)Forcing:

$$\mathbf{F}(\mathbf{X}_l^{n+1}) = \frac{\mathbf{U}^{n+1}(\mathbf{X}_l^{n+1}) - \mathbf{U}^{**}(\mathbf{X}_l^{n+1})}{\Delta t}$$

(c)Distribution:

$$\mathbf{f}^{n+1}(\mathbf{x}) = \sum_{l=1}^{N_l} \mathbf{F}(\mathbf{X}_l^{n+1})\delta_h(\mathbf{x}-\mathbf{X}_l^{n+1})h^2$$

(d) Velocity correction:

$$\mathbf{u}^m(\mathbf{x}) = \mathbf{u}^{m-1}(\mathbf{x}) + \Delta t\mathbf{f}^{n+1}(\mathbf{x})$$

end do

$\hat{\mathbf{u}}(\mathbf{x}) = \mathbf{u}^{n_f}(\mathbf{x})$

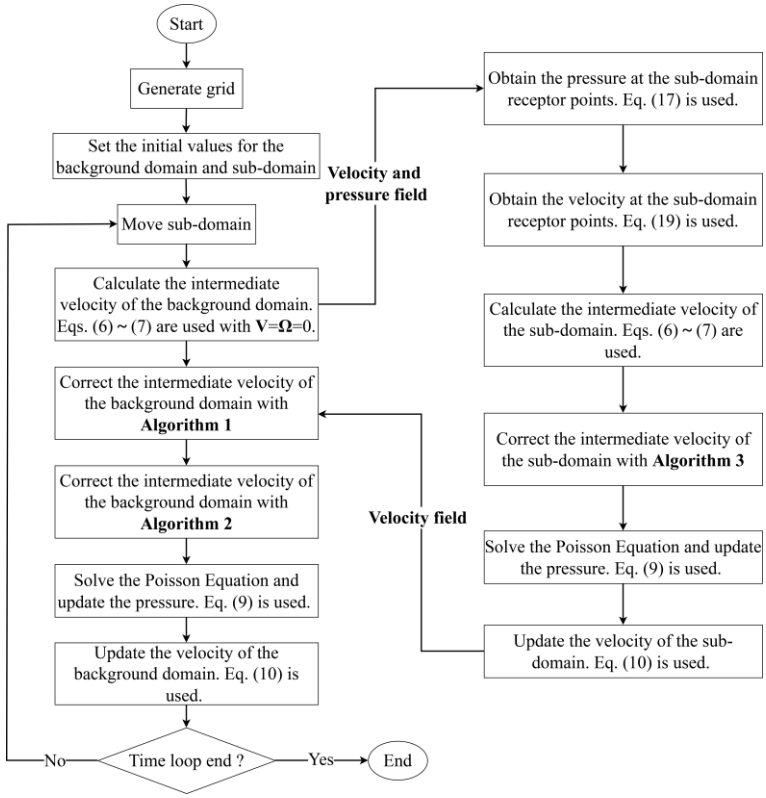


Fig. 7. Flowchart of the novel overset grid assembly method.

The flowchart of the present novel overset grid assembly method is shown in Fig. 7. The comparison between the traditional overset grid method, the HBM-IBM, and the present method is listed in Table 1. It can be seen that the complex hole-cutting operations, and the iterative process between the background domain and the sub-domains are not necessary in the present method. Additionally, compared to the HBM-IBM, there is no need to search for the background domain grid covered by the sub-domain, and only the flow field on the background domain grid near the body boundary and the artificial boundary needs to be corrected by the diffuse-interface IB method, as shown in Fig. 8. Therefore, it can save computational resources and improve computational efficiency, especially in three-dimensional simulations.

Table 1

Comparison between the traditional overset grid method, HBM-IBM, and present method.

	Traditional overset grid method	HBM-IBM	Present method
BTS	Explicit interpolation	Explicit interpolation	Explicit interpolation and time extrapolation
STB	Explicit interpolation	Explicit interpolation	Body force
DFF	Background domain and sub-domain	Background domain	Background domain and sub-domain
HC	Need	No need	No need
IT	Need	No need	No need
SBG	Need	Need	No need
CBG	No need	Need	No need

Notes: *BTS* represents the transfer of flow field information from the background domain to the sub-domain, *STB* represents the transfer of flow field information from the sub-domain to the background-domain, *DFF* represents

the computational domain displaying the flow field, *HC* represents the hole-cutting operations, *IT* represents the iterative process between the background domain and the sub-domain, *SBG* represents the search for background grid nodes covered by the sub-domain, and *CBG* represents the correction of all background grid nodes covered by the sub-domain using the sub-domain information.

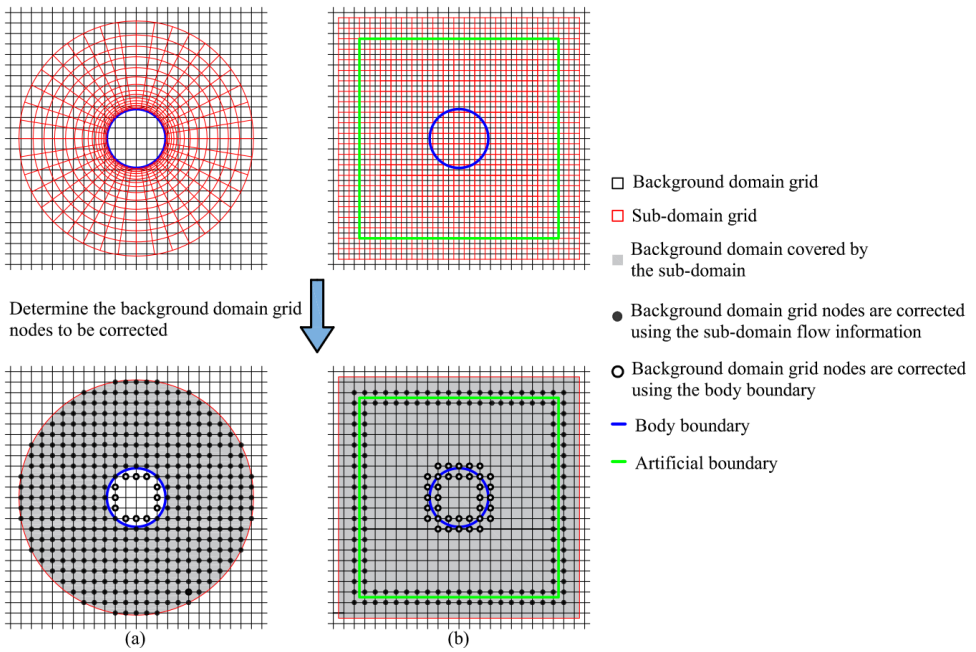


Fig. 8. The schematic diagram of the background domain flow field correction: (a) HBM-IBM; (b) present method.

3. Results and discussions

To verify and validate the novel overset grid assembly strategy, several benchmark cases have been conducted, including Taylor-Green vortex flow, lid-driven cavity flow, flow past a fixed circular cylinder, in-line oscillation of a circular cylinder, transversely oscillating cylinder in a free-stream, and flow past two tandem cylinders. It should be noted that the Taylor-Green vortex and lid-driven cavity flow simulations are conducted without considering any solid bodies. These cases solely serve to test the flow field information exchange process between the background domain and the sub-domains for the novel overset grid assembly strategy.

3.1 Taylor-Green vortex flow

The accuracy of the present method is verified by simulating the two-dimensional Taylor-Green vortex flow with analytical solution [52]:

$$u(x, y, t) = -\cos(\pi x)\sin(\pi y)\mathrm{e}^{-2\pi^2 t/Re} \tag{45}$$

$$v(x, y, t) = \sin(\pi x)\cos(\pi y)\mathrm{e}^{-2\pi^2 t/Re} \tag{46}$$

$$p(x, y, t) = -\frac{1}{4}(\cos(2\pi x) + \cos(2\pi y))\mathrm{e}^{-4\pi^2 t/Re} \tag{47}$$

In this case, the computational domain and grid system are shown in Fig. 9, wherein the size of the background domain (Ω_b) and sub-domain (Ω_s) are [0,4]×[0,4] and [1,3]×[1,3], respectively. The initial flow field and boundary conditions are specified by analytical solutions. In this paper, when flow field information is transferred from the sub-domains to the background domain, the

second body force term is considered to ensure flow field continuity. However, the first body force term is introduced to impose the no-slip condition in the background domain only when bodies exist in the computational domain. In addition, all the cases are simulated on the servers with Intel ® Xeon (R) CPU E5-2667 V2 @ 3.3 GHz in serial mode.

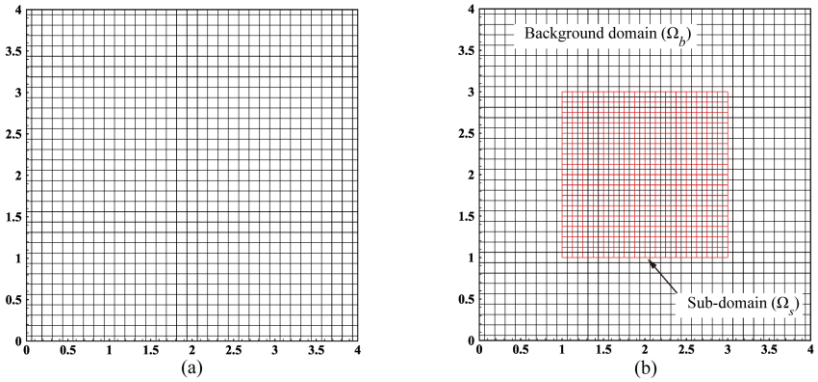


Fig. 9. Computational domain and grid system for the Taylor-Green vortex: (a) single domain solution; (b) overset grid solution.

To check the order of accuracy of the present method. Four uniform Cartesian grids are used in the whole computational domain: $\Delta x = 0.125$, 0.0625, 0.03125, and 0.015625. For all cases, in order to reduce the temporal discretization errors, the maximum Courant-Friedrichs-Lewy ($CFL = U_{max}\Delta t/\Delta x$) number is 10^{-3}. Here, U_{max} is the initial maximum velocity, Δt is the time step, and Δx is the grid size. For all the computations, the Reynolds number (Re) is set to10. At each time step, the L_∞ and L_2 errors of the velocity and pressure in the computational domain are calculated as [59]:

$$L_\infty = \left|\eta_i^{num} - \eta_i^{exact}\right|_{max} \tag{48}$$

$$L_2 = \sqrt{\sum_{i=1}^{N}(\eta_i^{num} - \eta_i^{exact})^2 \Big/ \sum_{i=1}^{N}(\eta_i^{exact})^2} \tag{49}$$

where, N is the total number of grid nodes. η_i^{num} and η_i^{exact} are the numerical and analytical solutions of velocity or pressure at the i-th grid node in the computational domain.

Fig. 10 shows the variation of the L_∞ and L_2 errors in velocity (u) and pressure (p) with the grid size. The errors are obtained from the single domain solution (i.e., without the sub-domain, see Fig. 9(a)). The results indicate that the single domain solution is second-order accurate in both velocity and pressure.

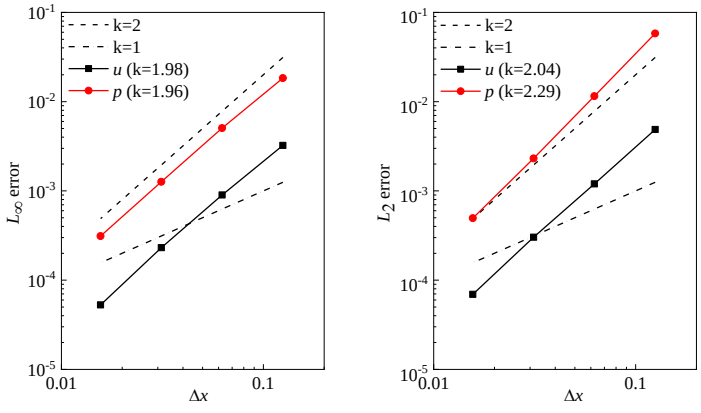
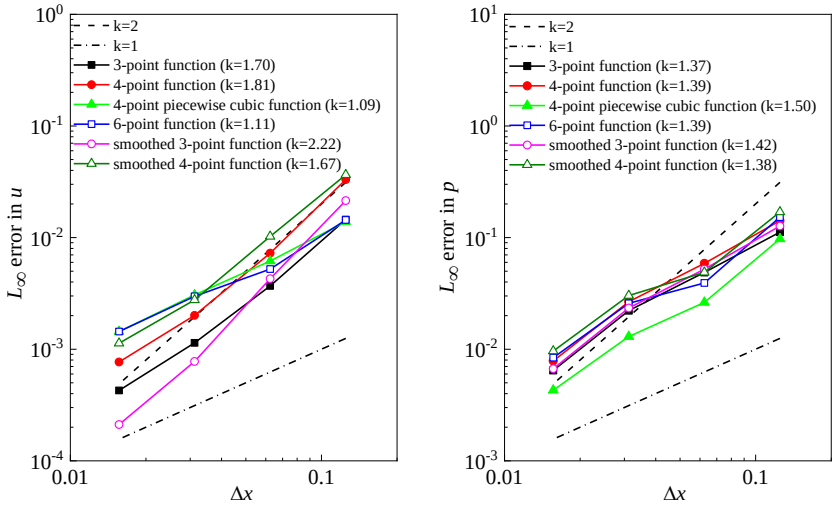
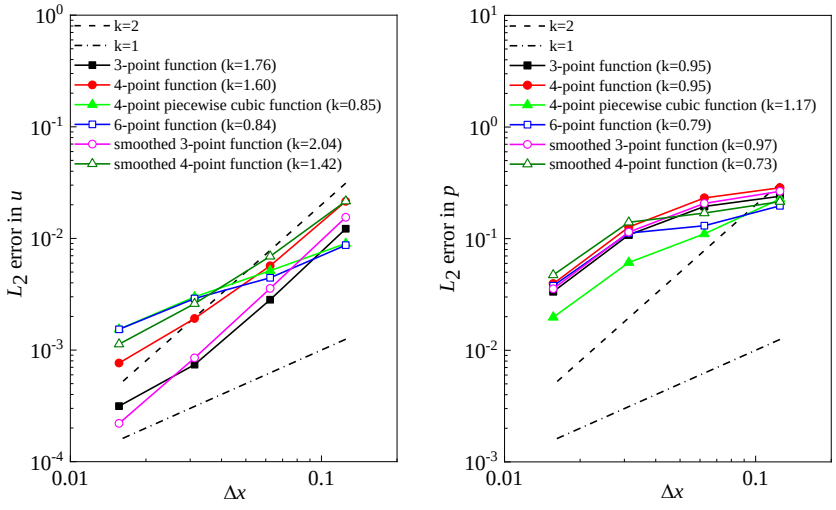


Fig. 10. L_∞ error (left) and L_2 error (right) of the velocity and pressure field at $t = 0.3$s.

The L_∞ and L_2 errors of the velocity (u) and pressure (p), obtained from the overset grid solution (i.e., with the sub-domain, see Fig. 9(b)), are shown in Fig. 11. Here, several regularized delta functions are used to calculate the second body force term, including the 3-point function [55], 4-point function [56], 4-point piecewise cubic function [58], 6-point function [57], smoothed 3-point function [35], and smoothed 4-point function [35]. In the present method, ensuring computational stability requires selecting an appropriate transition zone width for the specified regularized delta function. In this case, the minimum transition zone width in Table B1 (see Appendix B) is used. As shown in Fig. 11, it can be observed that the velocity accuracy decreases for all regularized delta functions, except for the smoothed 3-point function, compared to the single-domain solution. This discrepancy may be attributed to the varying peak values and influence ranges of different regularized delta functions, which can directly affect computational accuracy. For all regularized delta functions, the pressure is only about 1 to 1.5 orders accuracy. In the diffuse-interface IB method, due to the presence of the immersed boundary, the L_∞ and L_2 errors of the pressure are limited to orders of one and about 1.5, respectively [60]. In the overset grid method, the pressure accuracy is limited to 1 to 1.5 orders when linear interpolation is used for information exchange between the background domain and the sub-domains [61]. For the present method, the body force at the artificial boundary is obtained by the diffuse-interface IB method. Furthermore, bilinear interpolation is applied to transmit flow field information from the background domain to the sub-domains. Therefore, in the present simulations, the pressure accuracy is only about 1 to 1.5 orders for all regularized delta functions. It should be emphasized that velocity errors are more important than pressure errors, due to the fact that the gradient of pressure plays a pivotal role in incompressible flows, not the pressure itself [62]. As illustrated in Fig. 11, among the various regularized delta functions, the smooth 3-point function achieves second-order spatial accuracy for velocity in both L_∞ and L_2 errors. As a result, the smooth 3-point function is adopted for calculating the first and second body force terms in all following cases.



(a) L_∞ error of the velocity and pressure field



(b) L_2 error of the velocity and pressure field

Fig. 11. L_∞ error and L_2 error of the velocity (left) and pressure (right) field at $t = 0.3$s.

Fig. 12 illustrates the variation of pressure error at the sub-domain receptor points over time steps. The pressure error is calculated as follows:

$$L_\infty = \left| p_i^{ext} - p_i^{n+1} \right|_{max} \tag{50}$$

$$L_2 = \sqrt{\sum_{i=1}^{N} \left(p_i^{ext} - p_i^{n+1}\right)^2 \Big/ \sum_{i=1}^{N} \left(p_i^{n+1}\right)^2} \tag{51}$$

where, p_i^{ext} represents the pressure at the $n+1$ time step obtained by time extrapolation at the sub-domain receptor points, p_i^{n+1} represents the pressure at the sub-domain receptor points interpolated from the background domain flow field at the $n+1$ time step, and N is the total number of sub-domain receptor points.

