Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 399 (2022) 115449
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

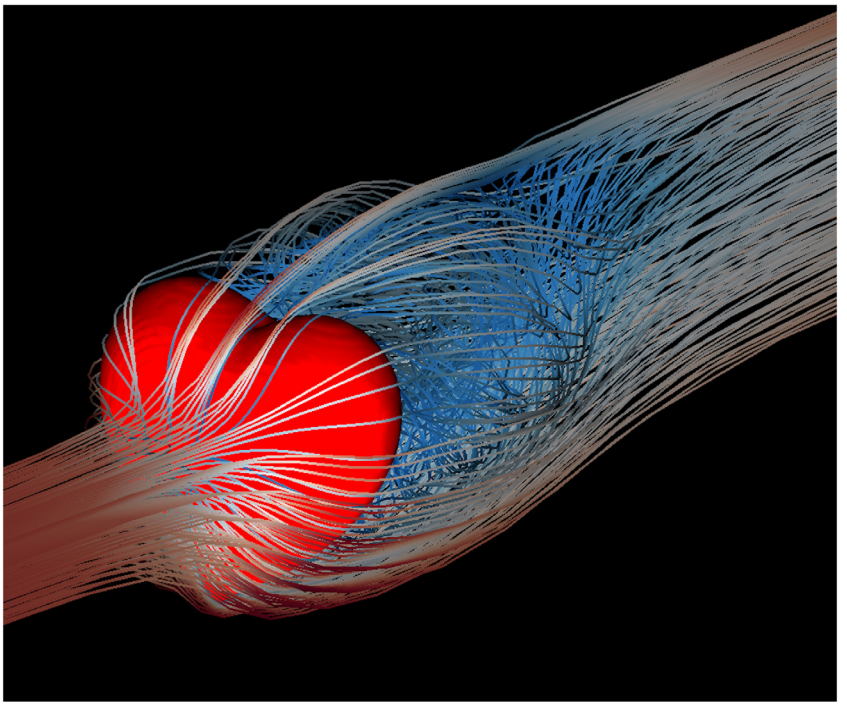
An implicit Cartesian cut-cell method for incompressible viscous flows with complex geometries

Zhihua Xie

School of Engineering, Cardiff University, Cardiff, CF24 3AA, UK

Received 23 March 2022; received in revised form 19 July 2022; accepted 19 July 2022
Available online xxxx

Graphical Abstract



Abstract

A versatile conservative three-dimensional Cartesian cut-cell method for simulation of incompressible viscous flows over complex geometries is presented in this paper. The present method is based on the finite volume method on a non-uniform

E-mail address: zxie@cardiff.ac.uk.

https://doi.org/10.1016/j.cma.2022.115449

staggered grid together with a consistent mass and momentum flux computation. Contrary to the commonly cut-cell methods, an implicit time integration scheme is employed in the present method, which avoids numerical instability without any additional small cut-cell treatment. Strict conservation of the mass and momentum for both fluid and cut cells is enforced through the PISO algorithm for the pressure–velocity coupling. The versatility and robustness of the present cut-cell method are demonstrated by simulating various two- and three-dimensional canonical benchmarks (flow over a circular cylinder, airfoil, sphere, pipe, and heart sculpture) and the computed results agree well with previous experimental measurements and various numerical results obtained from the boundary-fitted, immersed boundary/interface, and other cut-cell methods, verifying the accuracy of the proposed method.

Keywords: Cartesian cut-cell method; Finite volume method; Incompressible flow; Complex geometries; Three-dimensional

1. Introduction

In computational fluid dynamics, two- (2D) and three-dimensional (3D) fluid flow over complex geometries appear in many scientific research and engineering applications, e.g. flow over urban structures, automotive aerodynamics, marine hydrodynamics, hydraulic and coastal engineering, and fluid–structure interaction problems. In contrary to dealing with the complex geometries in boundary-fitted methods [1,2], the Cartesian grid method has become a popular alternative to solve fluid flow in a fixed Cartesian grid due to its simplicity for mesh generation and easy data structure management, especially for moving bodies [3,4].

There are several Cartesian grid methods, such as reviews for immersed boundary method (IBM) [5], immersed interface method (IIM) [6], and the Cartesian cut-cell method [7]. For the immersed boundary method, it can be classified as diffused interface and sharp interface method [3]. The original IBM [8] is a continuous forcing diffused interface IBM, in which the force at the immersed points is computed from appropriate constitutive law and then spreads to the background grid by using a discrete delta function. Later, direct forcing diffused interface IBM is also proposed for rigid bodies [9], in which the force at the immersed boundary points is calculated from the discretised momentum equation using the interpolated velocity at the immersed points, and then redistributed to the Eulerian grid by a delta function. Contrary to the diffused interface IBM, a sharp interface IBM has proposed for the spectral [10] and finite difference [11] methods. Later various sharp interface IBMs [12–19] have been developed, mainly based on the different interpolation of the velocity for the neighbouring or ghost cells. A projection approach is also proposed for IBM and its difference compared to other IBMs can be found in [20].

It is worth noting that the IBM is normally implemented in the finite difference method coupled with a fractional step approach. The force introduced for the momentum equation does not satisfy the global mass conservation [3]. Some research has been developed to alleviate this issue by adding mass sink/source term in the continuity equation [12], an implicit treatment of both the boundary force and pressure as a single set of Lagrange multipliers in the Poisson equation [20], and combining cut-cell and IBM to deal with the continuity and velocity correction in the vicinity of immersed boundary [21].

Contrary to IBM, the Cartesian cut-cell method is very attractive as it enforces strict conservation of mass, momentum and energy at a discretised level, even near the immersed boundary. The Cartesian cut-cell method has been applied for fixed solid boundaries for 2D inviscid [22,23] and viscous [24–26] fluid flow and 3D inviscid [27,28], viscous [29–31], and turbulent [32] flows. The Cartesian cut-cell method can also be further extended for free-surface/two-phase flows [33–39] and moving body problems [40–46]. In conjunction with other interface capturing/tracking methods or structural solvers, the Cartesian cut-cell methods have also been developed for multiphase flows with moving bodies [47–52] and fluid–structure interaction [53–55]. A hybrid cut-cell and ghost-cell method has been developed in [56]. The cut-cell method can also be employed in the finite element [57–63] and finite difference [23] method. A common problem for cut-cell methods is the small cut-cells generated near the solid boundary which lead to numerical instability if there is no special treatment for small cells [64], which makes it challenging to apply in 3D flow problems.

For incompressible Navier–Stokes equations, the coupling between velocity and pressure is a key concern and normally a staggered grid is used in most discretisation methods. However, most of the recent 3D cut-cell methods [30,31,41–43] are developed based on the collocated grid for compressible flows. In addition, most of the

3D cut-cell methods employ an explicit scheme and additional treatment for the small cut-cells is needed to avoid the numerical instability.

The objective of this work is, therefore, to present a versatile finite volume-based Cartesian cut-cell method for the unsteady, incompressible, Navier–Stokes equations on a 3D non-uniform staggered grid. The novelty of this paper is threefold: (1) the generation and configuration of both 2D and 3D cut-cells are presented in detail and the finite volume discretisation is presented for both fluid and cut cells on the staggered grid, which is different from the previously used collocated grids; (2) an implicit time integration scheme is used for the governing equations without any small-cell treatment, which avoids the common instability problems in small cut-cells existing in the literature; (3) a consistent mass and momentum flux computation is used to ensure strict conservation of mass and momentum, even for the cut-cells, which is different from most of the IBMs where mass conservation near the immersed boundary is not enforced. Compared to our previous study for simple moving bodies for single-phase [46] and two-phase [52] flows, this study focuses on the incompressible viscous flow over fixed bodies, but with more complex geometries. In addition, both external flow over bluff bodies and internal flow inside fixed structures are studied. Several canonical 2D and 3D benchmark problems are validated, with a cross comparison being made against available analytical solutions and experiments, as well as some other boundary-fitted, immersed boundary, immersed interface and cut-cell methods published in the literature.

The paper is organised as follows. The description of the mathematical formulation, finite volume discretisation, and Cartesian cut-cell method are presented in Section 2. The versatility and robustness of the present cut-cell method are demonstrated by a number of 2D and 3D external and internal flow problems in Section 3. Conclusion and future work are finally discussed in Section 4.

2. Mathematical formulation and numerical method

2.1. Governing equations

The governing equations considered are the non-dimensional unsteady incompressible Navier–Stokes equations, given as:

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1}$$

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

where $\boldsymbol{u}$ is the non-dimensional velocity vector with components (u, v, w) in the streamwise (x), vertical (y) and spanwise (z) direction, t and p are the non-dimensional time and pressure, and Re is the Reynolds number.

2.2. Computational grid and finite volume discretisation

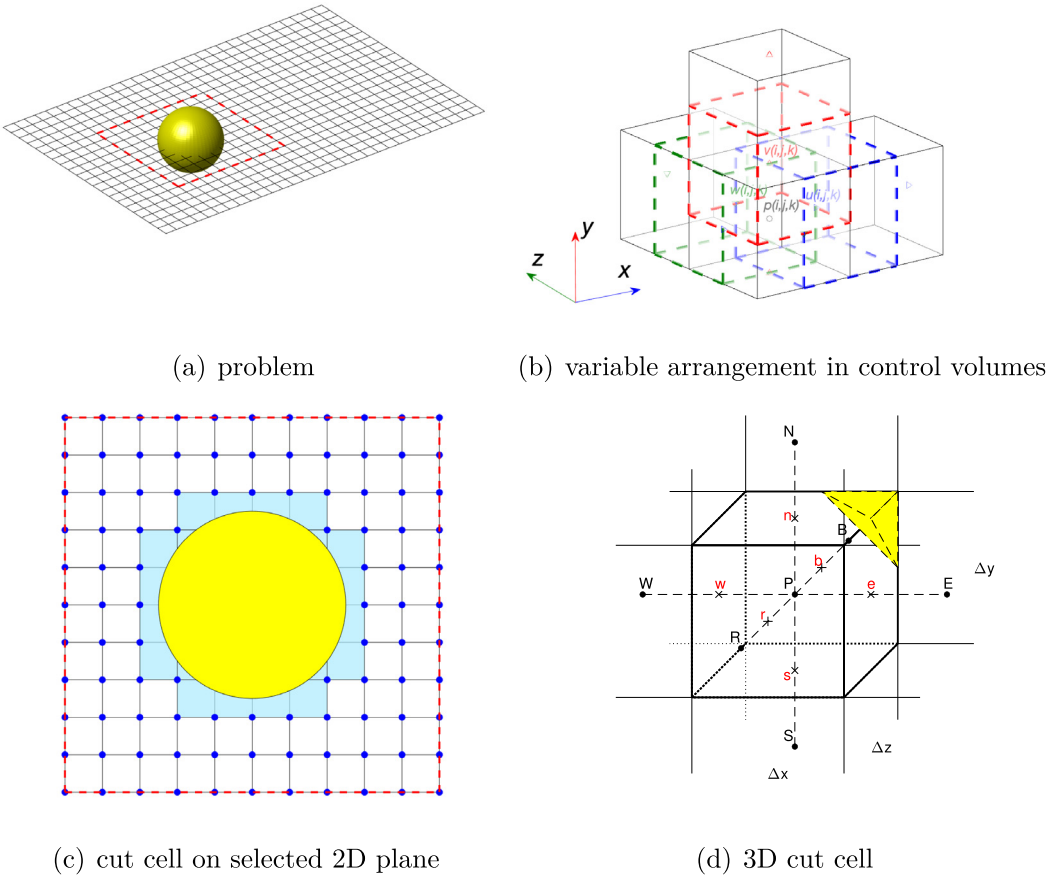
The finite volume method is employed to discretise the governing equations because it enforces the conservation of mass and momentum at a discretised level. The staggered Cartesian grid is used in this study, which has the advantage of strong coupling between the velocity and the pressure. Fig. 1 shows an example of the problem setup and the variable arrangement in a 3D Cartesian grid with cut-cells together with the name of locations used in the discretisation, in which the pressure is stored at the cell centre and the velocities are located on the face centre of the control volume.