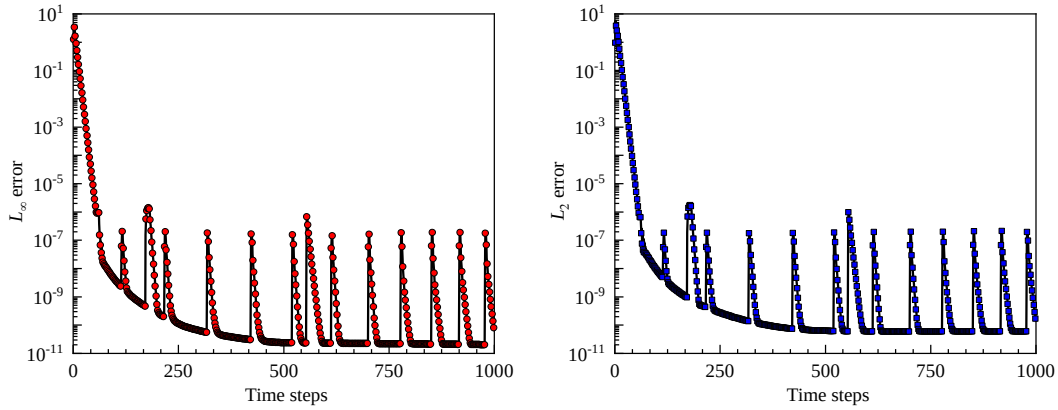


Fig. 12. Variation of pressure error with time step at the receptor points of the sub-domain.

Fig. 12 shows that a significant pressure error is observed during the initial time steps, which can be attributed to the identical initial values assigned to p_b^{n-2}, p_b^{n-1}, and p_b^n in Eq. (17). As the time steps advance, the pressure error rapidly decreases to 10^{-5} due to p_b^{n-2}, p_b^{n-1} and p_b^n being updated with computational results. As a result, the third-order polynomial time extrapolation technique can accurately estimate the pressure at the sub-domain receptor points for the $n+1$ time step. The distribution of pressure contours in the whole computational domain is shown in Fig. 13. It can be seen that using the third-order polynomial time extrapolation technique to estimate the pressure at the sub-domain receptor points can ensure the continuity of the pressure in the entire computational domain.

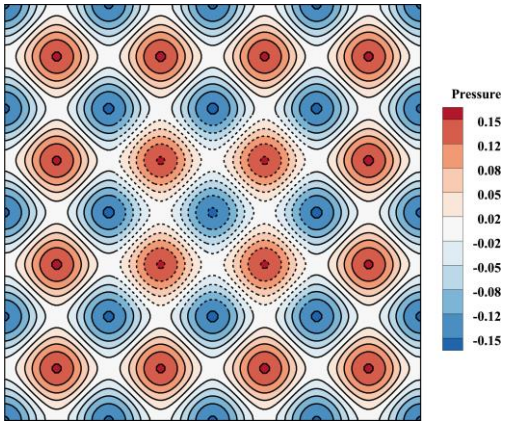
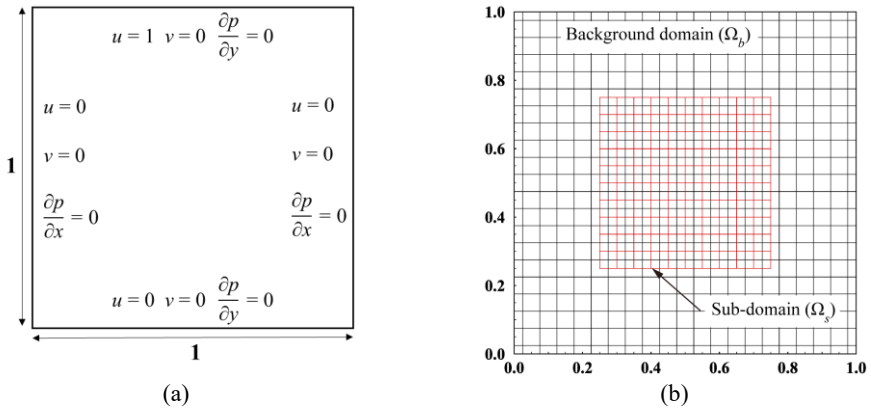


Fig. 13. The pressure distribution in the computational domain at $t = 0.3$s. (Solid line: background domain solution, dashed line: sub-domain solution, $\Delta x = 0.03125$)

3.2 Lid-driven cavity flow

In this section, 2D lid-driven cavity flow is used to validate the present method. In this case, the computational domain and boundary conditions are shown in Fig. 14(a), and the background domain grid (Ω_b) and sub-domain grid (Ω_s) are demonstrated in Fig. 14(b). The computational domain sizes for the background domain and the sub-domain are [0,1]×[0,1] and [0.25,0.75]× [0.25,0.75], respectively. The Reynolds number (*Re*) is 1000, the *CFL* number based on the minimum grid size Δx, the lid velocity, and the time step is $CFL = U_{max}\Delta t/\Delta x = 0.1$ (i.e., $U_{max} = 1.0$).



(a) (b)

Fig. 14. Boundary condition and grid layout for lid-driven cavity flow. (a) Boundary condition; (b) Grid layout with background domain and sub-domain.

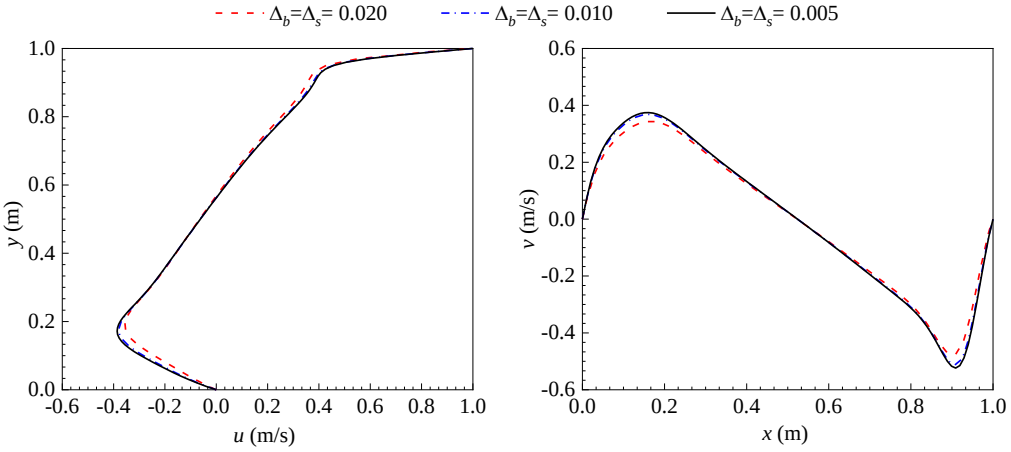


Fig. 15. Distribution of horizontal velocity (u) and vertical velocity (v) along the center lines of the cavity for three uniform mesh sizes. (Transition zone width (H) = $3\Delta_b$, Re = 1000, CFL number = 0.1)

In this study, the grid sizes in the background domain and the sub-domain are denoted as Δ_b and Δ_s, respectively. Fig. 15 shows the velocity profiles along the horizontal and vertical centerlines using three different uniform grid resolutions of $\Delta_b = \Delta_s = 0.02$, 0.01 and 0.005. It is observed that a grid resolution of $\Delta_b = \Delta_s = 0.01$ is sufficient to achieve grid-independent results. Therefore, the grid resolution of $\Delta_b = \Delta_s = 0.01$ are used for further validation. Fig. 16 presents the contour lines of the velocity and pressure fields with the transition zone width (H) of $3\Delta_b$ (see Table B1). The results indicate that the present method is able to obtain a continuous flow field in the whole computational domain.

In Fig. 17, the horizontal and vertical velocities along the vertical and horizontal centerlines are compared with the benchmark results of Ghia et al. [63]. As illustrated, the results from both the overset grid and the single domain (only using the background domain Ω_b) correspond well with the reference results.

The above validations demonstrate that the present method can not only accurately predict the flow fields of the lid-driven cavity flow but also maintain the continuity of velocity and pressure in the whole computational domain.

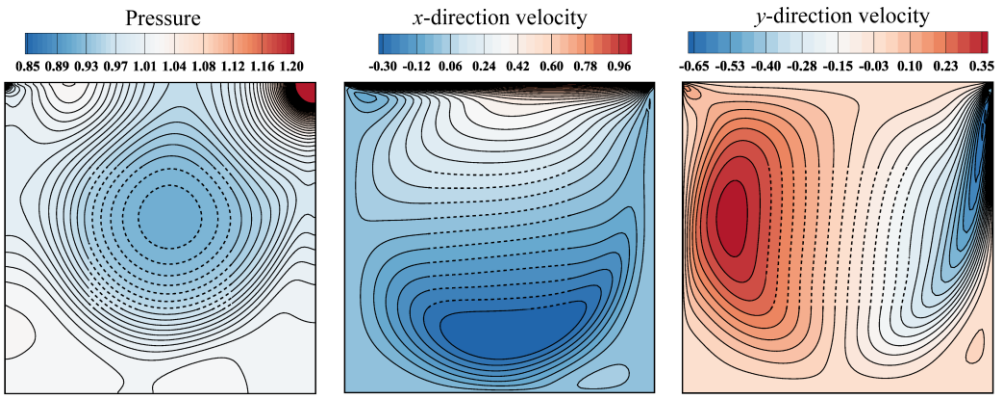


Fig. 16. The velocity and pressure distribution in the computational domain. (Transition zone width (H) = $3\Delta_b$, solid line: background domain solution, dashed line: sub-domain solution, Re = 1000, CFL number = 0.1, $\Delta_b = \Delta_s$ = 0.01)

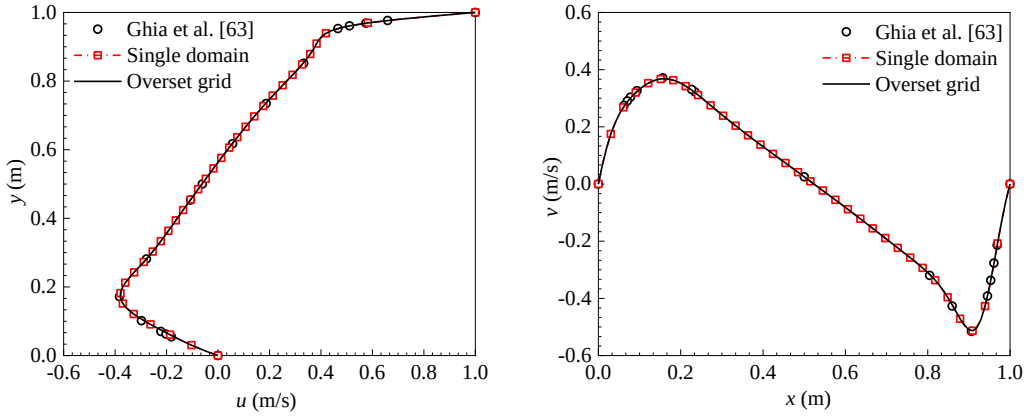
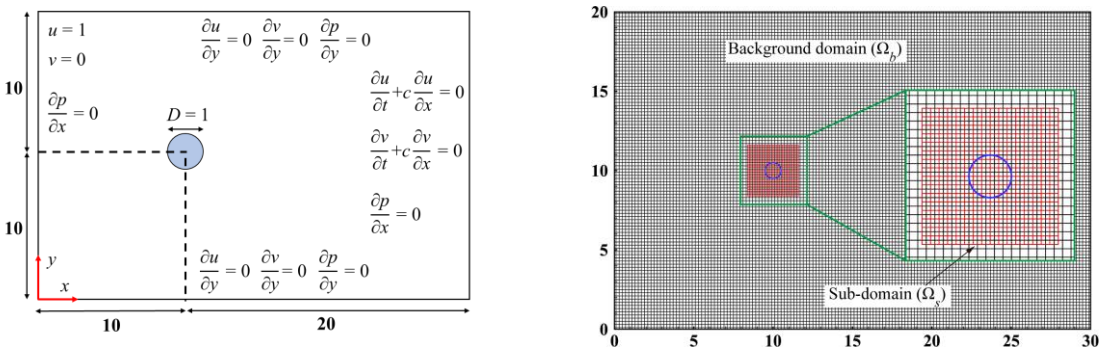


Fig. 17. The comparison of horizontal velocity (u) and vertical velocity (v) along the center lines of the cavity with the benchmark solution of Ghia et al. [63]. (Transition zone width (H) = $3\Delta_b$, $\Delta_b = \Delta_s$ = 0.01, Re = 1000, CFL number = 0.1)

3.3 Flow past a fixed circular cylinder

The flow past a fixed cylinder is simulated to further check the accuracy of the present method. In this simulation, two cases with Reynolds number (Re) of 100 and 185 are considered. The computation is performed in the background domain and the sub-domain. The domain sizes for the background domain and the sub-domain are [0,30]×[0,20] and [7,13]×[7,13], respectively. The cylinder position and the boundary conditions are shown in the Fig. 18(a), and the grid layouts are given in the Fig. 18(b). In the Fig. 18(a), D denotes the cylinder diameter, and the convection velocity c is set as the freestream velocity [64].



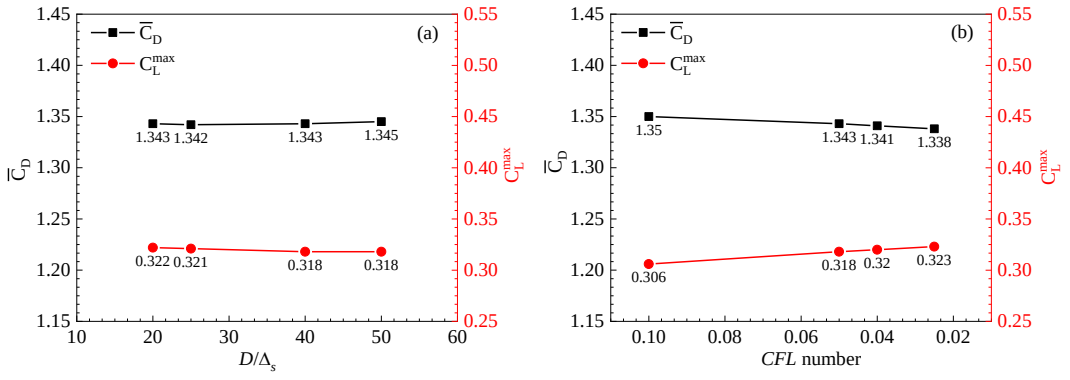
(a) Boundary conditions (b) Overset grid layout

671 **Fig. 18.** Boundary conditions of the computational domain and a schematic diagram of the grid layout. (Black line:
672 background domain grid, red line: sub-domain grid, blue line: cylinder)
673
674 Firstly, the mesh-independence test and time step size verifications are conducted for a fixed
675 circular cylinder with a *Re* of 100, and all the cases are simulated for 200s. In this paper, the force
676 acting on the cylinder is calculated [65]:

$$F_x = \rho \int_{S_b} f_x \, \mathrm{ds} - \rho \Lambda \dot{u}_c(t) \tag{52}$$

$$F_y = \rho \int_{S_b} f_y \, \mathrm{ds} - \rho \Lambda \dot{v}_c(t) \tag{53}$$

677 where ρ is the density, S_b is the cylinder surface, f_x and f_y are the Lagrangian force, Λ is the area of
678 the cylinder, $\dot{u}_c$ and $\dot{v}_c$ is the accelerations of cylinder motion.
679 The mesh-independence test is performed using four different grid resolutions of $D/\Delta_b =$
680 $D/\Delta_s = 20$, 25, 40 and 50 with a *CFL* number of 0.05. The *CFL* number based on the free-stream
681 velocity U_∞, the time step size Δt, and the sub-domain grid size Δ_s is $CFL = U_\infty \Delta t/\Delta_s$. As shown in
682 Fig. 19(a), a grid resolution of $D/\Delta_b = D/\Delta_s = 40$ is sufficient for obtaining mesh-independent
683 results. Therefore, this grid resolution is adopted for further time step size verifications. Fig. 19(b)
684 depicts the time-averaged drag coefficient and peak lift coefficient for *CFL* numbers of 0.100, 0.050,
685 0.040, and 0.025. As shown, the differences between the results of $CFL = 0.04$ and $CFL = 0.025$
686 are well below 1%. Based on the overall considerations of accuracy and efficiency, the grid
687 resolution of $D/\Delta_b = D/\Delta_s = 40$ and the time step size corresponding to $CFL = 0.04$ are adopted
688 for the following simulations.
689



690
691 **Fig. 19.** Variation of lift coefficient and drag coefficient of cylinders under different grid resolutions (a) and *CFL*
692 numbers (b).

For the overset grid method, it is important to assess the additional computational cost. This additional computation includes the overset grid operations (such as hole-cutting, interpolation searching, and iteration between the background domain and the sub-domains) and the solving of additional flow fields in the sub-domains. Table 2 compares the computational efficiency of the present method with the single-domain solution (IB method) at different grid resolutions for simulating flow past a fixed cylinder at $Re = 100$. As shown in Table 2, the additional computational time for the present method is less than 10%. This indicates that the present method does not significantly increase computational time compared to the single domain solution.

Table 2

Comparison of total CPU time between the IB method and the present method.

	Total CPU time (h)		
D/Δ_s	IB method (T_1)	Present method (T_2)	$(T_2 - T_1)/T_1 \times 100\%$
20	6.024	6.616	9.827
25	10.923	11.908	9.018
40	44.548	48.477	8.820
50	89.736	94.816	5.661

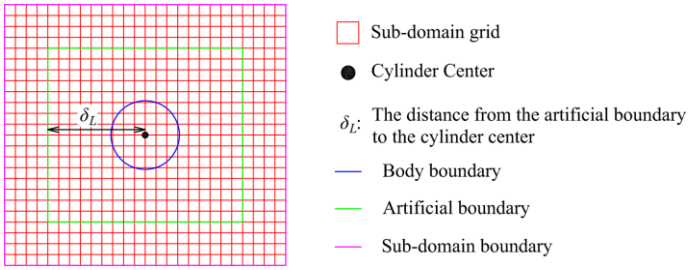


Fig. 20. The schematic diagram of the relative position between the artificial boundary and the cylinder.

In the overset grid method, the size of the sub-domain will also have an impact on the computation results. Here, the size of the sub-domain is controlled by changing the value of δ_L. The variable δ_L represents the distance from the artificial boundary to the center of the cylinder, as shown in Fig. 20. In all simulations, a transition zone width of $3\Delta_b$ is used. The single-domain solution, obtained using the IB method on a uniform grid ($\Delta x = 0.025$), provides a benchmark for evaluating the present method. The variations of the mean drag coefficient and the max lift coefficient as a function of the sub-domain size are shown in Fig. 21. For $\Delta_b = \Delta_s$, the intermediate velocity fields corrected by the body force term are consistent in the background domain and the sub-domain. At this point, the influence of sub-domain size on the results can be ignored, as shown in Fig. 21. For $\Delta_b > \Delta_s$, the intermediate velocity fields obtained by Eq. (7) are different in the background domain and the sub-domain. In addition, the body force on the body boundary differs in the background domain and the sub-domain due to $\Delta_b > \Delta_s$. This leads to differences in the intermediate velocity field corrected by the body force on the body boundary in the two domains. Under such conditions, the size of the sub-domain will affect the calculation results, as shown in Fig. 21. However, compared with the single-domain results, the maximum error of the average drag coefficient is only 0.37%, and the maximum error of the maximum lift coefficient is only 2.18%.

As the size of the sub-domain decreases, the error between the results of the present method and the single-domain results gradually decreases. In this study, in order to mitigate the effects of sub-domain size, $\delta_L/D = 1.0$ is used when $\Delta_b/\Delta_s = 2$.

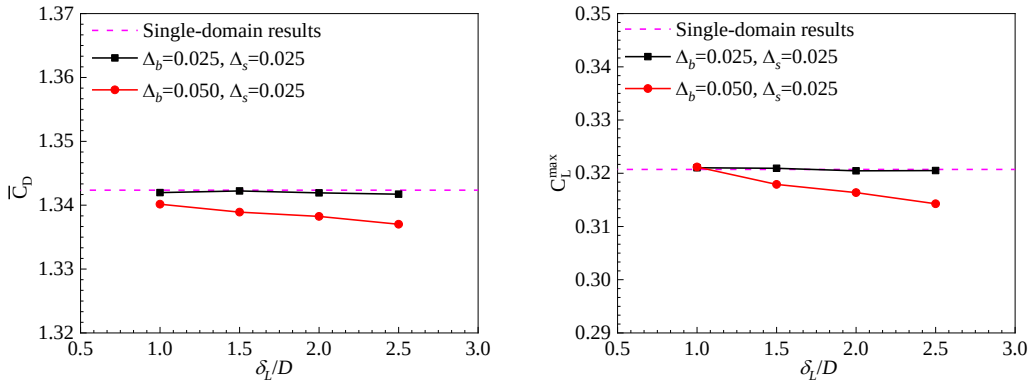


Fig. 21. Variation of average drag coefficient (left) and maximum lift coefficient (right) with δ_L/D. (Transition zone width (H) = $3\Delta_b$, Re = 100, CFL number = 0.04)

In the transition zone, the relative grid size (Δ_b/Δ_s) between the background domain and the sub-domain directly affects the accuracy and continuity of the simulation result. To test the effect of the relative grid size, three background domain grid resolutions of D/Δ_b = 10, 20, and 40 are performed, and a constant grid resolution of $D/\Delta_s = 40$ is used for the sub-domain. The Fig. 22. displays that the contour lines of the velocity and pressure fields become discontinuous when the resolution of background domain is $D/\Delta_b = 10$. However, the background domain resolutions of $D/\Delta_b = 20$ and $D/\Delta_b = 40$ ensure the continuity of the flow field contour lines. This indicates that, within the transition zone, the grid size of the background domain should be close the sub-domain grid size to achieve accurate and continuous results. In addition, it can be seen that $H = 3\Delta_b$ is adequate to ensure the continuity of the flow field in the entire computational domain for the background domain resolutions of $D/\Delta_b = 20$ and $D/\Delta_b = 40$. In this paper, considering the overall factors of accuracy, efficiency, and flow field continuity, the grid resolutions of $D/\Delta_b = 20$, $D/\Delta_s = 40$, $H = 3\Delta_b$, and $\delta_L/D = 1.0$ are adopted for all the following simulations unless otherwise stated.

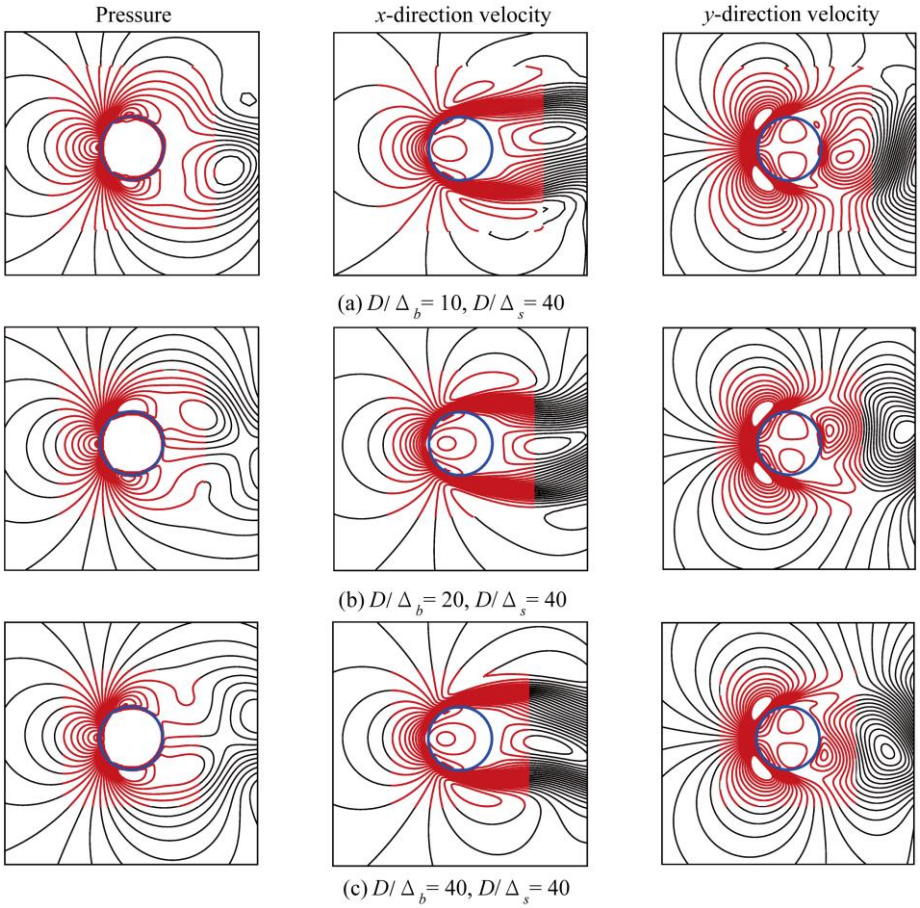


Fig. 22. The distribution of pressure and velocity contours in the computational domain with different background domain grid sizes. (Black line: background domain solution, red line: sub-domain solution, blue line: cylinder, transition zone width (H) = $3\Delta_b$, Re = 100, CFL number = 0.04, δ_L/D = 1.0)

To assess the accuracy of the no-slip condition imposed by the first body force term in the background domain, the time-averaged velocity error on the body boundary was computed, as shown in Fig. 23. The error of the velocity is defined as follows:

$$E(\mathbf{X}_l^{n+1}) = \left|\tilde{\mathbf{U}}_b(\mathbf{X}_l^{n+1}) - \mathbf{U}^{n+1}(\mathbf{X}_l^{n+1})\right| \tag{54}$$

where $\mathbf{X}_l^{n+1}$ is the coordinate of the body boundary points at the $n+1$ time step, $\mathbf{U}^{n+1}$ is the desired velocity at the body boundary. $\tilde{\mathbf{U}}_b$ denotes the velocity on the body boundary interpolated from the background domain intermediate velocity field $\tilde{\mathbf{u}}_b$. Here, $\tilde{\mathbf{u}}_b$ is the intermediate velocity field in the background domain corrected by the first body force term.

As shown in Fig. 23, the maximum velocity error occurs at the stagnation point, with an approximate value of 1.42×10^{-3}. This indicates that the intermediate velocity field $\tilde{\mathbf{u}}_b$, corrected by the first body force term, satisfies the no-slip condition on the body boundary. Therefore, the present method can implicitly impose the no-slip condition in the background domain by adding the first body force term to the momentum equation of the background domain. Furthermore, the conservation properties of the present method are also discussed, with a detailed analysis provided in Appendix D.

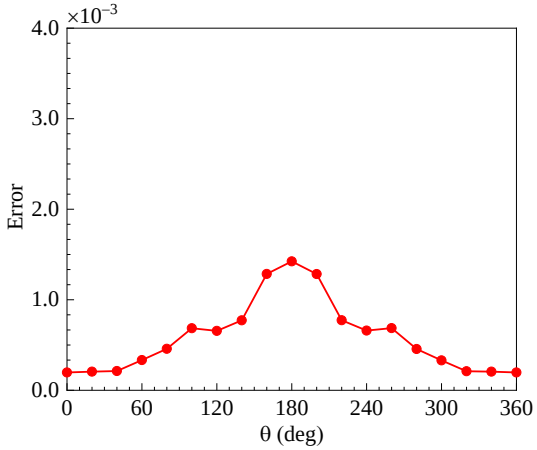


Fig. 23. The time-averaged velocity error at the body boundary in the background domain. (Transition zone width $(H) = 3\Delta_b$, $Re = 100$, CFL number $= 0.04$, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

The time-averaged drag coefficients ($\overline{C}_D$), root-mean-square of lift coefficients (C_L^{max}) and the Strouhal numbers (St) are compared with reference results in Table 3 and Table 4 for the two Reynolds numbers of 100 and 185, respectively. As shown in the table, the current results align well with the reference data.

Table 3

Comparison of drag coefficient, lift coefficient, and St with $Re = 100$

Source	Method	$\overline{C}_D$	C_L^{max}	St
Present	-	1.340	0.321	0.161
Williamson [66]	Experiment	-	-	0.164
Uhlmann [53]	Diffuse-interface IB method	1.453	0.339	0.169
Ji et al. [49]	Diffuse-interface IB method	1.402	0.349	0.167
Yu et al. [59]	Sharp-interface IB method	1.500	0.390	0.167
Shen et al. [65]	Diffuse-interface IB method	1.376	0.325	0.166
Lai & Peskin [67]	Diffuse-interface IB method	1.447	0.329	0.160

Table 4

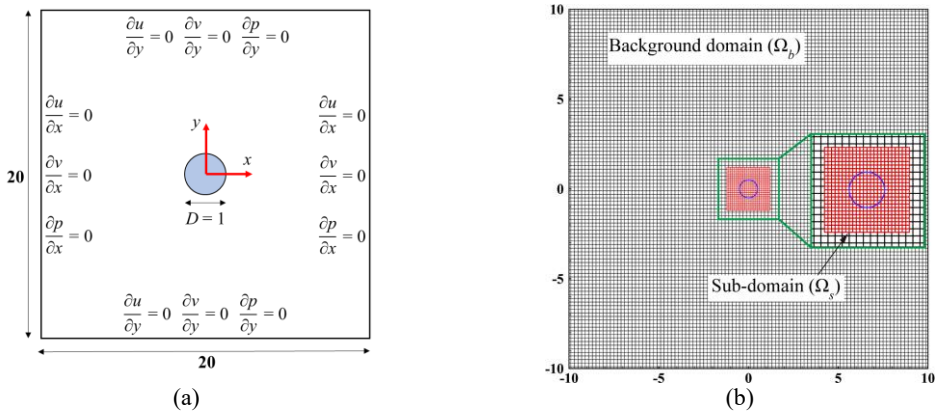
Comparison of drag coefficient, lift coefficient, and St with $Re = 185$

Source	Method	$\overline{C}_D$	C_L^{rms}	St
Present	-	1.349	0.441	0.189
Guilmineau and Queutey [68]	Experiment	1.280	-	0.190
Guilmineau and Queutey [68]	Body-fitted grid	1.287	0.443	0.195
Lu and Dalton [69]	Body-fitted grid	1.310	0.422	0.195
Gronskis and Artana [70]	Sharp-interface IB method	1.440	0.596	0.195

3.4 In-line oscillation of a circular cylinder

To investigate the accuracy of the present method in moving boundary problems, in-line oscillation of a cylinder is computed. The in-line motion of the cylinder is given as $x(t) =$

$-A\sin(2\pi ft)$, where A and f are the oscillating amplitude and frequency, respectively. In this simulation, there are two key parameters: Reynolds number ($Re = U_{max}D/\nu$) and Keulegan-Carpenter number ($KC = U_{max}/fD$), where U_{max} is the maximum moving velocity of the circular cylinder and D is the cylinder diameter. To compare with the experimental data and numerical results of Dutsch [71], simulations are performed with $Re = 100$ and $KC = 5$. In this case, the background domain size is [-10,10]×[-10,10] and the sub-domain size is [-1.15,1.15]×[-1.15,1.15]. The initial position of the cylinder is located at the center of the computational domain. The boundary conditions and grid layout are given in Fig. 24.



(a) (b)

Fig. 24. Boundary conditions and grid layouts for in-line oscillating circular cylinder. (a) Boundary conditions; (b) Grid layouts with background domain (black line: Ω_b), sub-domain (red line: Ω_s), and cylinder (blue line).

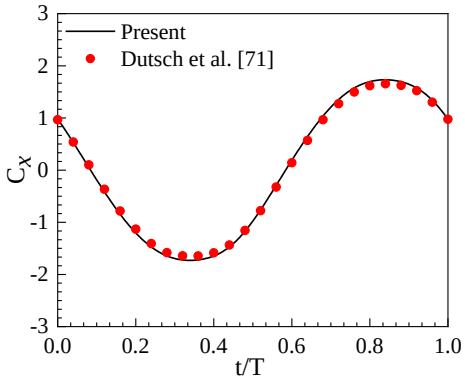


Fig. 25. Comparison of the instantaneous in-line force coefficient C_x of an in-line oscillating circular cylinder. (Transition zone width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

In Fig. 25, the instantaneous in-line force coefficient C_x is contrasted with the results of Dutsch et al. [71], wherein C_x is defined as $C_x = F_x/(\rho DU_{max}^2)$ with F_x being the in-line force on the circular cylinder. It is evident that the results of the present method are in good agreement with the numerical results of Dutsch et al. [71].

The velocity components along the transverse direction (y) at four in-line positions of x = -0.6, 0.0, 0.6 and 1.2 are compared with the experimental and numerical results of Dutsch et al. [71] for three different phase positions of 180°, 210° and 330° in Fig. 26, 27, and 28, respectively. As shown, very good agreement is reached in the comparisons.