Considering a volume of fluid cell Ω which has an arbitrary domain, the surface of the control volume is S and the unit outward normal vector to the face f is $\boldsymbol{n}$. The momentum equation (Eq. (2)) can be recast into an integral formulation as below

$$\iiint_{\Omega} \frac{\partial \boldsymbol{u}}{\partial t} \mathrm{d}\Omega + \iint_{S} (\boldsymbol{u} \cdot \boldsymbol{n})\boldsymbol{u} \mathrm{d}S = \iiint_{\Omega} -\nabla p \mathrm{d}\Omega + \iint_{S} \frac{1}{Re}\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} \mathrm{d}S, \tag{3}$$

For a full fluid cell, the cell volume is $\Omega = \iiint_{\Omega} \mathrm{d}\Omega = \Delta x \Delta y \Delta z$ and the area of the face A is similarly calculated, *e.g.,* the one of the east face A_e is $A_\mathrm{e} = \iint_\mathrm{e} \mathrm{d}S = \Delta y \Delta z$.

A backward finite difference is used for the time derivative $\frac{\partial \boldsymbol{u}}{\partial t} = \frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t}$, which leads to an implicit scheme for the Navier–Stokes equations for the current time level $n + 1$. The high-resolution scheme [65] is used for the



(a) problem (b) variable arrangement in control volumes

(c) cut cell on selected 2D plane (d) 3D cut cell

Fig. 1. Schematic of a three-dimensional flow over a complex geometry: (a) 3D computational setup with a 2D plane shown for the Cartesian grid (dash line frame: the selected 2D mesh shown in (c)); (b) Variables used for the control volume (i, j, k) in a 3D staggered grid. Pressure $p(i, j, k)$ is stored in the centre of the control volume whereas velocities $u(i, j, k)$, $v(i, j, k)$, and $w(i, j, k)$ are stored in the centre of the six faces of the control volume. (c) The selected 2D plane from (a) shows the background mesh together with the full fluid cells (white) and cut-cells (blue) and the solid domain (yellow); (d) Example of a 3D cut-cell with the yellow shaded area being solid. Name of the locations used in the discretisation is also shown, in which P is the present node, the upper-case letter E, W, N, S, B, and R denote neighbouring nodes on the east, west, north, south, back, and front with respect to the central node P. The lower-case e, w, n, s, b, and r denote the corresponding face of the control volume. Δx, Δy, and Δz are the grid spacing in the x, y, and z directions, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

advective flux. The second-order central difference scheme is used for diffusive flux, pressure gradient term and the pressure correction equations. It is worth noting that a consistent mass and momentum approach is used here for the staggered grid to discretise the nonlinear term, in which the mass flux $m_{\mathrm{f}} = \iint_S \boldsymbol{u} \cdot \boldsymbol{n} \mathrm{d}S$ for the momentum control volume is calculated based on the interpolation of the mass flux already available for the continuity equation. However, in cut cells, all these discretised terms will need to be modified and this will be discussed in detail in Section 2.3. Substituting all the discretised terms into Eq. (3), leads to

$$a_{\mathrm{P}}^{\boldsymbol{u}} \boldsymbol{u}_{\mathrm{P}}^{n+1} = \sum a_{\mathrm{nb}}^{\boldsymbol{u}} \boldsymbol{u}_{\mathrm{nb}}^{n+1} + b_{\mathrm{P}}^{\boldsymbol{u}} + A_{\mathrm{f}}(p_{\mathrm{P}} - p_{\mathrm{nb}}).,\tag{4}$$

where $a^{\boldsymbol{u}}$ is the coefficient for the momentum equation, the subscripts P and nb = E, W, N, S, B, R denote the variables in the present and neighbouring cells (shown in Fig. 1), respectively, and $b_{\mathrm{P}}^{\boldsymbol{u}}$ is the source term contained $\boldsymbol{u}^n$ and high-order terms due to the high-resolution scheme.

The PISO algorithm [66] is employed in this study for the pressure–velocity coupling and it is used to calculate the corrected pressure twice. For a guessed pressure distribution p^*, the discretised momentum equations can be

solved to produce the fluid velocities $\boldsymbol{u}^*$, which satisfy the momentum equation (Eq. (4))

$$a_{\mathrm{P}}^{\boldsymbol{u}}\boldsymbol{u}_{\mathrm{f}}^* = \sum a_{\mathrm{nb}}^{\boldsymbol{u}}\boldsymbol{u}_{\mathrm{nb}}^* + b_{\mathrm{P}}^{\boldsymbol{u}} + A_{\mathrm{f}}(p_{\mathrm{P}}^* - p_{\mathrm{nb}}^*). \tag{5}$$

To obtain the pressure correction p', the updated fluid velocities are substituted into the discretised continuity equation (Eq. (1)) and the resulting pressure correction equation has the following form

$$a_{\mathrm{P}}^{p} p_{\mathrm{P}}' = \sum a_{\mathrm{nb}}^{p} p_{\mathrm{nb}}' + b_{\mathrm{P}}', \tag{6}$$

where a^p is the coefficient for the continuity equation and the term b_{P}', called the mass residual, is the left-hand side of the discretised continuity equation evaluated in terms of the fluid velocities $\boldsymbol{u}^*$.

A second pressure correction (p'') step is introduced in the PISO algorithm [66] as

$$a_{\mathrm{P}}^{p} p_{\mathrm{P}}'' = \sum a_{\mathrm{nb}}^{p} p_{\mathrm{nb}}'' + b_{\mathrm{P}}'', \tag{7}$$

where the coefficients have the same value in the first pressure correction equation shown in Eq. (6) and the source term has been changed for calculating the mass residual based on the value of first velocity correction $\boldsymbol{u}'$.

In this study, the algebraic equations are solved by the strongly implicit procedure method or Bi-CGSTAB (Bi-Conjugate Gradients Stabilized) method [67]. After solving the first and second pressure corrections (Eqs. (6) and (7)), the solutions in a continuity control volume are updated as

$$\begin{aligned} p &= p^* + p' + p'', \\ \boldsymbol{u}_{\mathrm{f}} &= \boldsymbol{u}_{\mathrm{f}}{}^* + \boldsymbol{u}_{\mathrm{f}}' + \boldsymbol{u}_{\mathrm{f}}'', \end{aligned} \tag{8}$$

where

$$\begin{aligned} \boldsymbol{u}_{\mathrm{f}}' &= \frac{A_{\mathrm{f}}}{a_{\mathrm{P}}^{\boldsymbol{u}}}(p_{\mathrm{P}}' - p_{\mathrm{nb}}'), \\ \boldsymbol{u}_{\mathrm{f}}'' &= \frac{\sum a_{\mathrm{nb}}^{\boldsymbol{u}}\boldsymbol{u}_{\mathrm{nb}}' + A_{\mathrm{f}}(p_{\mathrm{P}}'' - p_{\mathrm{nb}}'')}{a_{\mathrm{P}}^{\boldsymbol{u}}}. \end{aligned} \tag{9}$$

It is worth noting that the implicit discretisation for the velocity prediction and pressure correction makes the PISO algorithm stable for fairly large time steps [66] and has also been employed for hybrid unstructured grids [68]. The detailed stability analysis of the PISO algorithm on both collocated and staggered grids can be found in [69] and will not be further discussed here.

2.3. Cartesian cut-cell method

2.3.1. Geometric representation of complex geometries

In this study, the complex geometry of a solid is represented by a general level set function $\varphi(x, y, z)$ (a formula for a 2D curve or 3D surface, and level-sets for a complex surface), in which the value $\varphi(x, y, z) > 0$ in the fluid domain and $\varphi(x, y, z) < 0$ inside the solid domain. The solid boundary is represented as a sharp piecewise linear interface when $\varphi(x, y, z) = 0$, and it is a straight line in 2D and a sloping plane in 3D.

2.3.2. Cut-cell information

In the finite volume discretisation for each 3D Cartesian grid cell, both the area for each face of the control volume and the total fluid volume in a cell are needed. For full fluid cells, this has been discussed above. For cut-cells, this information needs to be changed according to the geometric information for each cell. In order to have a general framework for both full fluid and cut-cells, an additional θ function is introduced in the spatial discretisation, which represents the ratio for the face area A or volume Ω between the cut-cell and the original full-cell. Thus, the face area and volume in a cut-cell can be obtained as $A_{\mathrm{f}} = \theta_{\mathrm{f}}A$ and $\Omega_{\mathrm{c}} = \theta_{\mathrm{c}}\Omega$, respectively. The θ function has values at the volume centre and each face of the control volume and it is calculated at the beginning of the simulation. Its value is 1 for a full fluid cell and 0 for a solid cell inside the solid boundary, whereas $0 < \theta < 1$ in cut-cells.

To calculate the area and volume of the cut-cell, how the solid boundary cuts the Cartesian cell is determined first. Fig. 2 shows an example of 2D cut-cell configurations, in which one to three of the four edge points have been cut in a control volume. The grid nodes can be classified as inside or outside of the solid based on the sign of

the level set function $\varphi(x, y, z)$ (shown as blue for a positive value and red for a negative value). It can be seen that there are 4 grid nodes in a 2D cell, thus there are 2^4 configurations. For a 3D cell, it would be more complicated and there are 8 grid nodes in each cell so it has 2^8 configurations. Following the marching tube algorithm [70] and using the inversion and rotation of different configurations, the 256 cases can be rearranged to 15 cases as shown in Fig. 3 for a typical cut cell in a 3D Cartesian grid, which is classified as zero, one, two, three, and four points being cut out of the eight points in the Cartesian grid cell.

Once the configuration of the cut cell has been determined, the intersection point along each edge can be obtained by linear interpolation of the level set function of two neighbouring grid nodes. Once the point of intersection of the line with the cut plane is found, the geometric information can be calculated. Finally, the face area and total volume of the truncated cell are calculated by numerical integration [64]. In contrary to a full fluid cell, the spatial discretisation at cell faces and cell centres is modified in a cut cell with the θ value, which will be presented for the momentum and continuity equations.

2.3.3. Cut-cell treatment for the momentum equation

The finite volume discretisation of the advection term in Eq. (3) is obtained as

$$\iint_S (\boldsymbol{u}\cdot\boldsymbol{n})\boldsymbol{u}\mathrm{d}S = \sum_{\mathrm{f}} m_{\mathrm{f}}\boldsymbol{u}_{\mathrm{f}}, \tag{10}$$

where $m = \boldsymbol{u}\cdot\boldsymbol{n}\theta A$ is the mass flux and the subscript f denotes the corresponding face of the control volume. In cut cells, the mass flux has also to be modified by the θ function on the boundary as shown in Figs. 2 and 3. If $\theta = 0$ (such as the west face of the first case in Fig. 3(e)), there is no mass flux through the face and the advective flux is obtained as $m_{\mathrm{f}} = 0$.