Fig. 29 shows the pressure and vorticity fields at different phase angles. Smooth transitions of

809 the pressure and vorticity contours are observed at the boundaries of the sub-domain. These results
810 demonstrate that the present method is able to accurately predict the flow field induced by a moving
811 boundary, and the continuity of physical quantities is ensured in the whole computational domain.
812 Table 5 summarizes the computational time required for each component of the present method.
813 To compare the present method with the single-domain solution (IB method), the background grid
814 resolution of the present method is adopted for the single-domain solution. In Table 5, the total CPU
815 time represents the overall computational cost over 30,000 time steps. As can be seen from Table 5,
816 compared with the single-domain solution, the total computational cost of the present method only
817 increases by 11.67%. Among them, the sub-domain flow field solution accounts for 4.45%, and the
818 overset grid assembly process accounts for 2.69%. Post-processing (including flow field output and
819 residual calculation, etc.) and sub-domain coordinate transformation account for 4.53%. The results
820 show that the present method does not significantly increase the computational cost compared with
821 the single-domain solution when simulating moving bodies. Moreover, the overset grid assembly
822 process only increases the computational cost by 2.69%. This highlights that the novel overset grid
823 assembly strategy has good computational efficacy.
824
825 **Table 5**
826 Total CPU time (h) of each component in the present method

	IB method (I)	Present method (P)	P_T1/I_T3	P_T2/I_T3	(P_T3-I_T3)/I_T3
Sub-domain time (T1)	-	0.248	-	-	-
OGA time (T2)	-	0.150	-	-	-
Total time (T3)	5.57	6.22	-	-	-
P/I×100%	-	-	4.45	2.69	11.67

827 Notes: Sub-domain time represents the total CPU time required for the sub-domain flow field solution. OGA time
828 represents the total CPU time for the overset grid assembly process, which includes determining donor and receptor
829 points, computing the first and second body force terms, and performing interpolation operations. Total time
830 represents the total CPU time for the whole computational domain solution, including the sub-domain flow field
831 solution, the overset grid assembly process, the background domain flow field solution, the coordinate
832 transformation, and the post-processing.
833

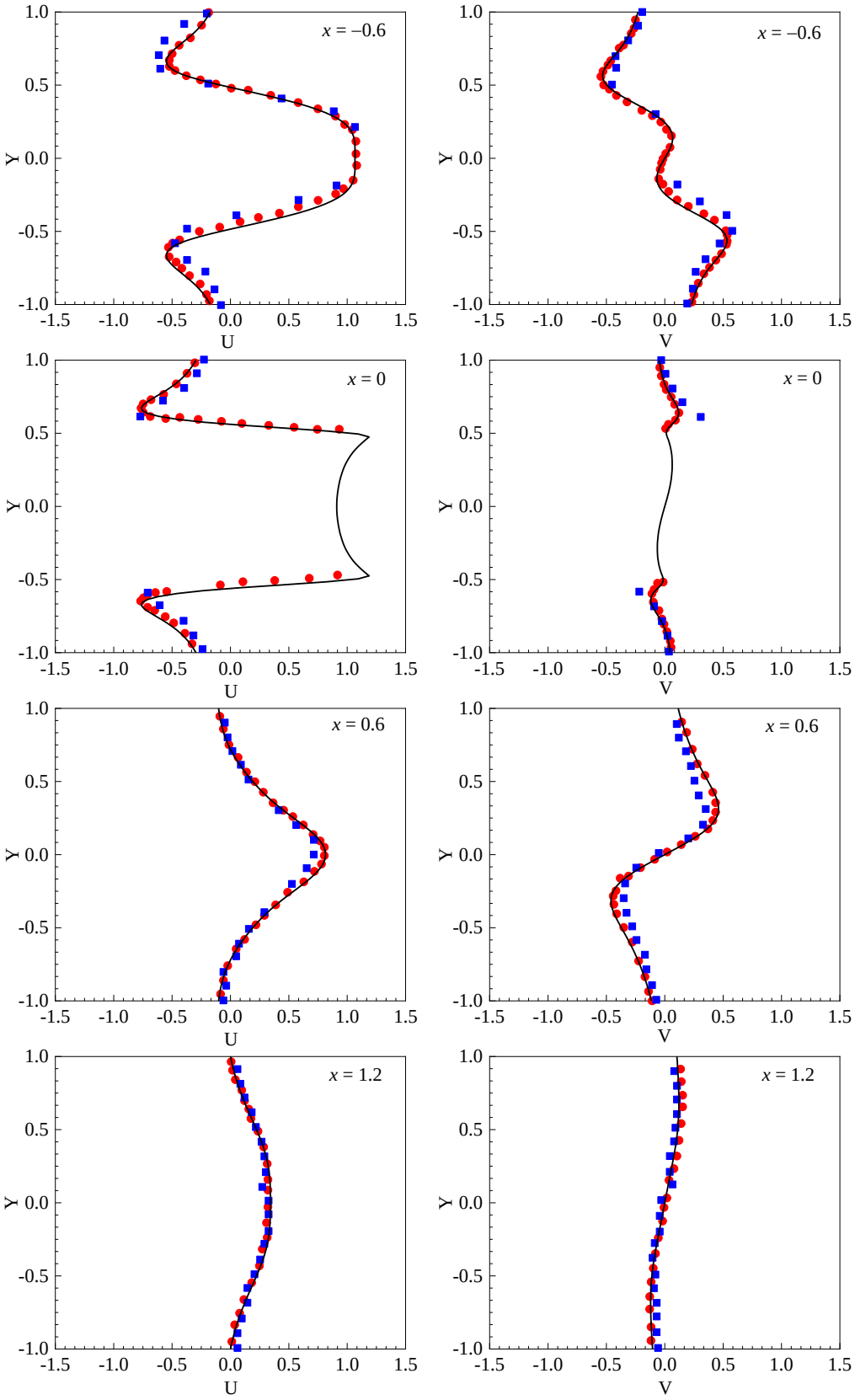


Fig. 26. Comparison of the x-direction velocity (U) and y-direction velocity (V) at different cross sections at phase position 180°. ■: experimental data [71]; ●: numerical results [71]; solid line: the present results. (Transition zone width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

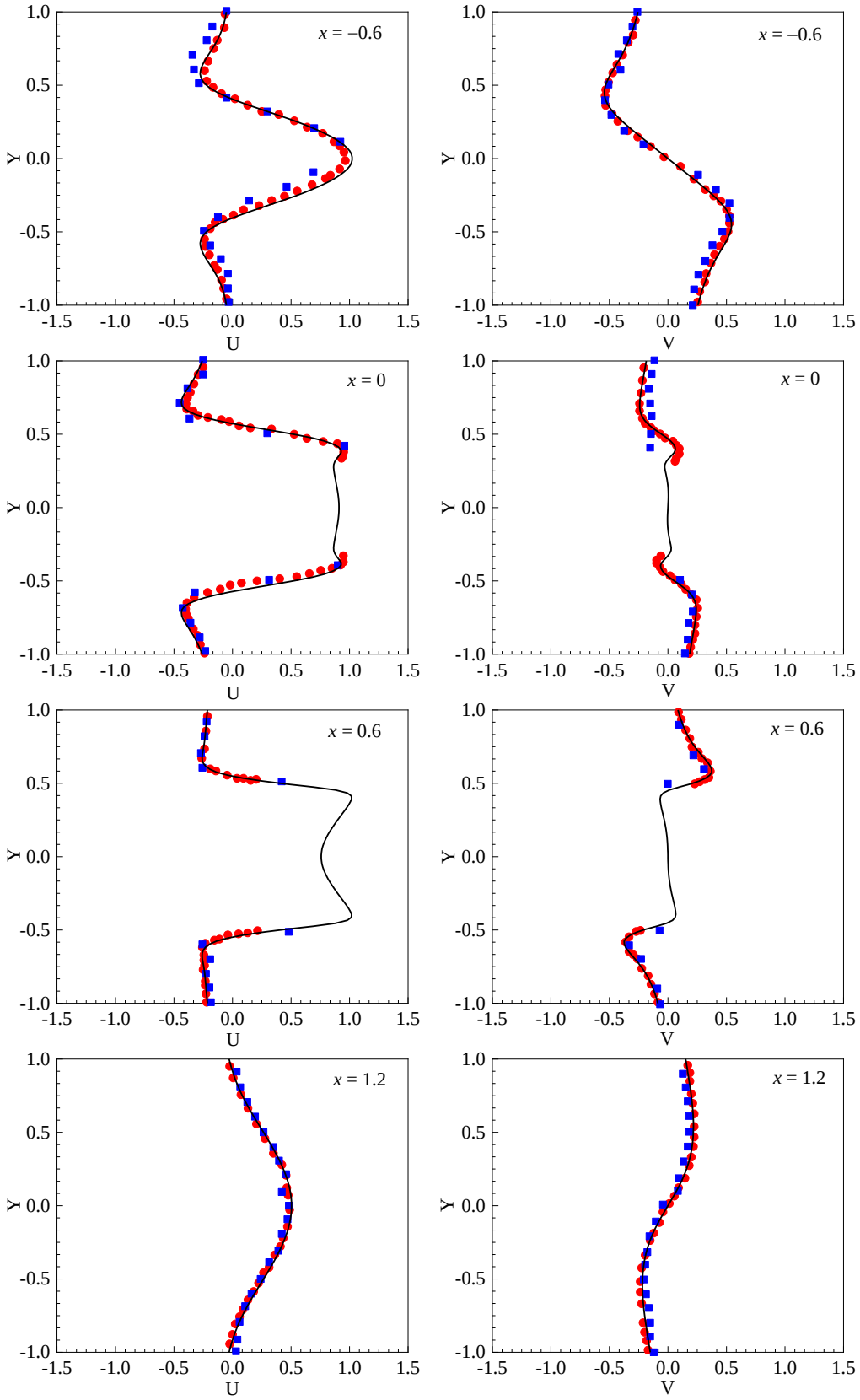


Fig. 27. Comparison of the x-direction velocity (U) and y-direction velocity (V) at different cross sections at phase position 210°. ■: experimental data [71]; ●: numerical results [71]; solid line: the present results. (Transition zone width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

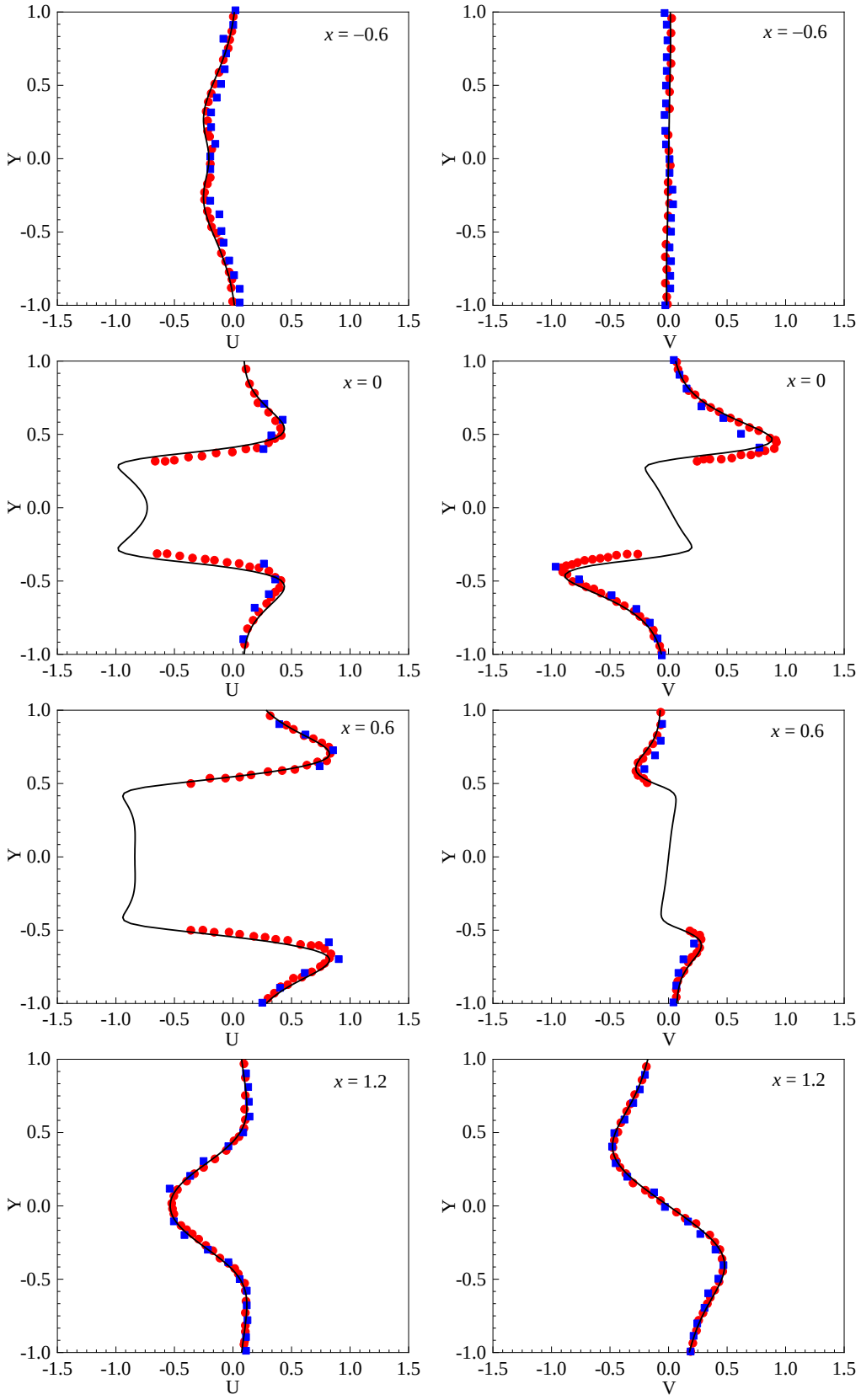


Fig. 28. Comparison of the x-direction velocity (U) and y-direction velocity (V) at different cross sections at phase position 330°. ■: experimental data [71]; ●: numerical results [71]; solid line: the present results. (Transition zone width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

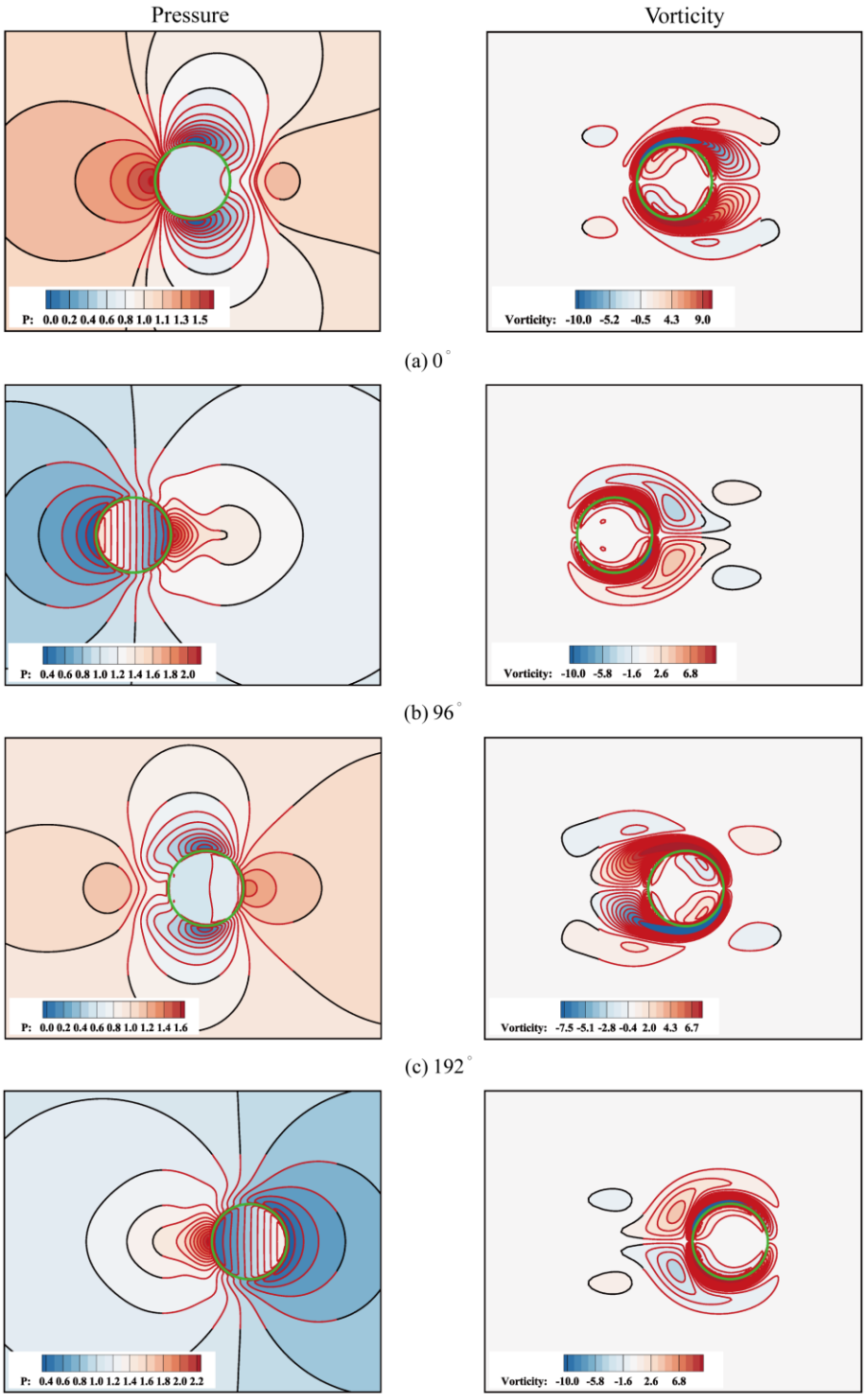


Fig. 29. Pressure and vorticity contours at four different phase positions for the present method solutions. (Black line: background domain solution, red line: sub-domain solution, green line: cylinder, transition zone width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

3.5 Transversely oscillating cylinder in a free-stream

To further validate the present method, a transversely oscillating cylinder in uniform free-stream is simulated. The oscillating function of the cylinder is $y(t) = A\cos(2\pi f_e t)$, where A is the amplitude of the oscillation and f_e is the excitation frequency. The Reynolds number (Re) is 185, and the oscillating amplitude A is $0.2D$. The excitation frequency range is $0.8 \leq f_e/f_0 \leq 1.2$, where $f_0 = 0.195$ is the vortex shedding frequency of a fixed cylinder at $Re = 185$. This simulation is carried out in two computational domains, wherein the background domain size is [0,30]×[0,20] and the sub-domain size is [8.85,11.15]×[8.85,11.15]. The boundary conditions and grid layout of the computational domain are shown in Fig. 30.

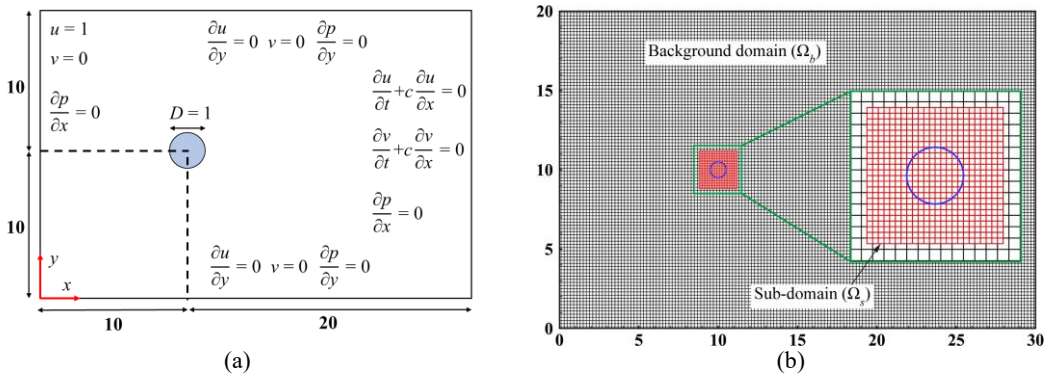


Fig. 30. Boundary condition and grid layout for transversely oscillating cylinder in a free-stream. (a) Boundary condition; (b) Grid layout with background domain (black line: Ω_b), sub-domain (red line: Ω_s), and cylinder (blue line).