The finite volume discretisation of the diffusion term in Eq. (3) for cut-cells is obtained as

$$\iint_S \frac{1}{Re}\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}\mathrm{d}S = \sum_{\mathrm{f}} \frac{1}{Re}\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}(\theta A)_{\mathrm{f}} + \tau_{\mathrm{w}}[(1-\theta)A]_{\mathrm{f}}, \tag{11}$$

where $\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}$ is calculated by the finite difference approach from the present point P to the neighbouring point nb as $\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} = \frac{\boldsymbol{u}_{\mathrm{nb}}-\boldsymbol{u}_{\mathrm{P}}}{\Delta_{\mathrm{Pnb}}}$, and τ_{w} is the wall shear stress on the face of the control volume.

The volume of the cell has also been changed and the finite volume discretisation of the pressure gradient in Eq. (3) for cut-cells is obtained as

$$\iiint_{\Omega} -\nabla p\mathrm{d}\Omega = -\nabla p\theta_{\mathrm{c}}\Omega, \tag{12}$$

and the pressure gradient is calculated as

$$\begin{aligned}\nabla p &= \left(\frac{\partial p}{\partial x}, \frac{\partial p}{\partial y}, \frac{\partial p}{\partial z}\right)\\ &= \left(\frac{p_{\mathrm{e}}-p_{\mathrm{w}}}{\Delta x}, \frac{p_{\mathrm{n}}-p_{\mathrm{s}}}{\Delta y}, \frac{p_{\mathrm{b}}-p_{\mathrm{r}}}{\Delta z}\right).\end{aligned} \tag{13}$$

2.3.4. Cut-cell treatment for the continuity equation

When dealing with the continuity equation, the exact conservation of mass is enforced to the discretised level in cut-cells and the mass residuals (b'_{P} and b''_{P}) for the pressure correction equations (Eqs. (6) and (7)) are obtained as

$$\begin{aligned} b'_{\mathrm{P}} &= \sum_{\mathrm{f}} m_{\mathrm{f}} = \sum_{\mathrm{f}} \boldsymbol{u}^*_{\mathrm{f}}\cdot\boldsymbol{n}(\theta A)_{\mathrm{f}},\\ b''_{\mathrm{P}} &= \sum_{\mathrm{f}} \boldsymbol{u}'_{\mathrm{f}}\cdot\boldsymbol{n}(\theta A)_{\mathrm{f}}.\end{aligned} \tag{14}$$

It can be seen that the cut-cell information for each control volume and their face area is taken into account in the discretisation of the continuity equation to satisfy the divergence-free constraint at the current time level $n+1$.

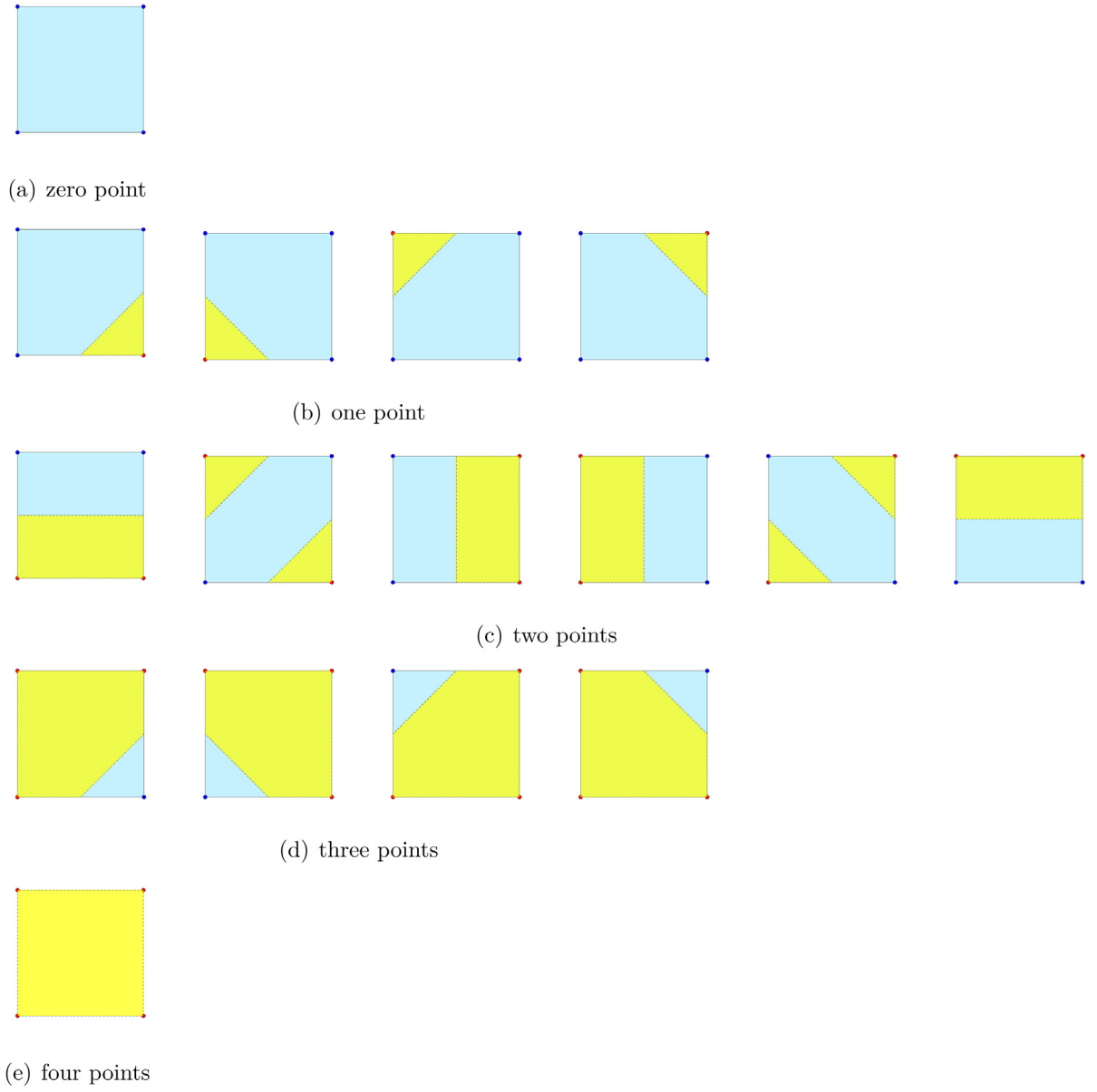


Fig. 2. Example of 2D cut cell configurations. Blue points are inside a fluid domain and red points are inside a solid domain while the solid boundary is represented by the interface between the blue and yellow interface. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2.4. Boundary and initial conditions

It is necessary to define the boundary conditions in a computational domain to completely describe the numerical model. For the inlet, the Dirichlet boundary condition is used for the velocity. No-slip or free-slip boundary conditions can be specified for the sidewalls. For the outlet, the zero-gradient condition is applied for the flow. For the solid boundary on the geometry, Dirichlet boundary condition is specified for the velocity and the zero-gradient condition is used for the pressure.

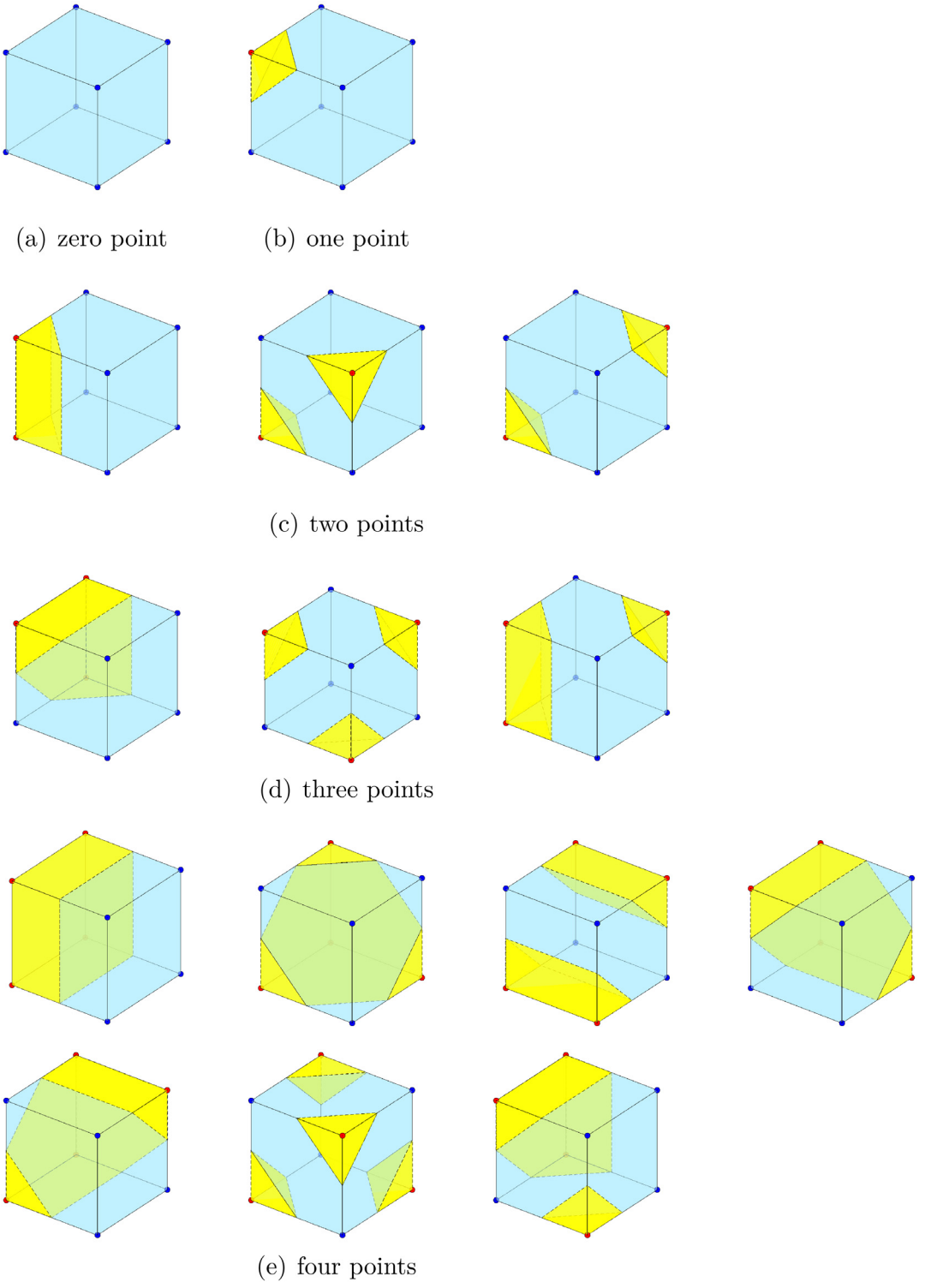


Fig. 3. Example of 3D cut cell configurations, which indicate how many edge points are in the fluid (blue) and solid (red) domain in a Cartesian grid cell: (a) zero point (either pure fluid or solid cell); (b) one point; (c) two points; (d) three points; (e) four points. In total, there are 256 cases, but all cases can be rearranged to 15 cases shown here after inversion and rotation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

In the computation, the initial flow field at $t = 0$ has to be prescribed. For calculations with the fluids initially at rest, the flow field is initialised with zero velocity. In order to speed up the flow development, the flow field can also be initialised with the same inlet velocity everywhere inside the computational domain.