The results of $f_e/f_0 = 0.8$ are summarized in Table 6, in which the average drag coefficient ($\overline{C}_D$), the root mean square value of the drag coefficient (C_D^{rms}) and the root mean square value of the lift coefficient (C_L^{rms}) are presented. All the three key quantities agree well with the reference data. Further comparison reveals that the root mean square value of the lift coefficient that calculated using the present method is more closely aligned with the body-fitted grid results of Guilmineau and Queutey [68]. The time histories of the drag and lift coefficients with six exciting frequencies ($f_e/f_0 = 0.8$, 0.9, 1.0, 1.1, 1.12 and 1.2) are presented in Fig. 31. As demonstrated, the well-known "knot phenomenon" for $f_e/f_0 = 1.1$ is also well captured by the present method.

Table 6

Comparison of the force coefficients for the transversely oscillating cylinder at $Re = 185$, $f_e/f_0 = 0.8$.

Source	Method	$\overline{C}_D$	C_D^{rms}	C_L^{rms}
Present	-	1.307	0.043	0.077
Yang et al. [35]	Diffuse-interface IB method	1.290	0.043	0.070
Guilmineau and Queutey [68]	Body-fitted grid	1.195	0.036	0.080
Yang and Balaras [34]	Sharp-interface IB method	1.266	0.042	0.070

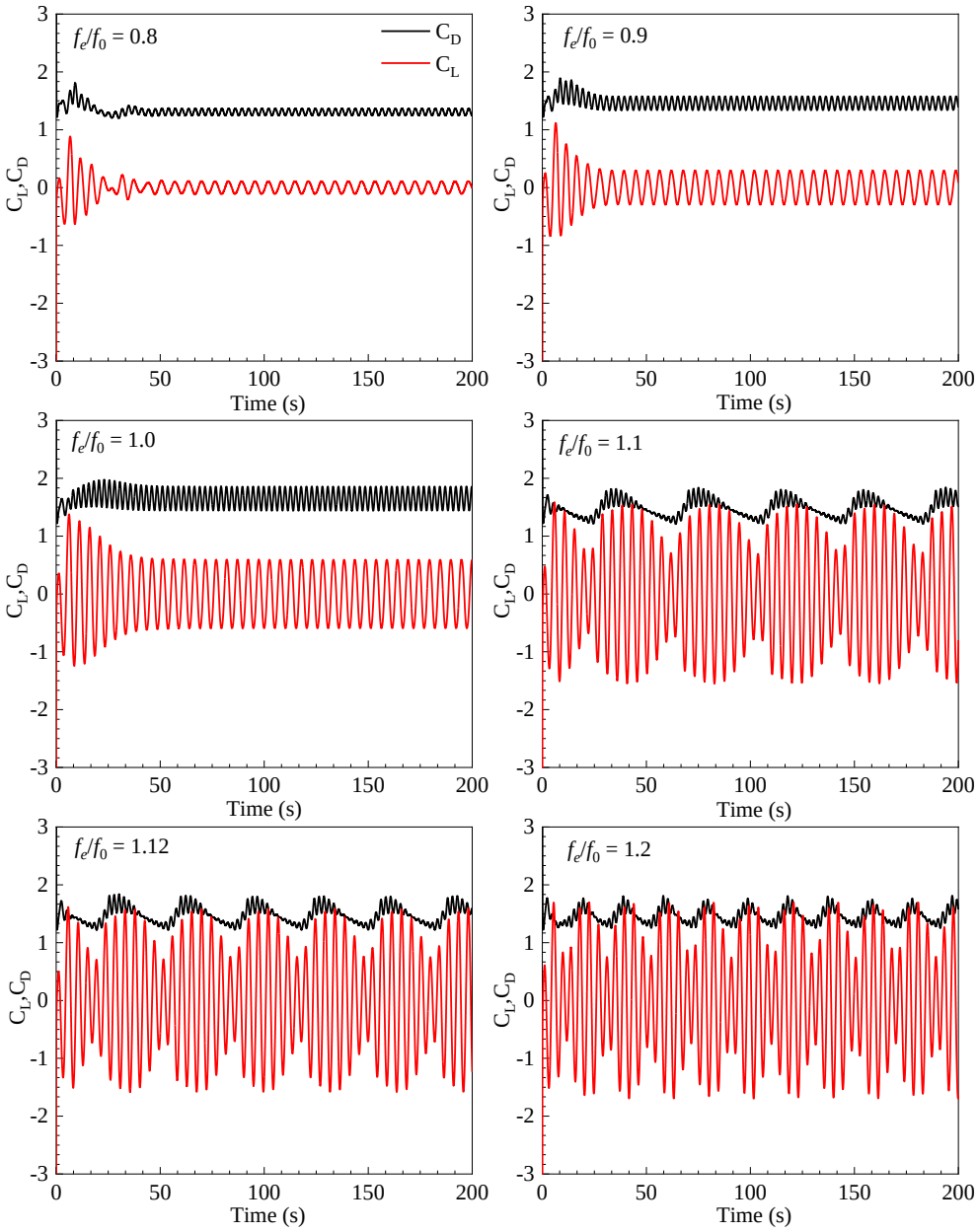


Fig. 31. Time histories of the lift (lower line) and drag (upper line) coefficients of a transversely oscillating cylinder with different excitation frequencies at $Re = 185$. (Transition zone width $(H) = 3\Delta_b$, CFL number $= 0.04$, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

The comparisons of the drag and lift coefficients for six frequencies with the different numerical results are shown in Fig. 32(a). The phase angle between the lift coefficient and the vertical displacement of the cylinder at different frequencies is shown in Fig. 32(b). The results are in good agreement with the corresponding results of Guilmineau [68] and Yang [34]. Fig. 33 shows the pressure and vorticity contours at three excitation frequencies when the cylinder is at the positive peak of the oscillation. It can be observed that the pressure and vorticity fields are continuous near the sub-domain boundary.

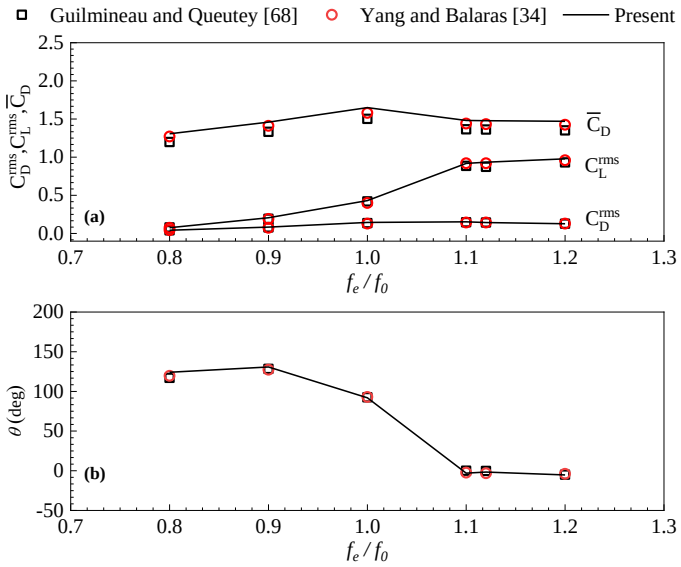


Fig. 32. Comparison of (a) the force coefficients and (b) the phase angle between lift coefficient and vertical displacement for a transversely oscillating cylinder with different excitation frequencies at $Re = 185$. (Transition zone width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

In the IB methods, a point within the body can become a point within the fluid at the next time step and vice versa, as the body boundary moves. This leads to the velocity and pressure at certain points in the flow acquiring non-physical values due to their previous location within the body boundary [34]. Therefore, for IB methods, non-physical pressure oscillations can be found when simulating moving boundaries. Fig. 34 compares the pressure obtained by the IB method and the present method at three monitoring points (see Fig. 35). Here, a uniform grid with $D/\Delta x = 40$ is used in the whole computational domain for the IB method. In the present method, a grid resolution of $D/\Delta x = 20$ is used for the background domain and $D/\Delta x = 40$ for the sub-domain to enhance computational efficiency. The results indicate that the IB method has non-physical pressure oscillations on the body surface for 2-point [72], 3-point [55], and 4-point [56] functions. In contrast, the present method effectively suppresses the non-physical pressure oscillations for all tested regularized delta functions. This is attributable to the body boundaries remaining stationary relative to the sub-domain during the simulation of moving bodies. In this way, the points within the body boundary do not become fluid points. As a result, the present method can effectively suppress non-physical pressure oscillations when simulating moving boundaries. This leads to the following benefits of the present method compared to the traditional IB method: (i) In the traditional diffuse-interface IB method, the 2-point function can achieve higher wall resolution. However, it leads to very significant non-physical pressure oscillations (see Fig. 34(a)). In addition, the 3-point and 4-point functions can achieve higher accuracy due to smaller non-physical pressure oscillations. Therefore, the 2-point function is rarely used. For the present method, the 2-point function can be used to impose the no-slip boundary condition in the sub-domain for achieving better wall resolution, since the non-physical pressure oscillations are effectively suppressed; (ii) The present method can effectively suppress non-physical pressure oscillations without requiring relatively fine grids. In three-dimensional simulations, it can save computational resources and improve computational efficiency; (iii) In traditional IB methods, non-physical pressure oscillations can also be suppressed with the non-uniform grids. However, in scenarios where bodies undergo large distance motion

induced by the flow (such as sedimenting particles), non-uniform grid approaches may become cumbersome, whereas the present method could be advantageous in these cases.

The above results show that the present method can accurately simulate moving bodies and ensure the continuity of physical quantities in the whole computational domain. Meanwhile, the present method can effectively suppress the non-physical pressure oscillations when the IB method is used to impose no-slip boundary conditions in the sub-domain.

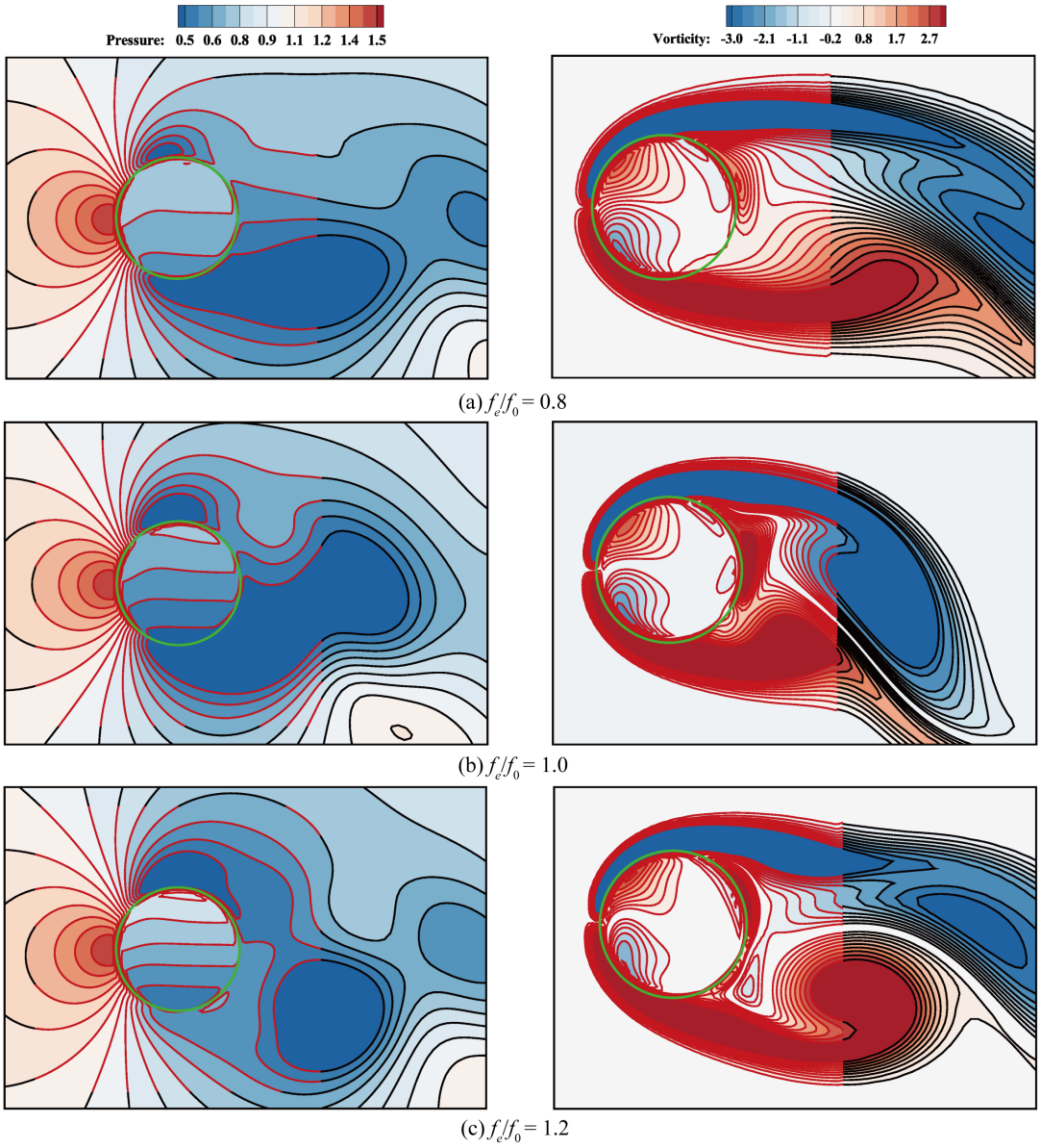


Fig. 33. Pressure and vorticity contours at three excitation frequencies. (Black line: background domain solution, red line: sub-domain solution, green line: cylinder, transition zone width $(H) = 3\Delta_b$, *CFL* number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

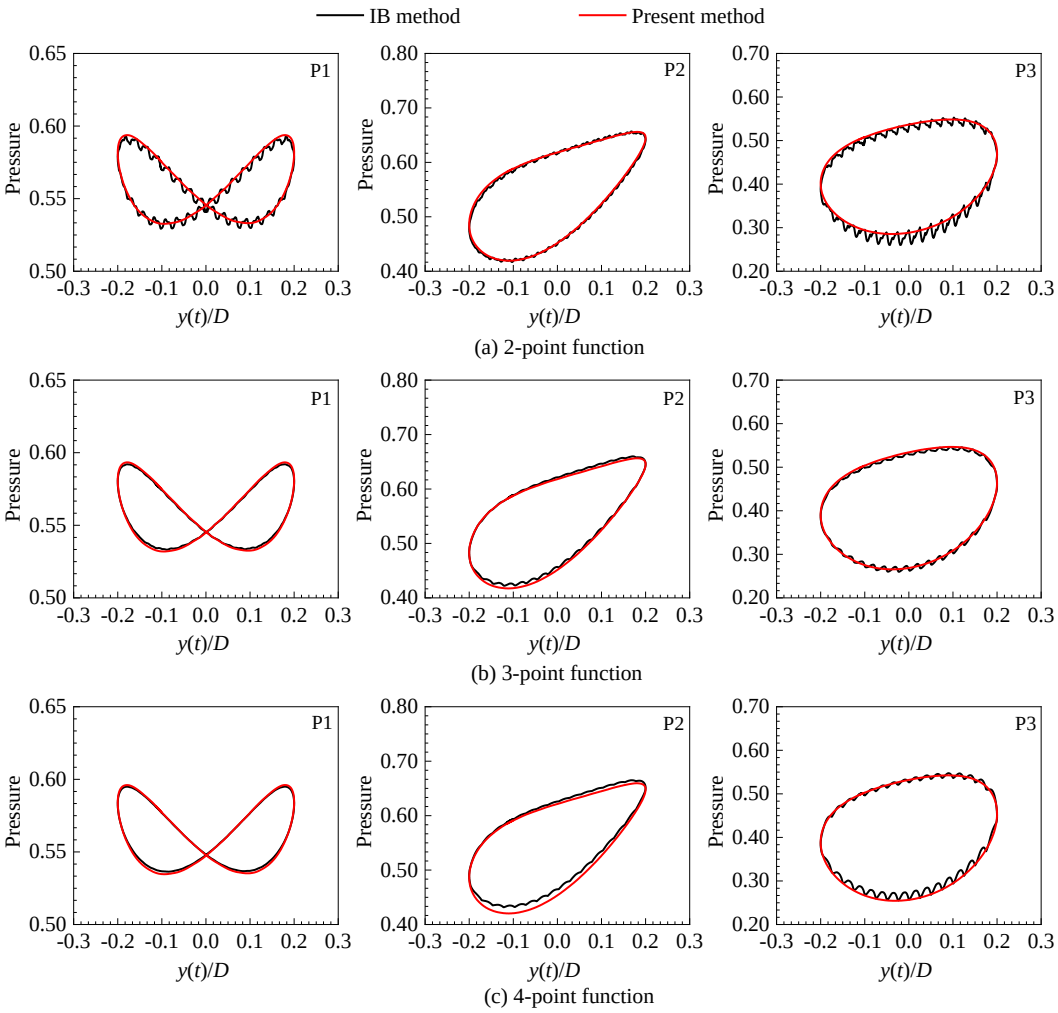


Fig. 34. Time-periodic variation of the pressure at the monitoring point of the body surface with different regularized delta functions. (Transition zone width (H) = $3\Delta_b$, CFL number = 0.04, δ_L/D = 1.0)

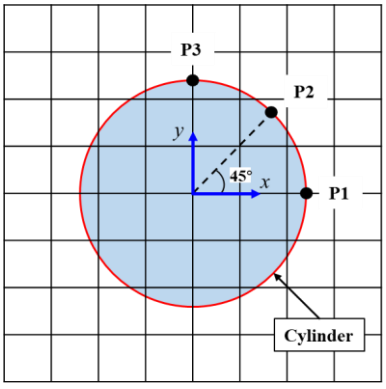


Fig. 35. Schematic diagram of pressure monitoring point location

3.6 Flow past two tandem cylinders

To assess the accuracy of the present method in simulating the interaction of multiple bodies, the flow past two tandem cylinders is computed. In this simulation, the upstream cylinder is

stationary, while the downstream cylinder is stationary or undergoes a transverse motion. For the two cases, a Reynolds number (Re) of 100 is used. The boundary conditions and grid layout are shown in Fig. 36.

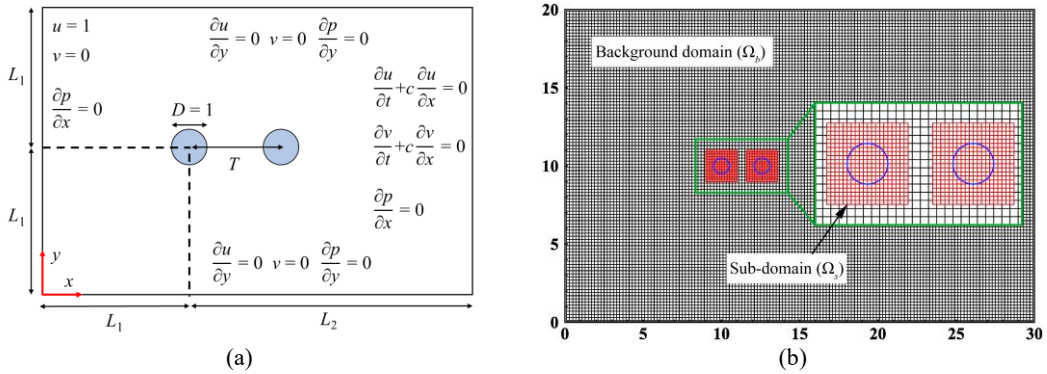


Fig. 36. Boundary condition and grid layout for flow past two tandem cylinders. (a) Boundary condition; (b) Grid layout with background domain (black line: Ω_b), sub-domain (red line: Ω_s), and cylinder (blue line).

For the stationary downstream cylinder, three center-to-center spacing distances are considered: $T/D = 2.5$, $T/D = 5.5$ and $T/D = 7.0$, where D is the cylinder diameter. In this simulation, the size of the background domain is $[0,30+T]\times[0,20]$. The corresponding values of L_1 and L_2 are 10 and $20+T$, respectively. The sizes of the sub-domains corresponding to the upstream cylinder and the downstream cylinder are $[8.85,11.15]\times[8.85,11.15]$ and $[8.85+T,11.15+T]\times[8.85,11.15]$, respectively. Table 7 presents the comparison of drag coefficient, lift coefficient, and Strouhal number between the present method and the reference data. It can be observed that the mean drag coefficient and the Strouhal number are close to the results of Ding et al [73]. The maximum error, which occurs for the downstream cylinder, is 3.96%. Regarding the peak value of the lift coefficient, it is close to the result of Mittal and Kumar [74], and the maximum error, also occurring for the downstream cylinder, is 6.38%. Overall, the present results are in good agreement with the reference data.

Table 7

Comparison of drag coefficient, lift coefficient, and Strouhal number for two stationary tandem cylinders at $Re = 100$.

		$T/D = 2.5$		$T/D = 5.5$	
		UC	DC	UC	DC
Drag coefficient	Present	1.202	−0.0859	1.378 ± 0.014	0.824 ± 0.131
	Ding et al. [73]	1.163	−0.0895	1.329 ± 0.013	0.858 ± 0.125
	Mittal and Kumar [74]	1.271	−0.075	1.433 ± 0.015	0.952 ± 0.164
Lift coefficient	Present	0.0	0.0	0.0 ± 0.395	0.0 ± 1.630
	Ding et al. [73]	0.0	0.0	0.0 ± 0.330	0.0 ± 1.554
	Mittal and Kumar [74]	0.0	0.0	0.0 ± 0.403	0.0 ± 1.741
Strouhal number	Present	-	-	0.158	0.158
	Ding et al. [73]	-	-	0.160	0.160
	Mittal and Kumar [74]	-	-	0.168	0.168

Note: UC refers to the upstream cylinder, DC refers to the downstream cylinder

The results for $Re = 100$ and $T/D = 7.0$ are summarized in Table 8, which presents the drag coefficient, lift coefficient, and Strouhal number. It can be seen that the maximum error in the drag coefficient, lift coefficient and Strouhal number are 1.33%, 1.93% and 3.61%, respectively. The results indicate that the present results are well consistent with the results of Mahir and Altaç [75]. Additionally, when investigating different center-to-center distances, a limitation of the present method was identified. Specifically, the two sub-domains must remain non-overlapping to prevent numerical divergence. In traditional overset grid methods, each computational domains are assigned a priority level, with higher-level domains overriding lower-level ones. The boundary values of higher-level domains are interpolated from the lower-level domains. In the present method, no priority level is assigned to the computational domains, and the boundary values for the two sub-domains are interpolated from the background domain. This may be the cause of the numerical divergence observed when the two sub-domains overlap. In future work, each computational domain will be assigned a priority level to overcome the current limitations.

Table 8

Comparison of drag coefficient, lift coefficient, and Strouhal number for two stationary tandem cylinders with $Re = 100$ and $T/D = 7.0$.

	Drag coefficient		Lift coefficient		Strouhal number	
	UC	DC	UC	DC	UC	DC
Present	1.373 ± 0.012	0.673 ± 0.07	± 0.356	± 1.319	0.160	0.160
Mahir and Altaç [75]	1.355 ± 0.014	0.682 ± 0.07	± 0.363	± 1.309	0.166	0.166
Error (%)	1.33	1.32	1.93	0.76	3.61	3.61

Note: UC refers to the upstream cylinder, DC refers to the downstream cylinder

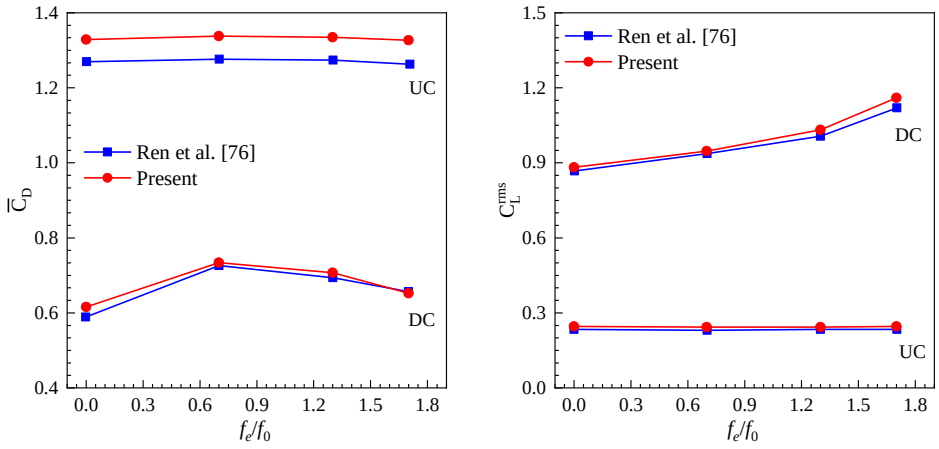


Fig. 37. Time-averaged drag coefficient ($\overline{C}_D$) and root mean square value of the lift coefficient (C_L^{rms}) versus vibration frequency at $T/D = 7.0$. (UC: upstream cylinder, DC: downstream cylinder, transition zone width (H) $= 3\Delta_b$, CFL number $= 0.04$, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

In the second case, the upstream cylinder is stationary, and the downstream cylinder undergoes a specified transverse motion. In the present method, the IB method is applied in the sub-domains. For validation purposes, the numerical results reported by Ren et al. [76] using the IB method serve as reference data for comparative analysis. In this simulation, the oscillating function of the cylinder

is $y(t) = A\sin(2\pi f_e t)$, where A is the amplitude of the oscillation and f_e is the excitation frequency. $T/D = 7.0$, $A = 0.15D$, and four excitation frequencies are used. The four excitation frequencies are $f_e/f_0 = 0.0$, $f_e/f_0 = 0.7$, $f_e/f_0 = 1.3$, and $f_e/f_0 = 1.7$. Here, $f_0 = 0.158$ is the vortex shedding frequency of the two tandem stationary cylinders at $T/D = 7.0$. The size of the background domain is [0, 45]×[0, 30]. The corresponding values of L_1 and L_2 are 15 and 30, respectively. The sizes of the sub-domains corresponding to the upstream cylinder and the downstream cylinder are [13.85, 16.15]×[13.85, 16.15] and [20.85, 23.15]×[13.85, 16.15], respectively.

The comparisons of the drag and lift coefficients for different frequencies with the numerical results of Ren et al. are shown in Fig. 37. It can be seen that the present results are in good agreement with the results of Ren et al. [76], except for the average drag coefficient of the upstream cylinder. For the upstream cylinder, the maximum error in the average drag coefficient is only 5.07%. In general, the present method shows a good agreement with the results obtained by Ren et al. [76]. Fig. 38 shows the pressure and vorticity contours at three excitation frequencies when the downstream cylinder is at the positive peak of the oscillation. It can be observed that the pressure and vorticity fields are continuous near the sub-domain boundaries. The above results demonstrate that the present method can accurately simulate the interactions of multiple bodies involving motion and ensure the continuity of physical quantities in the whole computational domain.

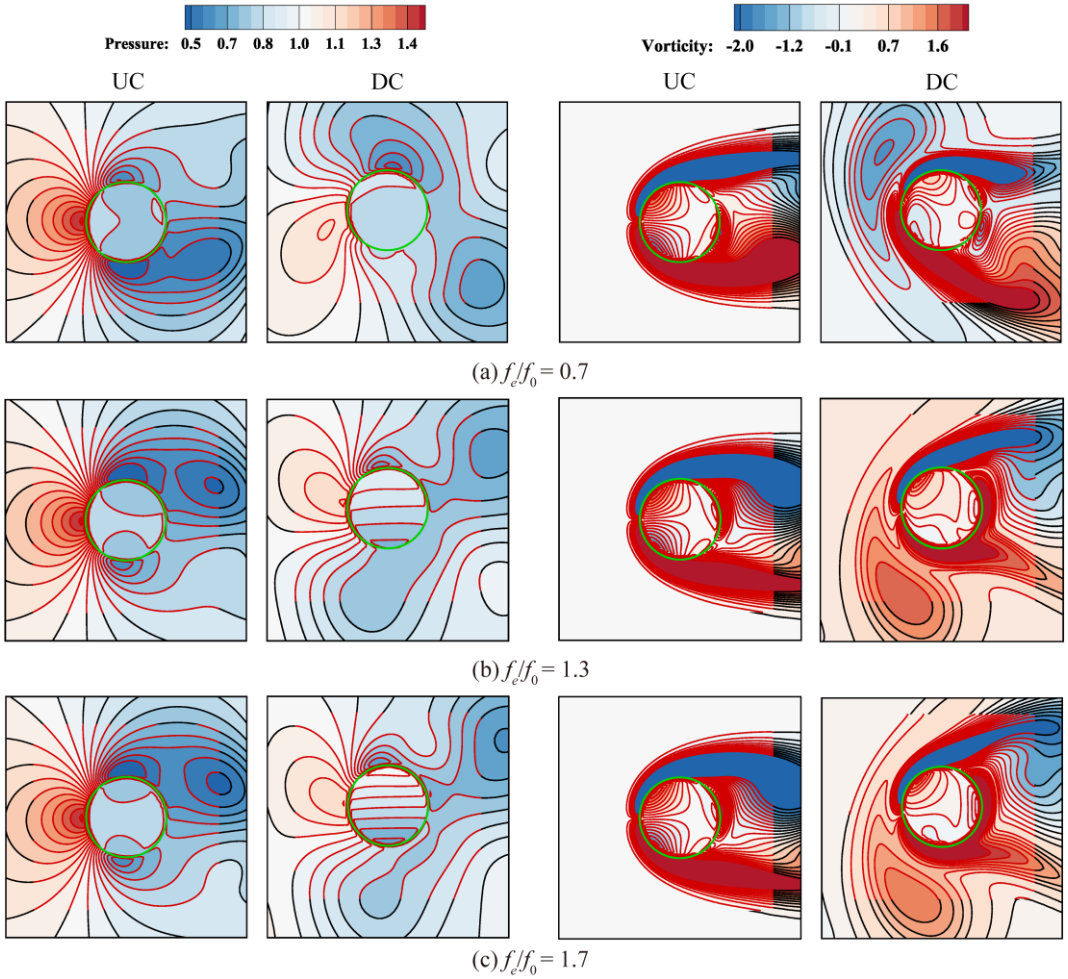


Fig. 38. Pressure and vorticity contours at three excitation frequencies. (UC: upstream cylinder, DC: downstream cylinder, black line: background domain solution, red line: sub-domain solution, green line: cylinder, transition zone

width (H) = $3\Delta_b$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

4. Conclusions

In this study, a novel overset grid assembly strategy based on the diffuse-interface IB method for a weak coupling framework is presented. The present method integrates the advantages of the overset grid method and the IB method while mitigating their limitations. The highlights of the present method include: (i) At each time step, the third-order polynomial time extrapolation technique is used to estimate the pressure at the sub-domain receptor points. The velocity at the sub-domain receptor points is determined using the estimated pressure and the intermediate velocity directly interpolated from the background domain. Thereby, the velocity and pressure boundary conditions of the sub-domains can be accurately imposed without the iterative process. Compared to traditional weak coupling overset grid methods, the iterative process between the background domain and the sub-domains is not necessary, which significantly enhances the computational efficiency; (ii) Two additional body force terms, obtained by the diffuse-interface IB method, are added to the momentum equations in the background domain. The first body force term implicitly imposes the no-slip condition on the body boundary in the background domain. The second body force term utilizes the information from the sub-domains to further correct the background flow field, thereby ensuring the continuity of the flow field between the background domain and the sub-domains. In this way, the explicit flow field information interpolation process from the sub-domains to the background domain is not necessary. Therefore, in contrast to traditional overset grid methods, the complex hole-cutting operation is avoided; (iii) There is non-physical pressure oscillations in IB methods when dealing with moving bodies. However, the present method can effectively suppress the non-physical pressure oscillations in moving body simulations; (iv) Compared to the HBM-IBM, there is no need to search for the background domain grid covered by the sub-domains, and only the flow field on the background domain grid near the body boundary and the artificial boundary (see Fig. 8) needs to be corrected by the diffuse-interface IB method. Thereby, it can save computational resources and improve computational efficiency, especially in three-dimensional simulations.

To validate the present method, a series of numerical simulations have been carried out. In the Taylor-Green vortex simulation, second-order spatial convergence accuracy for velocity is obtained. The lid-driven cavity flow is used to test the flow field information exchange process between the background domain and the sub-domains. In the simulation of fixed cylinder, several key parameters are discussed, such as transition zone width (H), sub-domain size and relative grid size (Δ_b/Δ_s), etc. In addition, the computational efficiency is analyzed by the fixed cylinder and the in-line oscillating cylinder simulation. The results show that for the fixed cylinder, the computational time of the present method only increases by about 10% compared to the single domain solution with the same grid size. For the moving cylinder, compared with the single-domain solution, the total computational cost of the present method only increases by 11.67%. In the in-line and transversely oscillating cylinder simulation, the computed results agree well with the reference data. The effectiveness of the present method in suppressing non-physical pressure oscillations is demonstrated by the simulation of a transversely oscillating cylinder. In addition, the flow past two tandem cylinders is used to assess the accuracy of the present method in simulating the interaction of multiple bodies.

In the overset grid method, body-fitted grids are typically used for sub-domains. This study

primarily focuses on data exchange between the background domain and the sub-domains. For simplicity, uniform Cartesian grids are used for sub-domains. In addition, the motions in FSI are also simplified from induced motions to specified motions. In future work, the novel overset grid assembly strategy combined with body-fitted sub-domains will be applied to simulate FSI involving induced motions and high Reynolds number flows. It should be noted that the explicit treatment of viscous terms and a low CFL number are not suitable for simulating high Reynolds number flows in this study. Thus, further refinements to the present method will be necessary in future research, such as the use of a semi-implicit treatment for viscous terms.

Acknowledgments

This work is financially supported by the Fundamental Research Funds for the Central Universities (grant number: 3072022FSC0101)

Appendix A. Regularized delta functions

Here we write the one-dimensional form of some of the regularized delta functions used in this work. In multiple dimensions, a tensor product of the one-dimensional regularized delta functions is formed.