2.5. Comparison with other cut-cell methods

For most of the Cartesian cut-cell methods, an explicit time integration scheme is normally used thus an instability problem might occur in small cut-cells in the vicinity of the solid boundary. Thus a very small time step has to be used to satisfy the stability criteria. Some approaches have been developed to deal with small cut-cells [71], such as the cell-merging technique [64] and using slightly different control volumes [48], both of which effectively increase the size of the cut cell. Recently, as an alternative, there are also some developments for the flux-redistribution schemes [41–43] and a state redistribution algorithm [22] to deal with the stability issue for the small cut-cells. A new mixed explicit implicit time stepping scheme is developed for advection and Euler equations [72], where an implicit scheme is used for the cut-cell for the stability and explicit scheme is employed for standard cells. Because of the difficulty of extra treatment in 3D, some of the previous studies are focused on 2D problems.

In cut finite element methods, there are also Ghost Penalty methods [73] to tackle the small cut-cell instability issue. The main role of the penalty term is to extend the coercivity from the physical domain to the mesh domain with some penalty parameters.

In the present Cartesian cut-cell method, there is no special treatment for the small cut-cells and the instability issues are not found in the simulations thanks to the implicit time integration scheme used for both full and cut cells in this study.

3. Results and discussion

In order to validate the proposed Cartesian cut-cell method, we first simulate the Taylor–Green vortices with and without the immersed domain and both spatial and temporal convergence are investigated. The proposed method is then applied to some canonical problems in two-dimensional flows (such as flow over a cylinder and an airfoil), and three-dimensional external flow over a sphere and internal flow in a pipe. The results are compared with available experimental measurements and analytical solutions, as well as previous simulation results, obtained by other established immersed boundary, immersed interface, and boundary-fitted methods. Finally, the method is further used to study the flow past a heart sculpture to demonstrate its versatility to deal with complex geometries.

3.1. Convergence study: Taylor–Green vortices

In order to study the effect of the cut-cell method to deal with complex geometries in a Cartesian grid and to investigate the spatial and temporal convergence of the proposed method, the two-dimensional unsteady flow for Taylor–Green vortices is considered here with an analytical solution as

$$\begin{aligned} u(x, y, t) &= -\cos(\pi x)\sin(\pi y)e^{-2\pi^2 t/Re}, \\ v(x, y, t) &= \sin(\pi x)\cos(\pi y)e^{-2\pi^2 t/Re}, \\ p(x, y, t) &= -\frac{1}{4}[\cos(2\pi x) + \cos(2\pi y)]e^{-4\pi^2 t/Re}. \end{aligned} \tag{15}$$

This case has also been computed using the immersed boundary method for an immersed square [12] and circular [9] domains. The computational setup is shown in Fig. 4, where the computational domain ($-1.5 \leq x/L, y/L \leq 1.5$) is normalised by the vortex length L and the immersed circular domain is centred at the origin of the domain with a radius of unity. The same parameter from [9] is used here with $Re = \frac{\rho UL}{\mu} = 5$, where U is the maximum initial velocity, and the simulation is run to $t = 0.3$ with a time step $\Delta t = 0.001$. Both cases with and without the immersed circular domain are investigated using various spatial and temporal resolutions. The initial condition and boundary condition along the simulation are provided by the analytical solution Eq. (15). For the circular domain simulation, the analytical solution Eq. (15) also provides the value for the cut region on the circular, which is not included in the computation for the present solver.

In order to study the spatial convergence, four different mesh resolutions (12, 24, 48, and 96 cells in each direction, respectively) are used in the simulation with a constant time step $\Delta t = 0.001$. Fig. 5(a) shows the

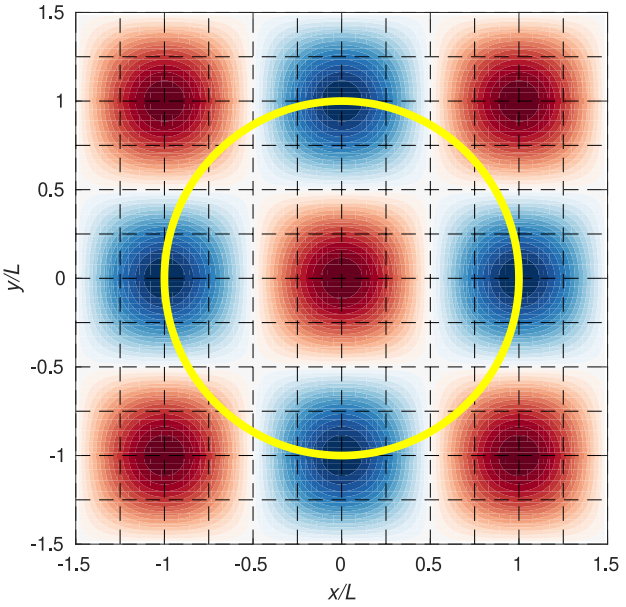


Fig. 4. Computational domain, grid system, and immersed boundary (yellow lines) for the Taylor–Green decaying vortices. The initial vorticity field is also shown here with blue indicating clockwise rotation and red indicating counter-clockwise rotation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

variation of the maximum error and L_2 norm of the velocity for grid points inside the immersed domain. It can be observed that second-order convergence is obtained for both cases with and without cut-cell treatments, which is consistent with the high-resolution scheme used here. It is also worth noting that the errors are not so sensitive to the Cartesian cut-cell method and only a slightly higher error is introduced for the cut-cell discretisation.

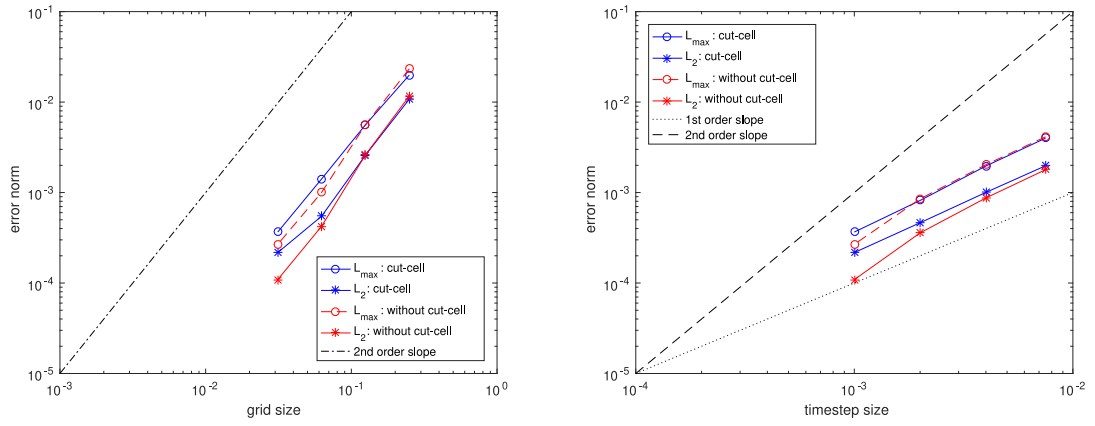
In order to study the temporal convergence, four different time steps ($\Delta t = 0.001, 0.002, 0.004, 0.008$) are used for the grid (96×96 cells). Fig. 5(b) shows the variation of the maximum error and L_2 norm with respect to the time step sizes and it can be seen that the convergence is first-order, which is consistent with the first-order backward Euler time integration method.

3.2. 2D uniform flow past a circular cylinder

Here the steady and unsteady flows past a circular cylinder in a uniform incoming flow are studied, which is considered as a standard benchmark problem for computational fluid dynamics. Some interesting flow patterns can be observed depending on the Reynolds numbers $Re = \frac{\rho U D}{\mu}$, where U is the incoming velocity magnitude and D is the diameter of the cylinder. Although the cylinder can be easily dealt with a boundary-fitted method, it is challenging to accurately predict the flow behaviour using Cartesian grid methods. As many experimental and numerical data sets exist in the literature and they will be selected to make cross-comparison between different methods.

In the present study, the time dependent drag and lift coefficient are defined as:

$$C_{\mathrm{D}}(t) = \frac{F_{\mathrm{D}}(t)}{\frac{1}{2}\rho U^2 D}, \qquad C_{\mathrm{L}}(t) = \frac{F_{\mathrm{L}}(t)}{\frac{1}{2}\rho U^2 D}, \tag{16}$$



(a) spatial convergence (b) temporal convergence

Fig. 5. Maximum error and L_2 norm of the velocity u for the Taylor–Green vortices at time $t = 0.3$ with and without Cartesian cut-cell treatment.

Table 1
Comparison between experimental results, other simulations, and the present study for flow over a cylinder at $Re = 40$.

Study	Method	l/D	a/D	b/D	α	C_D
Tritton [74]	experiment	–	–	–	–	1.59
Coutanceau and Bouard [75]	experiment	2.13	0.76	0.59	53.8	–
Fornberg [76]	body-fitted	2.24	–	–	55.6	1.50
Tseng and Ferziger [13]	IBM-ghost cell	2.21	–	–	–	1.53
Taira and Colonius [20]	IBM-projection	2.30	0.73	0.60	53.7	1.54
Mittal et al. [17]	IBM-sharp interface	–	–	–	–	1.53
Berthelsen and Faltinsen [18]	IBM-ghost cell 1D	2.29	0.72	0.60	53.9	1.59
Linnick and Fasel [77]	IIM	2.28	0.72	0.60	53.6	1.54
Ye et al. [24]	cut-cell	2.27	–	–	–	1.52
Kirkpatrick et al. [29]	cut-cell	2.259	–	–	53.55	1.535
Cheny and Botella [78]	cut-cell	2.299	–	–	–	1.508
Hartmann et al. [30]	cut-cell	2.306	–	–	–	1.521
Muralidharan and Menon [31]	cut-cell	–	–	–	–	1.56
Present ($D/h = 32$)	cut-cell	2.21	0.708	0.594	53.3	1.526
Present ($D/h = 64$)	cut-cell	2.21	0.708	0.594	53.3	1.534

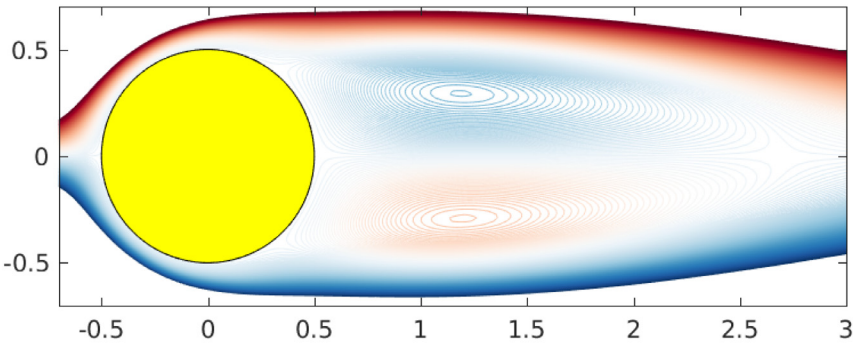
where $F_D(t)$ and $F_L(t)$ are the drag and lift force, respectively, and the force $\mathbf{F} = (F_D, F_L)$ is calculated by integration of the pressure and shear stresses along the cut-cell boundaries Γ as:

$$\mathbf{F} = \int_{\Gamma} (-\boldsymbol{n}p + \boldsymbol{n} \cdot \boldsymbol{\tau}) \mathrm{d}\Gamma. \tag{17}$$

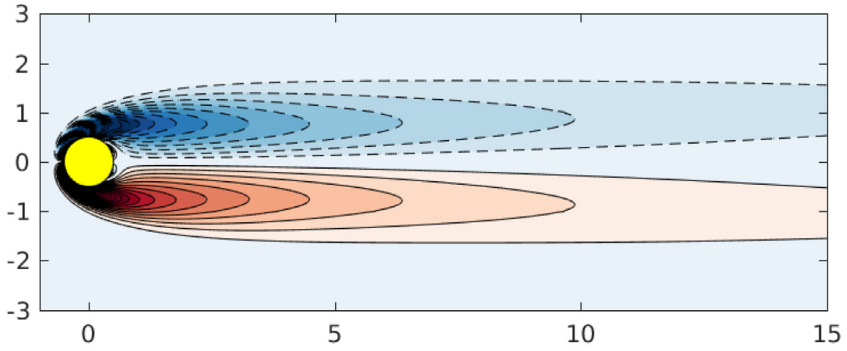
The Strouhal number is calculated as $St = \frac{fD}{U}$, where f is the vortex shedding frequency calculated from the lift force $F_L(t)$ in the unsteady flow regime.