A 2-point function [72]:

$$\phi_1(r)=\begin{cases}1-|r|, & |r|<1.0\\ 0, & 1.0\le|r|\end{cases} \tag{A.1}$$

A 3-point function [55]:

$$\phi_3(r)=\begin{cases}\frac{1}{3}\left(1+\sqrt{-3r^2+1}\right), & |r|<0.5\\ \frac{1}{6}\left(5-3|r|-\sqrt{-3(1-|r|)^2+1}\right), & 0.5\le|r|<1.5\\ 0, & 1.5\le|r|\end{cases} \tag{A.2}$$

A 4-point function [56]:

$$\phi_4(r)=\begin{cases}\frac{1}{8}\left(3-2|r|+\sqrt{1+4|r|-4r^2}\right), & |r|<1\\ \frac{1}{8}\left(5-2|r|-\sqrt{-7+12|r|-4r^2}\right), & 1\le|r|<2\\ 0, & 2\le|r|\end{cases} \tag{A.3}$$

A 6-point function [57]:

$$\phi_6(r)=\begin{cases}\frac{61}{112}-\frac{11}{42}|r|-\frac{11}{56}|r|^2+\frac{1}{12}|r|^3+\frac{\sqrt{3}}{336}(243+1584|r| & |r|<1\\ -748|r|^2-1560|r|^3+500|r|^4+336|r|^5-112|r|^6)^{\frac{1}{2}}, & \\ \frac{21}{16}+\frac{7}{12}|r|-\frac{7}{8}|r|^2+\frac{1}{16}|r|^3-\frac{3}{2}\phi_6(|r|-1), & 1\le|r|<2\\ \frac{9}{8}-\frac{23}{12}|r|+\frac{3}{4}|r|^2-\frac{1}{12}|r|^3+\frac{1}{2}\phi_6(|r|-2), & 2\le|r|<3\\ 0, & 3\le|r|\end{cases} \tag{A.4}$$

A 4-point piecewise cubic function [58]:

$$\phi_4^*(r)=\begin{cases}1-\dfrac{1}{2}|r|-|r|^2+\dfrac{1}{2}|r|^3, & |r|<1\\ 1-\dfrac{11}{6}|r|+|r|^2-\dfrac{1}{6}|r|^3, & 1\le|r|<2\\ 0, & 2\le|r|\end{cases} \tag{A.5}$$

A smoothed 3-point function [35]:

$$\phi_3'(r)=\begin{cases}\dfrac{17}{48}+\dfrac{\sqrt{3}\pi}{180}+\dfrac{|r|}{4}-\dfrac{r^2}{4}+\dfrac{1-2|r|}{16}\sqrt{-12r^2+12|r|+1}\\ \quad-\dfrac{\sqrt{3}}{12}\arcsin\left(\dfrac{\sqrt{3}}{2}(2|r|-1)\right), & |r|<1\\ \dfrac{55}{48}-\dfrac{\sqrt{3}\pi}{180}-\dfrac{13|r|}{12}+\dfrac{r^2}{4}+\dfrac{2|r|-3}{48}\sqrt{-12r^2+36|r|-23}\\ \quad+\dfrac{\sqrt{3}}{36}\arcsin\left(\dfrac{\sqrt{3}}{2}(2|r|-3)\right), & 1\le|r|<2\\ 0, & 2\le|r|\end{cases} \tag{A.6}$$

A smoothed 4-point function [35]:

$$\phi_4'(r)=\begin{cases}\dfrac{3}{8}+\dfrac{\pi}{32}-\dfrac{r^2}{4}, & |r|<0.5\\ \dfrac{1}{4}+\dfrac{1-|r|}{8}\sqrt{-2+8|r|-4r^2}-\dfrac{1}{8}\arcsin\left(\sqrt{2}(|r|-1)\right), & 0.5\le|r|<1.5\\ \dfrac{17}{16}-\dfrac{\pi}{64}-\dfrac{3|r|}{4}+\dfrac{r^2}{8}+\dfrac{|r|-2}{16}\sqrt{-14+16|r|-4r^2}\\ \quad+\dfrac{1}{16}\arcsin\left(\sqrt{2}(|r|-2)\right), & 1.5\le|r|<2.5\\ 0, & 2.5\le|r|\end{cases} \tag{A.7}$$

Appendix B. Effect of transition zone width on numerical stability

To facilitate description, the grid nodes used to impose sub-domain boundary conditions are designated as receptor points, while the background grid nodes providing interpolation information to the receptor points are referred to as donor points. The background domain corrected by the artificial boundary is designated as the correction zone, as shown in Fig. B1. If the donor points are located within the correction zone, the flow field information at the donor points becomes contaminated, providing error flow field data to the receptor points. As time stepping progresses, this ultimately leads to computational divergence. Therefore, the donor points must be located outside the correction zone to ensure numerical stability. From Fig. B1, it can be observed that the spatial position between the donor point and the correction zone can be controlled by adjusting the transition zone width. In addition, the correction zone size depending on the specific regularized delta function. This indicates that a minimum transition zone width must be determined for the specified regularized delta function to ensure numerical stability. In the Fig. B1, the donor points and correction zone are located on the background domain. Thus, the transition zone width must be determined based on the background domain grid size. In this study, the transition zone widths are set as integer multiples of the background domain grid size due to the coincidence of the artificial boundary with the sub-domain grid. For the present method, the minimum transition zone widths corresponding to different regularized delta functions are presented in Table B1.

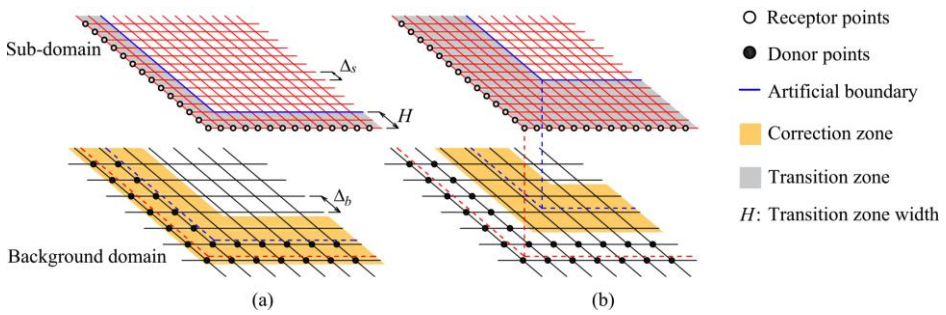


Fig. B1. Two-dimensional example of information exchange in the novel overset grid assembly strategy. (a) The donor point is located within the correction zone; (b) The donor point is located outside the correction zone.

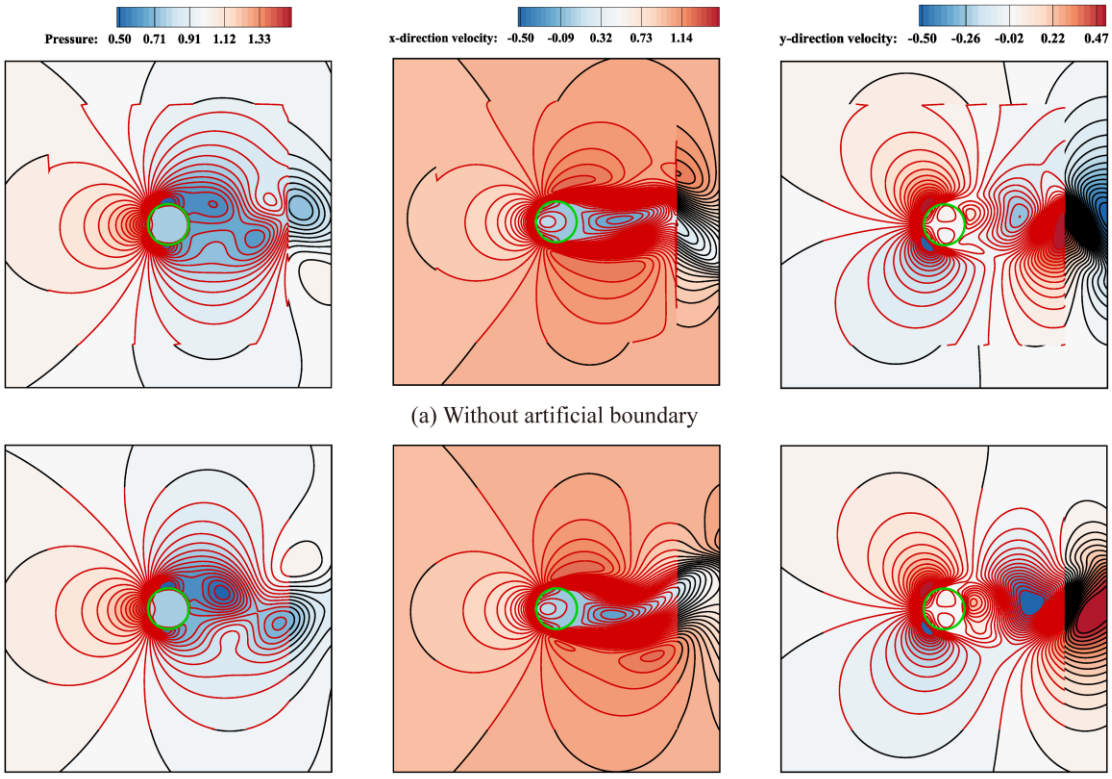
Table B1

The minimum transition zone width required to ensure the stability of the novel overset grid assembly strategy.

Regularized delta function	Support domain radius (r_s)	Minimum transition zone width (H)
3-point function	1.5	$3\Delta_b$
4-point function	2	$3\Delta_b$
6-point function	3	$4\Delta_b$
4-point piecewise cubic function	2	$3\Delta_b$
Smoothed 3-point function	2	$3\Delta_b$
Smoothed 4-point function	2.5	$4\Delta_b$

Appendix C. Effect of artificial boundary on flow field continuity

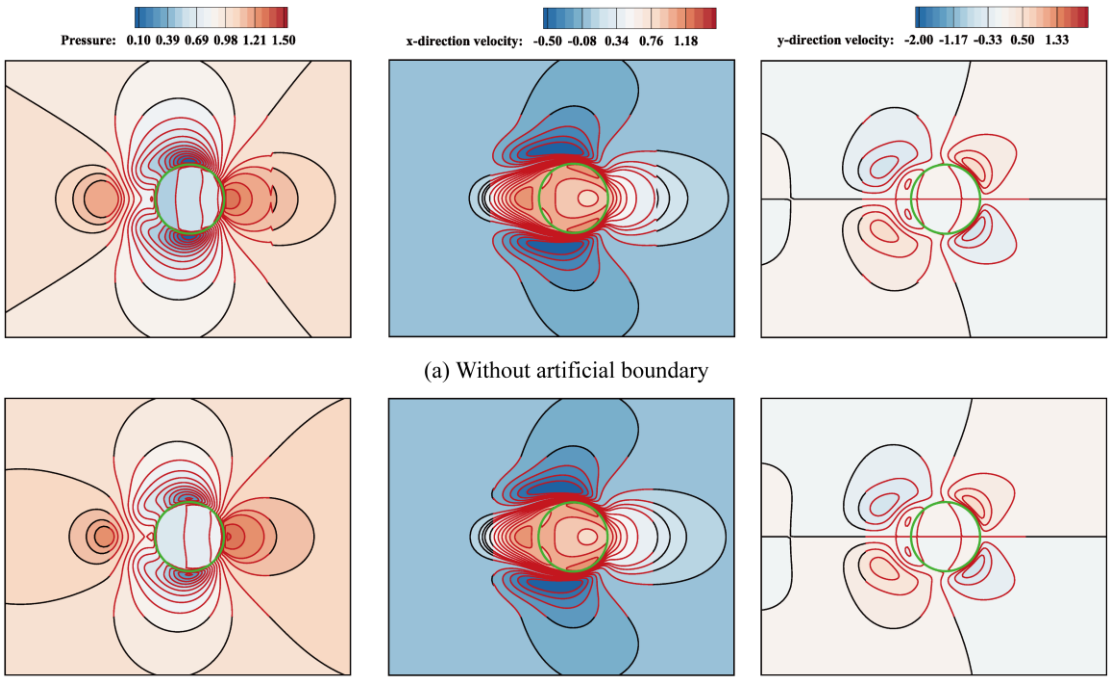
In the present method, in order to ensure the continuity of the flow field in the whole computational domain, the second body force term, calculated based on the velocity on the artificial boundary, is added to the momentum equations in the background domain. To more clearly demonstrate the impact of the artificial boundary on flow field continuity, two cases are presented: flow past a fixed circular cylinder and the in-line oscillation of a circular cylinder. From Fig. C1 and Fig. C2, it can be observed that for the stationary and moving circular cylinders, the absence of the artificial boundary will result in a discontinuous flow field. In contrast, the presence of the artificial boundary can ensure the continuity of the flow field in the whole computational domain.



(a) Without artificial boundary

(b) With artificial boundary

Fig. C1. Pressure and velocity contours at 200 s for a fixed circular cylinder. (Black line: background domain solution, red line: sub-domain solution, green line: cylinder, transition zone width (H) = $3\Delta_b$, Re = 100, CFL number = 0.04, δ_L/D = 3.0, D/Δ_b = 20, D/Δ_s = 40)



(a) Without artificial boundary

(b) With artificial boundary

Fig. C2. Pressure and velocity contours at 192° phase positions for the in-line oscillation of a circular cylinder. (Black line: background domain solution, red line: sub-domain solution, green line: cylinder, transition zone width (H) = $3\Delta_b$, CFL number = 0.04, δ_L/D = 1.0, D/Δ_b = 20, D/Δ_s = 40)

Appendix D. The conservation properties

In the IB methods, the lack of mass conservation has been identified as one of the causes for generating spurious force oscillations in case of moving body simulations [62]. In the present method, two additional body force terms, obtained by the diffuse-interface IB method, are considered into the momentum equations for solving the background domain. Thus, it is necessary to discuss the mass conservation properties of the present method.

Here, flow past a fixed circular cylinder and a transversely oscillating cylinder in a free-stream are used to examine the efficacy of the present method in mass conservation. The simulations are carried out in two computational domains, where the background domain size is [0,30]×[0,20] and the sub-domain size is [8.85,11.15]×[8.85,11.15]. For the moving cylinder, the oscillating function of the cylinder is $y(t) = A\cos(2\pi f_e t)$, where A is the amplitude of the oscillation and f_e is the excitation frequency. The oscillating amplitude A is $0.2D$. The excitation frequency is $f_e/f_0 = 0.8$, where $f_0 = 0.195$ is the vortex shedding frequency of a fixed cylinder at $Re = 185$.

The temporal history of the local maximum divergence $|\nabla\cdot\mathbf{u}|$ in the background domain and the sub-domain for the fixed cylinder is presented in Fig. D1. It can be seen that the velocity field satisfies the divergence-free condition in both the background domain and the sub-domain, with the maximum value of the local velocity divergence being 10^{-5}. This proves that the present method satisfies the discrete local mass conservation. For the moving cylinder, Fig. D2 depicts the temporal evolution of the local maximum divergence in both the background domain and the sub-domain, which remains bounded and less than 10^{-4} over the course of simulation. This further confirms the efficacy of the proposed method in satisfying the discrete mass conservation constraint even in simulations of moving bodies.

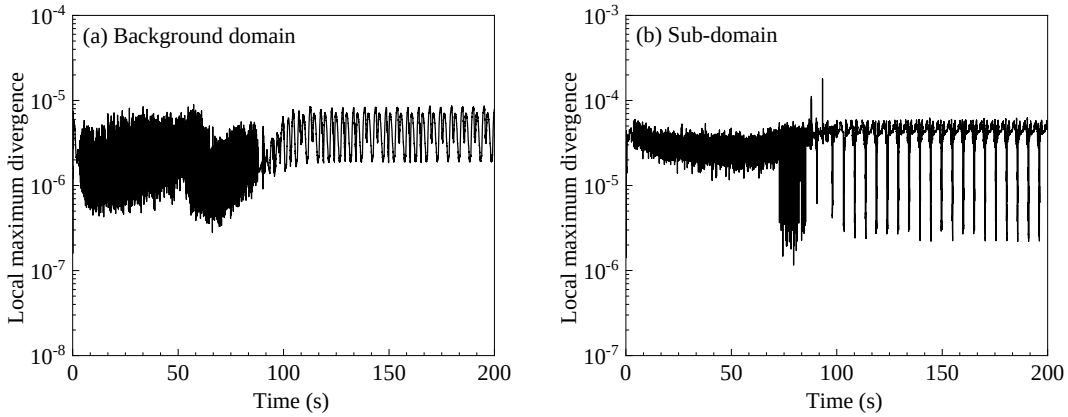


Fig. D1. Local maximum divergence in the background domain and sub-domain for the flow past a fixed circular cylinder. (Transition zone width (H) = $3\Delta_b$, $Re = 185$, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20$, $D/\Delta_s = 40$)

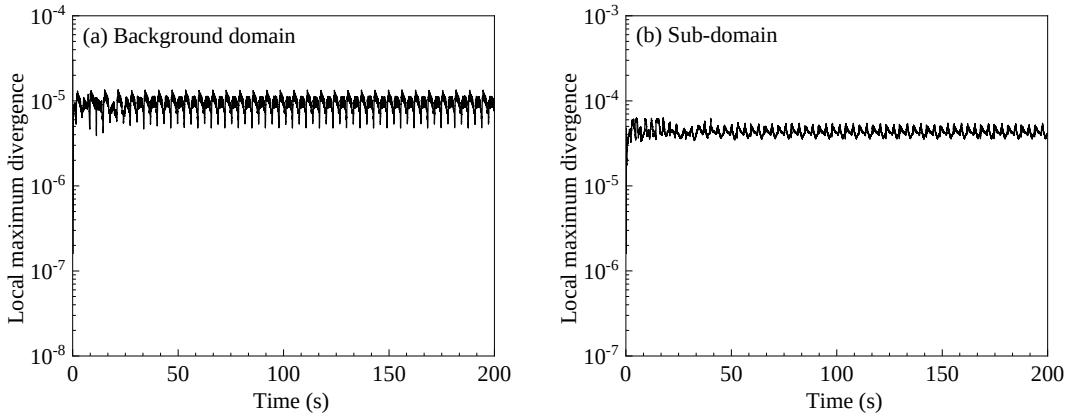


Fig. D2. Local maximum divergence in the background domain and sub-domain for the transversely oscillating cylinder in a free-stream. (Transition zone width (H) = $3\Delta_b$, Re = 185, CFL number = 0.04, $\delta_L/D = 1.0$, $D/\Delta_b = 20, D/\Delta_s = 40$)

References

[1] S. Aram, P. Mucha, Computational fluid dynamics analysis of different propeller models for a ship maneuvering in calm water, Ocean. Eng. 276 (2023) 114226, https://doi.org/10.1016/j.oceaneng.2023.114226.