3.2.1. $Re = 40$

The simulations for flow at $Re = 40$ have been carried out in a large square domain ($[-20D, 20D] \times [-20D, 20D]$) with the cylinder located at the centre. The uniform velocity U is specified at the inlet whereas the zero gradient boundary condition is applied for other variables along the boundaries. Two non-uniform Cartesian grids (200×200, 400×400) are used to discretise the computational domain with uniform meshes $h = \Delta x_{\min} = \Delta y_{\min}$ ($h/D = 1/32, 1/64$) in the vicinity of the cylinder. A constant time step $\Delta t = 0.001D/U$ is used in the simulation in order to minimise the temporal discretisation error.



(a) streamline



(b) vorticity

Fig. 6. Uniform flow past a cylinder at $Re = 40$. Contours of stream function and vorticity are shown as red for positive value and as blue for negative value. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 6 shows predicted streamlines and vorticity for flow over the cylinder for $Re = 40$ at the steady-state. It can be seen that the flow is symmetric and vorticity is generated with an opposite sign at both sides of the cylinder, which is similar to other results obtained in the literature. The predicted drag coefficients and wake characteristics are also compared with experimental and numerical results reported in the literature in Table 1. The same parameters l, a, b, α discussed in [20,75] are used here, where l is the length of the circulation zone, a is the distance between the centre of the wake vortex and the cylinder, b is the vertical distance between the centre of two wake vortices, and α is the separation angle respected to the horizontal axis, respectively.

Table 1 shows the comparison of the values obtained for the same case with several experimental and numerical results, including the body-fitted, various immersed boundary, immersed interface, and other cut-cell methods. It can be seen that the mesh convergent drag coefficients and wake vortex parameters are obtained for the present method, which is in good agreement with those results reported in the literature.

Fig. 7 shows the pressure coefficient (C_p) and skin-friction (C_f) along the surface of the cylinder for $Re = 40$, together with those results obtained by the boundary-fitted and ghost-cell methods reported in [13]. It can be seen that the present results are very close to the results obtained by the ghost-cell method, and both are very similar to the boundary-fitted results which have better representation for the wall-normal derivatives.

In order to demonstrate the benefit of the implicit scheme to deal with small cut cells, additional three simulations have been carried out for the fine mesh with larger time steps when CFL$= U\Delta t/h = 0.25$, 0.5, and 1.0, in which the CFL number is defined based on the full cells. Fig. 8 shows the predicted drag (C_D) as a function of the CFL number. It can be seen that the drag value is convergent and it is gradually decreased with the decrease of CFL

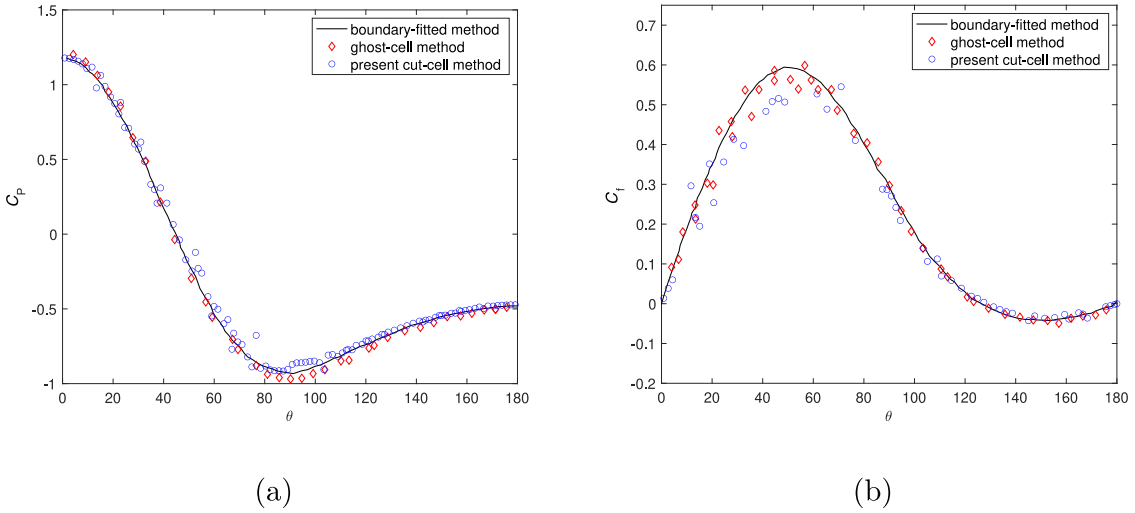


Fig. 7. The pressure coefficient C_p (a) and skin-friction C_f (b) for flow over a stationary cylinder at $Re = 40$ and comparison with other methods reported in [13].

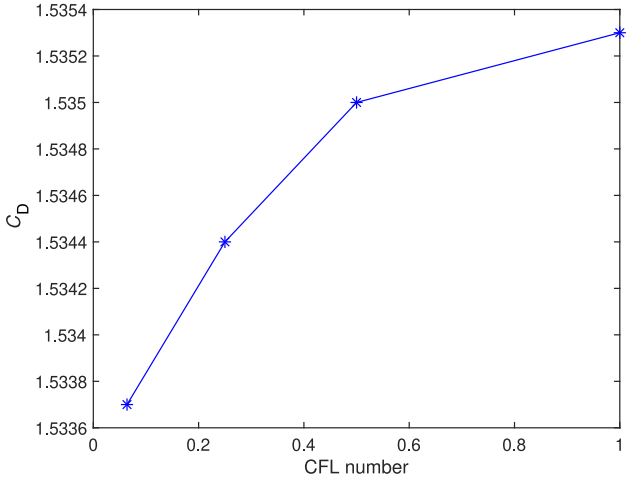


Fig. 8. The CFL number effect on the drag coefficient C_D for flow over a stationary cylinder at $Re = 40$.

number. It is worth noting that when the CFL number is equal to 1.0, the local CFL number for cut cells would be larger than 1.0 as the local grid size in cut cells is smaller than h, which demonstrates the advantage of the present implicit cut-cell method.

3.2.2. $Re = 100$

It is noted that when at a higher Reynolds number, the two vortices and symmetric flow shown in Fig. 6 become unstable, and the vortex shedding will start. In order to capture the vortex shedding, the flow at $Re = 100$ is simulated in a slightly larger computational domain ($[-20D, 40D] \times [-20D, 20D]$) with the cylinder located at (0, 0). As this is a benchmark case for CFD, especially for Cartesian grid methods. Four different non-uniform meshes (150×132, 300×264, 600×528, 1200×1056) are considered with various minimum resolutions ($h/D = 1/16, 1/32, 1/64, 1/128$) to capture the cylinder in a Cartesian coordinate.

Fig. 9 shows an instantaneous vorticity field for flow over a stationary cylinder at $Re = 100$. It can be seen that the von Kármán vortex sheet with counter-rotating vortices are well captured by the present method.

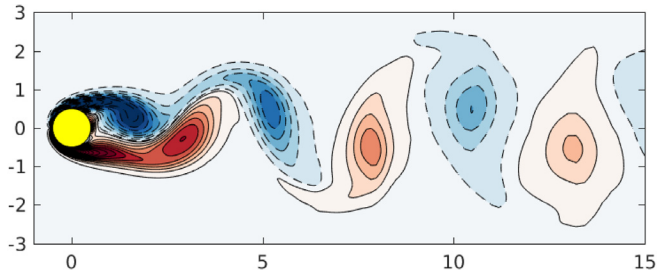


Fig. 9. The instantaneous vorticity field in the wake of the flow over a stationary cylinder with the mesh resolution $h/D = 1/64$ at $Re = 100$.

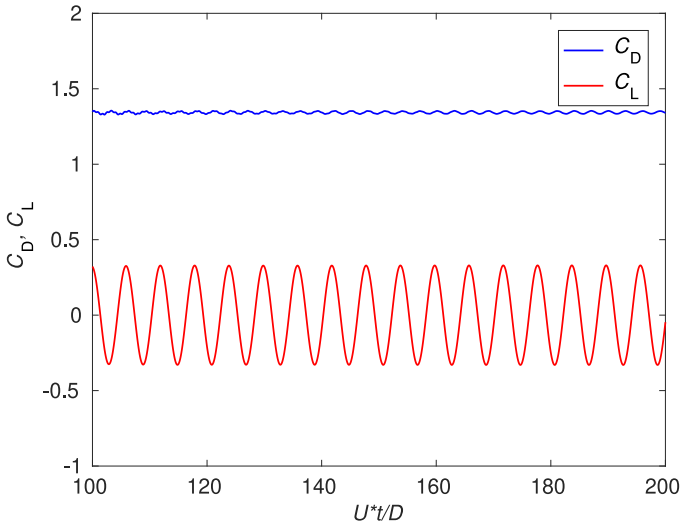


Fig. 10. Computed temporal variation of the drag (C_D) and lift (C_L) coefficients for the flow over a stationary cylinder with the mesh resolution $h/D = 1/64$ at $Re = 100$.

Fig. 10 shows the time evolution of the predicted drag (C_D) and lift (C_L) as a function of time. The time-averaged drag coefficient with the fluctuation, the RMS-averaged lift coefficient, and the resulting Strouhal number are presented in Table 2. The results reported previously in the literature are also included for comparison, including those obtained from the experiment, body-fitted, different kinds of immersed boundary, immersed interface, and cut-cell methods. For completeness, the size of the computational domain and mesh resolution are also summarised in Table 2. It can be seen that the present results obtained from the four sets of meshes are convergent and they are in good agreement with those results obtained from previous numerical methods. When compared to the mesh size for other immersed boundary methods, a reasonable good result can be obtained with a relatively coarse mesh, mainly due to the sharp-interface representation in the cut-cell method. When compared to other cut-cell methods, there is no additional special treatment for the numerical instability for small cut-cells in the proposed method, which would be more efficient when dealing with moving body problems [46].

Fig. 11 shows the spatial convergence study for the drag and lift coefficients for the three grids ($h/D = 1/16, 1/32, 1/64$), where the solution for the finest mesh is taken as a reference to calculate the errors. It can be seen that the convergence rate is close to second-order for the finer grids. When the mesh is refined in the vicinity of the cylinder, the predicted drag and lift coefficients and the Strouhal number are in very close agreement with the results obtained from the body-fitted method and experimentally measured frequency, which demonstrate the accuracy of the present cut-cell method to simulate the vortex shedding for flow over a circular cylinder.