[2] S.F. Fu, Z. Li, W.J. Zhu, X.X. Han, X.L. Liang, H. Yang, W.Z. Shen, Study on aerodynamic performance and wake characteristics of a floating offshore wind turbine under pitch motion, Renew. Energ. 205 (2023) 317-325, https://doi.org/10.1016/j.renene.2023.01.040.

[3] M. Daghooghi, I. Borazjani, The hydrodynamic advantages of synchronized swimming in a rectangular pattern, Bioinspir. Biomim. 10(5) (2015) 056018, http://doi:10.1088/1748-3190/10/5/056018.

[4] J.T. Batina, Unsteady Euler algorithm with unstructured dynamic mesh for complex aircraft aerodynamic analysis, AIAA J. 29 (1991) 327-333, https://doi.org/10.2514/3.10583.

[5] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10(2) (1972) 252-271, https://doi.org/10.1016/0021-9991(72)90065-4.

[6] J.L. Steger, F.C. Dougherty, J.A. Benek, A chimera grid scheme, in: K.N. Ghia, U. Ghia (Eds.), Advances in Grid Generation, ASME FED-vol. 5, June 1983, https://ntrs.nasa.gov/citations/19840028795.

[7] https://mdx.plm.automation.siemens.com/star-ccm-plus.

[8] http://www.ansys.com/Products/Fluids/ANSYS-Fluent.

[9] https://www.openfoam.com/.

[10] D. Chandar, Development of a Parallel Overset Grid Framework for Moving Body Simulations in Open FOAM, J. Appl. Math. Comput. 9(2) (2015) 22-30, https://doi.org/10.4316/JACSM.201502004

[11] L. Peng, G. Zhenxun, L. Ganlin, J. Chongwen, L. Chun-Hian, Overset structured grids assembly method for numerical simulations of multi-bodies and moving objects, Comput. Fluids 175 (2018) 260-275, https://doi.org/10.1016/j.compfluid.2018.08.009.

[12] S.L. Tian, K. Xu, J. Xia, L. Chen, Efficient parallel dynamic overset unstructured grid for flow simulation with moving bodies, AIAA J. 61(5) (2023), https://doi.org/10.2514/1.J062579.

[13] A. Sharmaa, S. Ananthana, J. Sitaramanb, S. Thomasa, M. A. Spraguea, Overset meshes for incompressible flows: On preserving accuracy of underlying discretizations, J. Comput. Phys. 428(1) (2021) 109987, https://doi.org/10.1016/j.jcp.2020.109987.

[14] I.T. Chiu, R. Meakin, On automating domain connectivity for overlapping grids, in: 33rd Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 1995, http://doi:10.2514/6.1995-854.

[15] J.A. Benek, J.L. Steger, F.C. Dougherty, A flexible grid embedding technique with application to the Euler equations, in: 6th Computational Fluid Dynamics Conference Danvers, Danvers, MA, USA, 1983, https://doi.org/10.2514/6.1983-1944.

[16] R. Meakin, Object X-rays for cutting holes in composite overset structured grids, in: 15th AIAA Computational Fluid Dynamics Conference, Anaheim, CA, USA, 2001,

http://doi:10.2514/6.2001-2537.
[17] T.H. Xiao, H.L. Zhi, S.H. Deng, Z.L. Chen, X.Y. Li, Enhancement on parallel unstructured overset grid method for complex aerospace engineering applications, Chinese Journal of Aeronautics 36(1) (2023) 115-138, https://doi.org/10.1016/j.cja.2022.07.015.
[18] K. Nakahashi, F. Togashi, D. Sharov, Intergrid-boundary definition method for overset unstructured grid approach, AIAA J. 38(11) (2000) 2077-2084, https://doi.org/10.2514/2.869.
[19] Y. Lee, J. Baeder, High-order overset method for blade vortex interaction, in: 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 2002, https://doi:10.2514/6.2002-559.
[20] S.E. Rogers, N.E. Suhs, W.E. Dietz, PEGASUS 5: An automated preprocessor for overset-grid computational fluid dynamics, AIAA J. 41(6) (2003) 1037-1045, https://doi.org/10.2514/2.2070.
[21] H. Fu, J. Xia, S.L. Tian, A new overset grid assembly strategy for dynamic grid systems, Math. Comput. Simul. 211 (2023) 171-191, https://doi.org/10.1016/j.matcom.2023.04.002.
[22] J.S. Oh, J.S. Rhee, T.Y. Kung, K.H. Kim, Exact and efficient wall distance pre-classification overset grid assembly for unstructured grids, J. Comput. Phys. 511 (2024) 113126, https://doi.org/10.1016/j.jcp.2024.113126.
[23] M.H. Carpenter, D. Gottlieb, S. Abarbanel, Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes, J. Comput. Phys. 111(2) (1994) 220-236, https://doi.org/10.1006/jcph.1994.1057
[24] N. Sharan, C. Pantano, D.J. Bodony, Time-stable overset grid method for hyperbolic problems using summation-by-parts operators, J. Comput. Phys. 361 (2018) 199-230, https://doi.org/10.1016/j.jcp.2018.01.049.
[25] H.S. Tang, S.C. Jones, F. Sotiropoulos, An overset-grid method for 3D unsteady incompressible flows, J. Comput. Phys. 191(2) (2003) 567-600, https://doi.org/10.1016/S0021-9991(03)00331-0.
[26] H. Hadzic, Development and application of finite volume method for the computation of flows around moving bodies on unstructured, overlapping grids, Ph.D. thesis, Hamburg University of Technology 2006, https://doi.org/10.15480/882.231.
[27] A.W. Vreman, Immersed boundary and overset grid methods assessed for Stokes flow due to an oscillating sphere, J. Comput. Phys. 423 (2020) 109783, https://doi.org/10.1016/j.jcp.2020.109783.
[28] D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip flow boundary with an external force field, J. Comput. Phys. 105 (1993) 354-366, https://doi.org/10.1006/jcph.1993.1081.
[29] J. Mohd-Yusof, Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries, in: Annual Research Briefs, Center for Turbulence Research, Stanford University, Stanford, CA, 1997, pp. 317-327.
[30] E. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 35-60, https://doi.org/10.1006/jcph.2000.6484.
[31] R. Verzicco, Immersed boundary methods: historical perspective and future outlook, Annu. Rev. Fluid Mech. 55(1) (2023) 129-155, https://doi.org/10.1146/annurev-fluid-120720-022129.
[32] S. Péron, C. Benoit, V. Gleize, I. Mary, M. Terracol, A mixed overset grid/immersed boundary

approach for CFD simulations of complex geometries, in: 54th AIAA Aerospace Sciences Meeting, San Diego, California, USA, 2016, https://doi:10.2514/6.2016-2055.
[33] A. Posa, R. Broglia, An immersed boundary method coupled with a dynamic overlapping-grids strategy, Comput. Fluids 191 (2019) 104250, https://doi.org/10.1016/j.compfluid.2019.104250.
[34] J.M. Yang, E. Balaras, An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, J. Comput. Phys. 215(1) (2006) 12-40, https://doi.org/10.1016/j.jcp.2005.10.035.
[35] X.L. Yang, X. Zhang, Z.L. Li, G.W. He, A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations, J. Comput. Phys. 228(20) (2009) 7821-7836, https://doi.org/10.1016/j.jcp.2009.07.023.
[36] C. Liu, C.H. Hu, An efficient immersed boundary treatment for complex moving object, J. Comput. Phys. 274 (2014) 654-680, https://doi.org/10.1016/j.jcp.2014.06.042.
[37] P.H. Chiu, cDFIB: A convolutional direct forcing immersed boundary method for solving incompressible flows with time-varying geometries, J. Comput. Phys. 487 (2023) 112178, https://doi.org/10.1016/j.jcp.2023.112178.
[38] H. Yu, C. Pantano, A regularized projection immersed boundary method for smooth boundary forces, J. Comput. Phys. 496 (2024) 112571, https://doi.org/10.1016/j.jcp.2023.112571.
[39] D. Xu, O. Colomés, A. Main, K. Li, N.M. Atallah, N. Abboud, G. Scovazzi, A weighted shifted boundary method for immersed moving boundary simulations of Stokes' flow, J. Comput. Phys. 510 (2024) 113095, https://doi.org/10.1016/j.jcp.2024.113095.
[40] S.G. Cai, J. Degrigny, J.F. Boussuge, P. Sagaut, Coupling of turbulence wall models and immersed boundaries on Cartesian grids, J. Comput. Phys. 429 (2021) 109995, https://doi.org/10.1016/j.jcp.2020.109995.
[41] W. Zhang, Y. Pan, J. Wang, V. Di Santo, G.V. Lauder, H. Dong, An efficient tree-topological local mesh refinement on Cartesian grids for multiple moving objects in incompressible flow, J. Comput. Phys. 479 (2023) 111983, https://doi.org/10.1016/j.jcp.2023.111983.
[42] M. Natarajan, R. Grout, W. Zhang, M. Day, A moving embedded boundary approach for the compressible Navier-Stokes equations in a block-structured adaptive refinement framework, J. Comput. Phys. 465 (2022) 111315, https://doi.org/10.1016/j.jcp.2022.111315.
[43] X. Zhao, L.M. Yang, C. Xu, An overset boundary condition-enforced immersed boundary method for incompressible flows with large moving boundary domains, Phys. Fluids 34 (2022) 103613, https://doi.org/10.1063/5.0122257.
[44] I. Borazjani, L. Ge, T. Le, F. Sotiropoulos, A parallel overset-curvilinear-immersed boundary framework for simulating complex 3D incompressible flows, Comput. Fluids 77 (2013) 76-96, http://dx.doi.org/10.1016/j.compfluid.2013.02.017.
[45] M. Hedayat, A.M. Akbarzadeh, I. Borazjani, A parallel dynamic overset grid framework for immersed boundary methods, Comput. Fluids 239 (2022) 105378, https://doi.org/10.1016/j.compfluid.2022.105378.
[46] J.Y. Wang, P. Shan, H.T. Zhu, Immersed boundary method with moving half-domain body-fitted meshes for 3-D compressible viscous flow, in: Proceedings of Global Power and Propulsion Society, Beijing, China, 2019, https://doi:10.33737/gpps19-bj-121.
[47] J.M. Yang, F. Stern, A sharp interface direct forcing immersed boundary approach for fully resolved simulations of particulate flows, J. Fluids Eng. 136(4) (2014) 040904, https://doi.org/10.1115/1.4026198.

[48] H. Choi, P. Moin, Effects of the computational time step on numerical solutions of turbulent flow, J. Comput. Phys. 113(1) (1994) 1-4, https://doi.org/10.1006/jcph.1994.1112.
[49] C. Ji, A. Munjiza, J.J.R. Williams, A novel iterative direct-forcing immersed boundary method and its finite volume applications, J. Comput. Phys. 231 (2012) 1797-1821, https://doi.org/10.1016/j.jcp.2011.11.010.
[50] M.M. Rajamuni, M.C. Thompson, K. Hourigan, Efficient FSI solvers for multiple-degrees-offreedom flow-induced vibration of a rigid body, Comput. Fluids 196 (2020) 104340, https://doi.org/10.1016/j.compfluid.2019.104340.
[51] G.F. Yao, Immersed Boundary Method for CFD: Focusing on its Implementation, second ed., CreateSpace Independent Publishing Platform, 2018.
[52] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, J. Comput. Phys. 171(1) (2001) 132-150, https://doi.org/10.1006/jcph.2001.6778.
[53] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209(2) (2005) 448-76, https://doi.org/10.1016/j.jcp.2005.03.017.
[54] T. Kempe, J. Fröhlich, An improved immersed boundary method with direct forcing for the simulation of particle laden flows, J. Comput. Phys. 231(9) (2012) 3663-3684, https://doi.org/10.1016/j.jcp.2012.01.021.
[55] A.M. Roma, C.S. Peskin, M.J. Berger, An adaptive version of the immersed boundary method, J. Comput. Phys. 153 (1999) 509-534, https://doi.org/10.1006/jcph.1999.6293.
[56] C.S. Peskin, The immersed boundary method, Acta Numerica 11 (2002) 479-517, https://doi:10.1017/S0962492902000077.
[57] J.M. Stockie, Analysis and computation of immersed boundaries, with application to pulp fibres, Ph.D. thesis, Institute of Applied Mathematics, University of British Columbia, 1997, https://doi:10.14288/1.0080530.
[58] A.K. Tornberg, B. Engquist, Numerical approximations of singular source terms in differential equations, J. Comput. Phys. 200(2) (2004) 462-488, https://doi.org/10.1016/j.jcp.2004.04.011.
[59] X.Y. Yu, D.R. Fuhrman, Y.L. Shao, K.P. Liao, W.Y. Duan, Y.X. Zhang, Enhanced solution of 2D incompressible Navier–Stokes equations based on an immersed-boundary generalized harmonic polynomial cell method, Eur. J. Mech. B-Fluid. 89 (2021) 29-44, https://doi.org/10.1016/j.euromechflu.2021.04.009.
[60] K. Taira, T. Colonius, The immersed boundary method: a projection approach, J. Comput. Phys. 225 (2) (2007) 2118-2137, https://doi:10.1016/j.jcp.2007.03.005.
[61] D.D.J. Chandar, On overset interpolation strategies and conservation on unstructured grids in OpenFOAM, Comput. Phys. Commun. 239 (2019) 72-83, https://doi.org/10.1016/j.cpc.2019.01.009.
[62] J.K. Patel, G. Natarajan, Diffuse interface immersed boundary method for multi-fluid flows with arbitrarily moving rigid bodies, J. Comput. Phys. 360 (2018) 202-228, https://doi:10.1016/j.jcp.2018.01.024.
[63] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387-411, https://doi.org/10.1016/0021-9991(82)90058-4.
[64] M. Uhlmann, First experiments with the simulation of particulate flows, Technical Report No.

1020, CIEMAT, Madrid, Spain, ISSN 1135-9420, 2003.
[65] L.W. Shen, E.S. Chan, P.Z. Lin, Calculation of hydrodynamic forces acting on a submerged moving object using immersed boundary method, Comput. Fluids 38(3) (2009) 691-702, https://doi.org/10.1016/j.compfluid.2008.07.002.
[66] C.H.K. Williamson, Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low Reynolds numbers, J. Fluid Mech. 206 (1989) 579-627, https://doi.org/10.1017/S0022112089002429.
[67] M.C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160(2) (2000) 705-719, https://doi.org/10.1006/jcph.2000.6483.
[68] E. Guilmineau, P. Queutey, A numerical simulation of vortex shedding from an oscillating circular cylinder, J. Fluids Struct. 16(6) (2002) 773-794, https://doi.org/10.1006/jfls.2002.0449.
[69] X.Y. Lu, C. Dalton, Calculation of the timing of vortex formation from an oscillating cylinder, J. Fluids Struct. 10(5) (1996) 527-541, https://doi.org/10.1006/jfls.1996.0035.
[70] A. Gronskis, G. Artana, A simple and efficient direct forcing immersed boundary method combined with a high order compact scheme for simulating flows with moving rigid boundaries, Comput. Fluids 124 (2016) 86-104, https://doi.org/10.1016/j.compfluid.2015.10.016.
[71] H. Dutsch, F. Durst, S. Becker, H. Lienhart, Low-Reynolds-number flow around an oscillating circular cylinder at low Keulegan-Carpenter numbers, J. Fluid Mech. 360 (1998) 249-271, https://doi.org/10.1017/S002211209800860X.
[72] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019–1044, https://doi.org/10.1137/0731054.
[73] H. Ding, C. Shu, K.S. Yeo, D. Xu, Numerical simulation of flows around two circular cylinders by mesh-free least square-based finite difference methods, Int. J. Numer. Meth. Fluids 53 (2007) 305-332, https://doi.org/10.1002/fld.1281.
[74] S. Mittal, V. Kumar, A. Raghuvanshi, Unsteady incompressible flows past two cylinders in tandem and staggered arrangements, Int. J. Numer. Meth. Fluids 25 (1997) 1315-1344, https://doi.org/10.1002/(SICI)1097-0363(19971215)25:11<1315::AID-FLD617>3.0.CO;2-P.
[75] N. Mahir, Z. Altaç, Numerical investigation of convective heat transfer in unsteady flow past two cylinders in tandem arrangements, Int. J. Heat Fluid Fl. 29 (2008) 1309-1318, https://doi:10.1016/j.ijheatfluidflow.2008.05.001.
[76] W.W. Ren, J.Y. Shao, L.M. Yang, T.Q. Li, Numerical study of heat transfer from two cylinders in tandem with transverse oscillation, Adv. Appl. Math. Mech. 12(5) (2020) 1137-1165, https://doi:10.4208/aamm.OA-2019-0253.