Table 2
Comparison between experimental results, other simulations from immersed boundary methods (IBM), immersed interface method (IIM), cut-cell method, and the present study for flow over a cylinder at $Re = 100$.

Study	Method	$L/D \times W/D$	D/h	C_D	C_L	St
Williamson [79]	experiment	–	–	–	–	0.164
Liu et al. [80]	body-fitted	–	–	1.35 ± 0.012	0.339	0.165
Lai and Peskin [81]	IBM-smooth interface	27×27	128	1.447	0.33	0.165
Kim et al. [12]	IBM-mass source	70×100	30	1.33	0.32	0.165
Tseng and Ferziger [13]	IBM-ghost cell	32×16	72	1.42	0.29	0.164
Uhlmann [9]	IBM-direct forcing	40×40	38.4	1.45 ± 0.011	0.339	0.169
Mittal et al. [17]	IBM-sharp interface	40×40	66.7	1.35	–	0.166
Berthelsen and Faltinsen [18]	IBM-ghost cell 1D	50×30	128	1.38 ± 0.01	0.34	0.169
Yang and Stern [82]	IBM-smooth delta	–	25	1.393	0.335	0.165
Abdol Azis et al. [19]	IBM-unstructured	32×32	60	1.371	–	0.166
Linnick and Fasel [77]	IIM	$- \times 43$	–	1.34 ± 0.009	0.333	0.166
Cheny and Botella [78]	cut-cell	23×12	100	1.31 ± 0.009	0.349	0.170
Hartmann et al. [30]	cut-cell	64×30	32	1.35	0.337	0.165
Muralidharan and Menon [31]	cut-cell	30×30	80	1.36	–	0.167
Present study	cut-cell	60×40	16	1.294 ± 0.005	0.238	0.17
Present study	cut-cell	60×40	32	1.30 ± 0.008	0.293	0.17
Present study	cut-cell	60×40	64	1.33 ± 0.009	0.320	0.17
Present study	cut-cell	60×40	128	1.34 ± 0.009	0.329	0.165

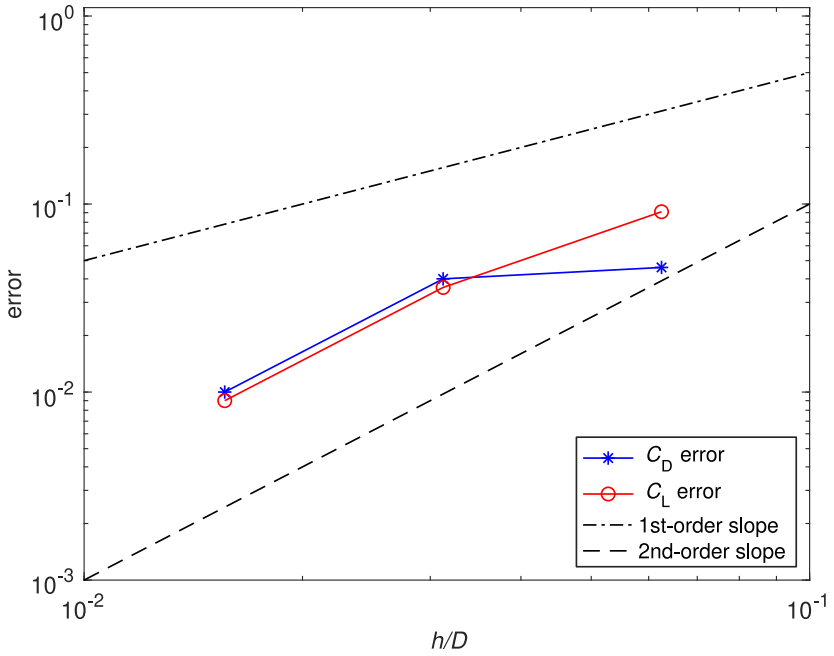


Fig. 11. Error for the drag and lift coefficients as a function of grid resolution for the flow over a circular cylinder at $Re = 100$. Lines for first-order and second-order behaviour are also plotted for reference.

3.3. 2D flow past an airfoil

After successfully validating the flow over a circular cylinder, another benchmark case for flow over a NACA0012 airfoil is considered here. The leading edge of the airfoil has a small radius compared to the chord length and also there is a sharp corner for the trailing edge, which is quite challenging for some interpolations in immersed boundary methods and some cut-cell methods due to the additional treatment for small cut-cells. In this study, a computational

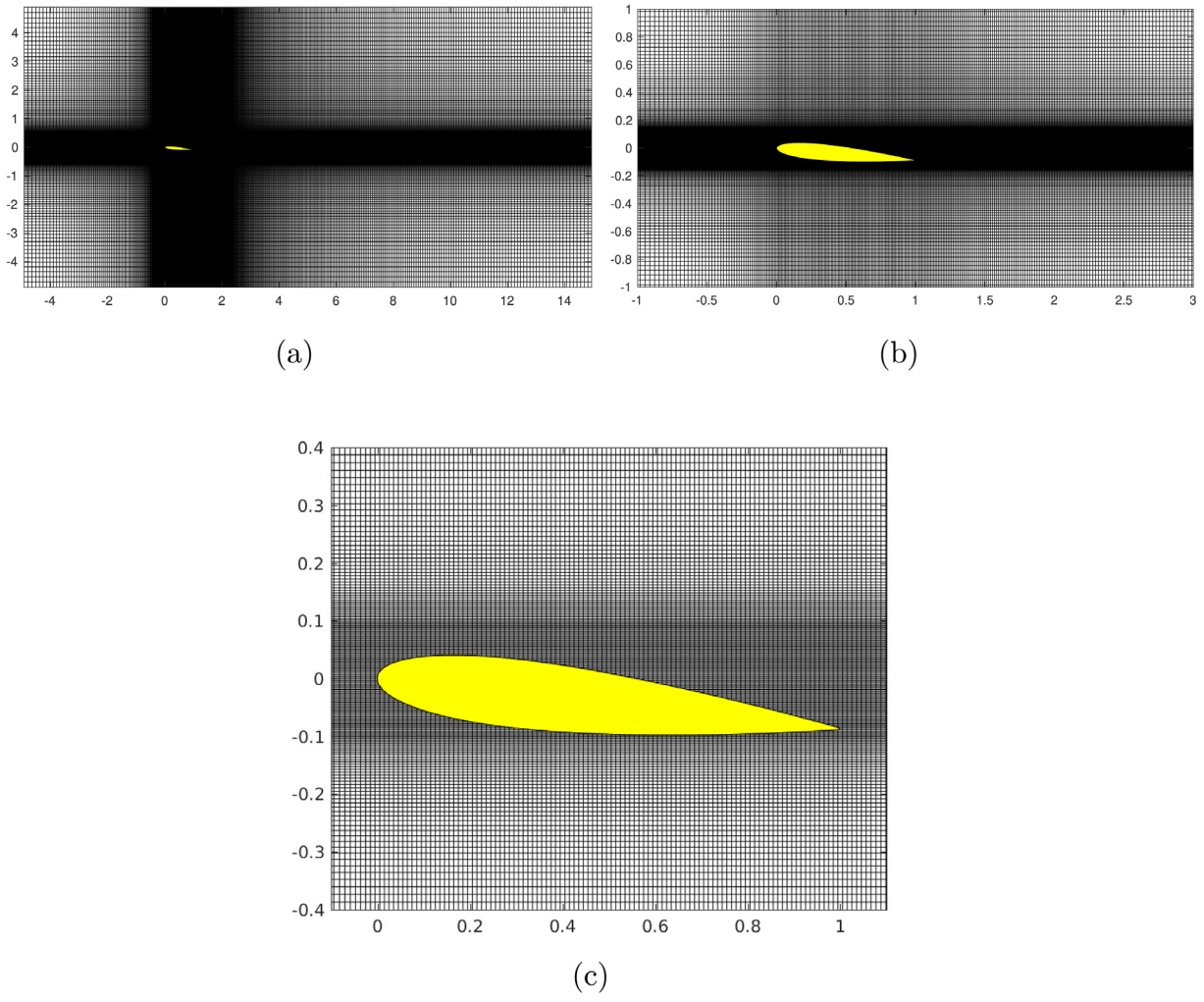


Fig. 12. Schematic of the computational setup for the flow past a NACA0012 airfoil at $Re = 1000$. The whole computational domain (a), local region near the airfoil (b) and zoom-in view mesh in the vicinity of the airfoil (c) are shown here.

domain of $[-5c, 15c] \times [-5c, 5c]$, shown in Fig. 12(a), is used and the leading edge of the airfoil is located at $(0, 0)$, where c is the chord length. A non-uniform 500×300 grid is employed here with the uniform minimum mesh ($\Delta x_{\min} = 0.01c$, $\Delta y_{\min} = 0.0025c$) in the vicinity of the airfoil shown in Fig. 12(c). Two angles of attack ($\alpha = 5^\circ$ and $\alpha = 10^\circ$) are studied here for the $Re = \frac{\rho U c}{\mu} = 1000$, based on the incoming velocity magnitude U and the airfoil chord length c.

Fig. 13 shows the predicted vorticity and normalised pressure for the flow over the airfoil at $\alpha = 5^\circ$ angle of attack. It can be seen that the pre-vortex regime is observed for this angle of attack as negative and positive vorticity is observed on the upper and lower side of the airfoil, respectively. This is in close agreement with those results obtained by the body-fitted [83] and sharp interface immersed boundary [84] methods. The normalised pressure distribution also indicates that higher pressure is obtained at the bottom surface of the airfoil and lower pressure is obtained at the upper surface, which is driven force to produce the lift.

When the angle of attack increases to $\alpha = 10^\circ$, it can be seen from Fig. 14 that periodic vortex shedding is generated from the trailing edge of the airfoil, with alternating negative and positive vorticities moving upwards and downwards, respectively. The vortex shedding pattern is also similar to those reported in the literature [83,84].

Fig. 15 shows the computed temporal variation of the drag and lift coefficients for the flow over a NACA0012 airfoil at $Re = 1000$ for two different angles of attack α. It can be seen that both drag and lift coefficients increase

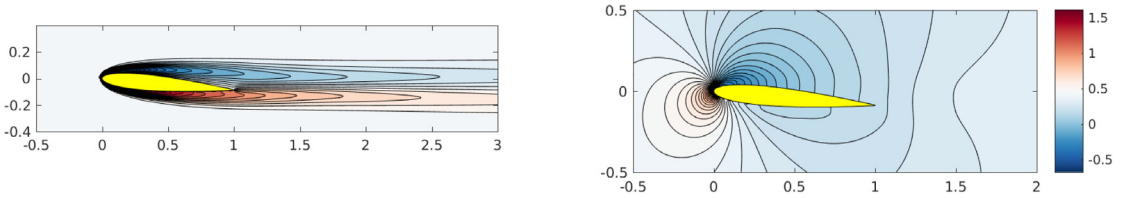


Fig. 13. Contour plots for the vorticity (a) and normalised pressure (b) fields for the flow over a NACA0012 airfoil at $Re = 1000$ and $\alpha = 5°$.

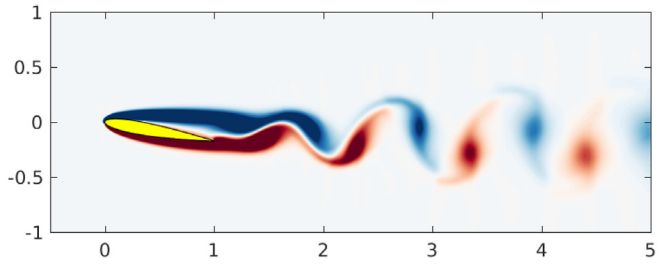


Fig. 14. Contour plots for the vorticity of the flow over a NACA0012 airfoil at $Re = 1000$ and $\alpha = 10°$.

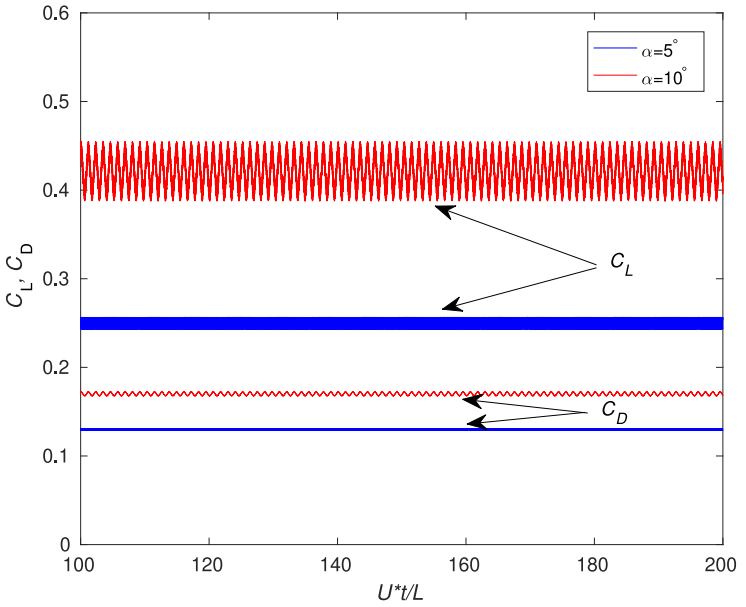


Fig. 15. Computed temporal variation of the drag (C_{D}) and lift (C_{L}) coefficients for the flow over a NACA0012 airfoil at $Re = 1000$ for different angles of attack α.

when the angle of attack increases. For the pre-vortex shedding at $\alpha = 5°$, the drag is nearly constant and there is only a small fluctuation for the lift coefficient. During vortex shedding at $\alpha = 10°$, periodic flow downstream the airfoil can be observed with larger fluctuation for the lift and small amplitude oscillation for the drag. The time history of the drag and lift coefficients including fluctuation at $\alpha = 10°$ is in very good agreement with the body-fitted finite element simulation results shown in Fig. 7 of [85], which confirms the accuracy of the present cut-cell method.

Table 3
Comparison between other simulations (body-fitted and immersed boundary methods), and the present study for flow over a NACA0012 airfoil at $Re = 1000$.

Angle of attack	Method	5°			10°		
At $Re = 1000$		C_D	C_L	St	C_D	C_L	St
Mittal and Tezduyar [85]	body-fitted (FEM)	–	–	–	0.166	0.425	0.85
Kurtulus [83]	body-fitted (FLUENT)	0.13	0.25	–	0.163	0.42	0.876
Menon and Mittal [84]	IBM-sharp interface	0.11	0.25	–	0.17	0.44	0.92
Present study	cut-cell	0.13	0.25	–	0.17	0.42	0.875

The time-averaged drag and lift coefficients, and the resulting Strouhal number are presented in Table 3. The results reported previously in the literature are also included in the comparison, including those obtained from the body-fitted finite element method [85], the commercial software FLUENT [83], and the sharp interface immersed boundary method [84]. It shows that the predicted drag, lift, and Strouhal numbers are in close agreement with the previous studies for both pre-vortex and vortex shedding flow phenomena.

3.4. 3D flow past a sphere

For three-dimensional flow, the laminar flow over a sphere is considered here, which is a conical and challenging benchmark for Cartesian grid methods. Some of the previous studies have used different numerical methods to examine this problem, such as the boundary-fitted method [1,86,87], different immersed boundary method [11,12,14,17], and cut-cell method [30,31]. Similar to the 2D circular cylinder case, the flow phenomenon depends on the Reynolds number and the flow pattern changes from steady motion to unsteady motion beyond about $Re = 280$ [17].

In order to capture both steady and unsteady vortex shedding, two cases with $Re = 100$ and $Re = 300$, based on the diameter of the sphere D and the inlet velocity U, are considered here. The computational domain of $15D \times 15D \times 15D$ is discretised by two non-uniform grids ($180 \times 120 \times 120$ and $360 \times 240 \times 240$) with minimum meshes $h = \Delta x_{\min} = \Delta y_{\min} = \Delta z_{\min}$ ($h/D = 1/25, 1/50$) in the vicinity of the sphere. A fixed time step $\Delta t = 0.005D/U$ is used and the simulations are run to $200D/U$.

Fig. 16 shows the computed vortical structures and the streamlines obtained in the simulations, in which the λ_2 method [88] is employed to identify the vortices. At $Re = 100$ shown in Fig. 16(a), it can be seen that a steady axisymmetric vortex ring is observed just downstream of the sphere and there is flow separation in the wake region with an almost uniform pattern of the streamlines. When the Reynolds number increases to 300 shown in Fig. 16(b), the flow is non-axisymmetric anymore and unsteady vortex shedding is developed. Periodic hairpin vortices are developed with stronger flow separation being observed from the streamlines in the wake of the sphere. These phenomena are consistent with those results reported in the literature.

In order to quantitatively compare with different methods in the literature, the time-averaged drag coefficients are compared in Table 4 for $Re = 100$ and the time-averaged drag and lift coefficients, and the resulting Strouhal number are presented in Table 5 for $Re = 300$. It can be seen that mesh convergent results are obtained with the present cut-cell method, which are consistent with those results obtained by the body-fitted, immersed boundary, and cut-cell method. It is also noted that a reasonably good result can be obtained by the present method using a relatively coarse mesh to capture the flow dynamics over a 3D sphere.

3.5. 3D flow in a pipe

After validating the three-dimensional external flow over a bluff body, the internal flow inside a 3D pipe is further investigated. Only laminar flow is considered here as we know the analytical solution for the parabolic velocity profile $u = u_{\max}\left[1 - 4(r/D)^2\right]$ and also the friction factor as a function of the Reynolds number as $f = 64/Re$, where $u_{\max}$ is the maximum centreline velocity and r is the radial distance from the pipe centre. The computational domain of $2\pi D \times 1.1D \times 1.1D$ is used in the streamwise, vertical, and spanwise direction with a uniform mesh $300 \times 110 \times 110$ of $\Delta y = \Delta z = 0.01D$ to resolve the pipe boundary.

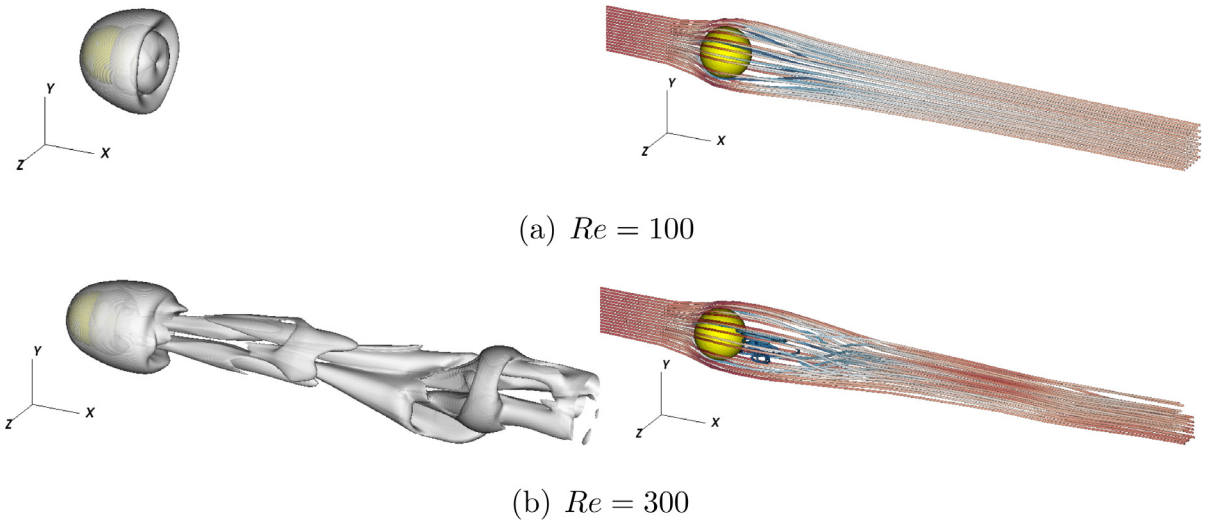


Fig. 16. Computed vortical structures (identified by λ_2 method) and the streamlines for flow over a stationary sphere at $Re = 100$ (a) and $Re = 300$ (b).

Table 4
Comparison between boundary-fitted simulations, immersed boundary methods, and the present study for flow over a sphere at $Re = 100$.

Study at $Re = 100$	Method	D/h	C_D
Fornberg [86]	boundary-fitted	–	1.0852
Fadlun et al. [11]	IBM-direct forcing	–	1.0794
Kim et al. [12]	IBM-mass source	40	1.087
Gilmanov et al. [14]	IBM-sharp interface	40	1.153
Hartmann et al. [30]	cut-cell	32	1.083
Present study	cut-cell	25	1.0737
Present study	cut-cell	50	1.0843

Table 5
Comparison between boundary-fitted simulations, immersed boundary methods, and the present study for flow over a sphere at $Re = 300$.

Study at $Re = 300$	Method	D/h	C_D	C_L	St
Johnson and Patel [1]	boundary-fitted	–	0.656	0.069	0.137
Constantinescu and Squires [87]	boundary-fitted	–	0.655	0.065	0.136
Kim et al. [12]	IBM-mass source	40	0.657	0.067	0.134
Mittal et al. [17]	IBM-sharp interface	–	0.66	–	0.135
Hartmann et al. [30]	cut-cell	32	0.657	0.069	0.135
Present study	cut-cell	25	0.6532	0.062	0.131
Present study	cut-cell	50	0.6524	0.067	0.134

Fig. 17 shows the predicted velocity distribution along a cross-section together with the mesh for the cut-cell treatment. The pressure distribution along the streamwise plane is also shown. It can be seen that the velocity is higher in the centre and gradually decreases towards the pipe wall due to the no-slip boundary condition there. The pressure is higher at the inlet and lower at the outlet and the pressure gradient is balanced by the wall shear stress.

In order the make a quantitative comparison, the velocity profile is compared with the analytical solution in Fig. 18(a), in which a very good agreement is obtained by the cut-cell method. In addition, several simulations for different Reynolds numbers at $Re = 10, 100, 1000$ have been performed and the computed friction factors are shown in Fig. 18(b), which agree well with the analytical formula for the friction factor for laminar flow in a pipe.

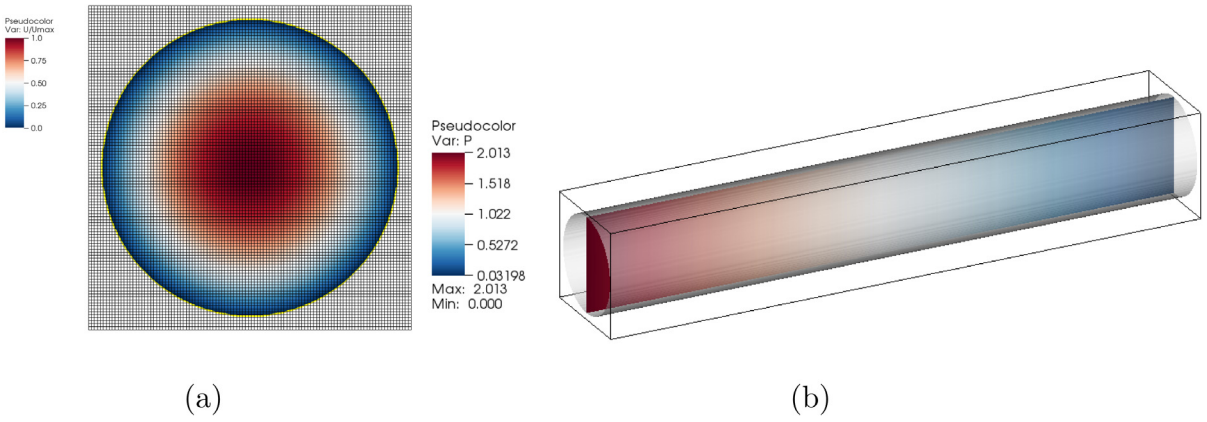


Fig. 17. The computed velocity distribution along a cross section and the pressure distribution along the streamwise plane. The mesh to deal with the cut-cells (a) and also the computational domain (b) are shown.

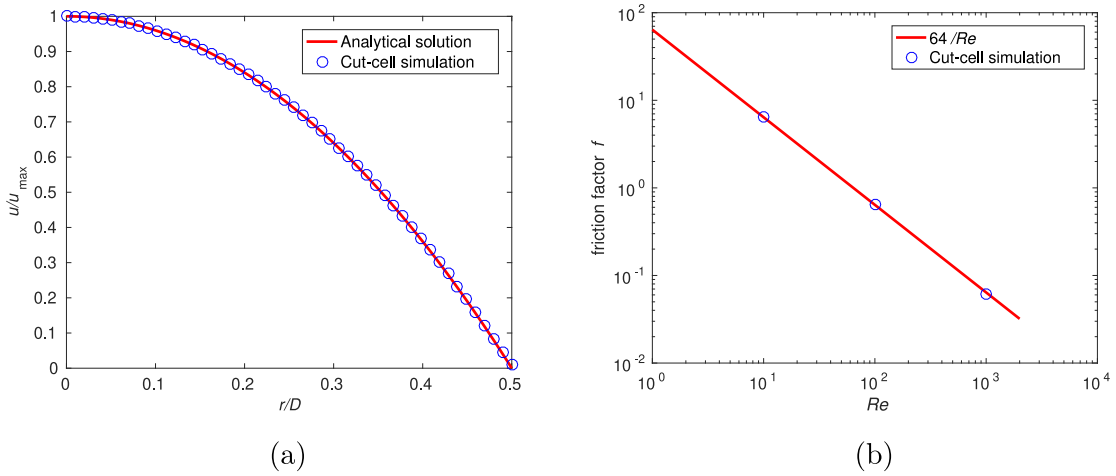


Fig. 18. The computed velocity profile (a) and friction factor (b) for laminar flow in a pipe.

3.6. 3D flow past a heart sculpture

Finally, three-dimensional flow over a complex geometry, i.e. a heart sculpture, is shown here. The same setup and mesh used for the sphere case is employed here, whereas the sphere is replaced by the geometry of the heart surface taken as [89]

$$(\frac{9}{4}x^2 + y^2 + z^2 - 1)^3 - \frac{9}{80}x^2z^3 - y^2z^3 = 0 \tag{18}$$

The computed vortical structure and streamlines for the flow over the heart sculpture at $Re = 100$ are shown in Fig. 19. It can be seen that the steady flow structure is observed, similar to the flow over a sphere at the same Reynolds number. However, the vortex is not axisymmetric but follows the shape of the heart. More chaotic flow separation can be observed from the streamline in the wake of the heart, demonstrating stronger mixing when the bluff body changed from the sphere to the heart.

4. Conclusions

A versatile three-dimensional Cartesian cut-cell method has been developed for the simulation of incompressible viscous flow over complex geometries in the code Xdolphin3D. Both 2D and 3D cut-cell configurations are presented

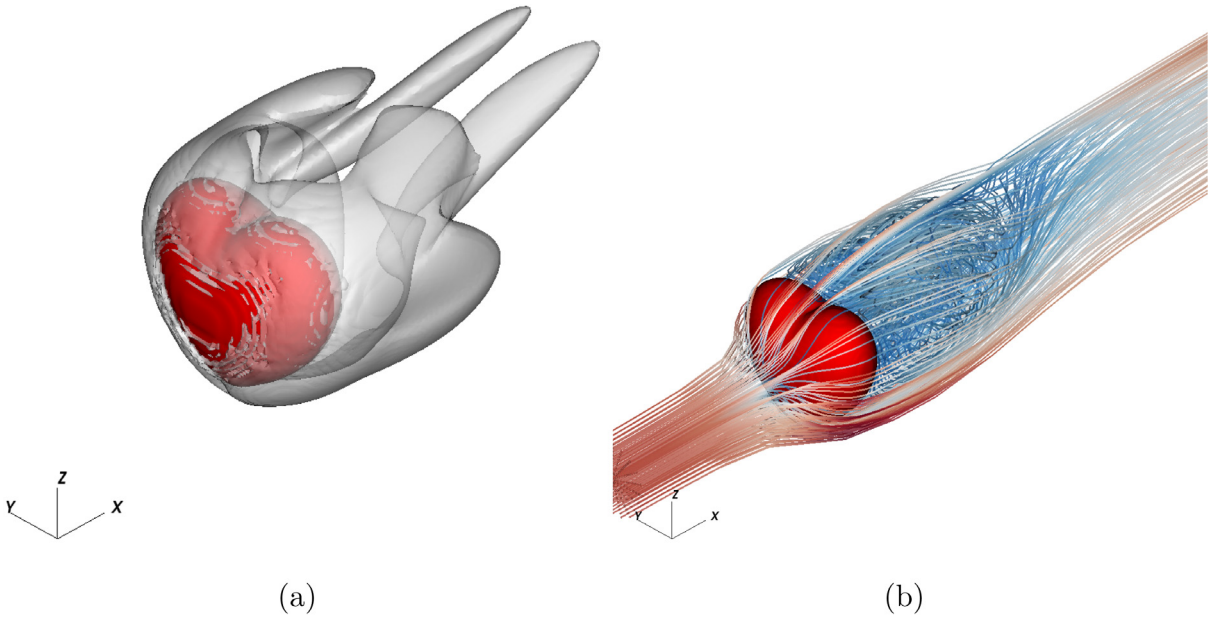


Fig. 19. The computed vortical structure (a) and streamlines (b) for flow over a heart sculpture at $Re = 100$.

in detail on a staggered grid, in which a consistent mass and momentum transport is used for the finite volume discretisation in both fluid and cut cells. Contrary to existing cut-cell methods, an implicit time stepping scheme is employed in the present study, which avoids numerical instability without any additional small-cell treatment. Different from most immersed boundary methods, strict conservation of the mass and momentum is enforced through the PISO algorithm for the pressure–velocity coupling, together with cut-cell information for the momentum and continuity equations.

In order to validate the cut-cell method, several canonical benchmark cases in both 2D and 3D have been investigated. First, both spatial and temporal convergence tests are performed for Taylor–Green vortices to check the performance of the cut-cell method. A 2D laminar flow over a circular cylinder is first computed, with and without vortex shedding. The wake characteristics, drag and lift coefficients, and Strouhal number for vortex shedding have been shown with different mesh resolutions. A 2D flow over a more challenging NACA0012 airfoil is also studied, and the flow pattern and forces are shown, which are similar to those obtained from body-fitted methods. Then 3D external flow over a sphere is considered at two Reynolds numbers, with different vortical structures and streamlines being observed in the simulation. For the cylinder, airfoil and sphere cases, it is worth mentioning that all the computed drag and lift coefficients and Strouhal numbers are compared with experimental measurements, boundary-fitted, various immersed boundary methods, and other cut-cell methods, with a close agreement being obtained with those results reported in the literature. Furthermore, a 3D internal pipe flow is studied and the computed velocity profile and friction factors are compared with the analytical solution. Finally, the ability of the present solver to deal with more complex geometry, a heart sculpture, is demonstrated with a different vortical structure and wake flow separation.

It is worth mentioning that the stability analysis is important to understand the cut-cell schemes at the theoretical level. However, the hypothesis of the Fourier method cannot be rigorously fulfilled in the present study (as there is assumption for cyclic boundary conditions and constant discretisation coefficients) [69], so we will focus this theoretical investigation in our future work by finding an ideal case to simplify the problem. All the benchmark cases in this paper are for low-Reynolds number flows. Future study will also focus on the cut-cell method for high-Reynolds number incompressible [32] and compressible [90] flow, which is an area of interest in engineering applications.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was financially supported by the Engineering and Physical Sciences Research Council (EPSRC), United Kingdom through grant (EP/V040235/1), the Royal Society Newton Advanced Fellowship (NAF/R1/201156) and International Exchanges Award (IEC/NSFC/211143, IES/R2/202095). Constructive comments from two anonymous reviewers for the improvement of the manuscript are gratefully acknowledged.

References

[1] T.A. Johnson, V.C. Patel, Flow past a sphere up to a Reynolds number of 300, J. Fluid Mech. 378 (1999) 19–70.
[2] Z. Cheng, J. Li, C.Y. Loh, L.-S. Luo, An exactly force-balanced boundary-conforming arbitrary-Lagrangian-Eulerian method for interfacial dynamics, J. Comput. Phys. 408 (2020) 109237.
[3] F. Sotiropoulos, X. Yang, Immersed boundary methods for simulating fluid–structure interaction, Prog. Aerosp. Sci. 65 (2014) 1–21.
[4] S. Roy, A. De, E. Balaras, Immersed Boundary Method: Development and Applications, Springer, Singapore, 2020.
[5] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[6] B.E. Griffith, N.A. Patankar, Immersed methods for fluid–structure interaction, Annu. Rev. Fluid Mech. 52 (2020) 421–448.
[7] D.M. Ingram, D.M. Causon, C.G. Mingham, Developments in Cartesian cut cell methods, Math. Comput. Simulation 61 (2003) 561–572.
[8] C. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (1972) 252–271.
[9] M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows, J. Comput. Phys. 209 (2005) 448–476.
[10] J. Mohd-Yusof, Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries, CTR, Annu. Res. Brief (1997) 317–327.
[11] E.A. 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.
[12] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, J. Comput. Phys. 171 (2001) 132–150.
[13] Y.H. Tseng, J.H. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comput. Phys. 192 (2003) 593–623.
[14] A. Gilmanov, F. Sotiropoulos, E. Balaras, A general reconstruction algorithm for simulating flows with complex 3D immersed boundaries on Cartesian grids, J. Comput. Phys. 191 (2003) 660–669.
[15] E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations, Comput. & Fluids 33 (2004) 375–404.
[16] J. Yang, E. Balaras, An embedded-boundary formulation for large-eddy simulation of turbulent flows interacting with moving boundaries, J. Comput. Phys. 215 (2006) 12–40.
[17] R. Mittal, H. Dong, M. Bozkurttas, F.M. Najjar, A. Vargas, A. von Loebbecke, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, J. Comput. Phys. 227 (2008) 4825–4852.
[18] P.A. Berthelsen, O.M. Faltinsen, A local directional ghost cell approach for incompressible viscous flow problems with irregular boundaries, J. Comput. Phys. 227 (2008) 4354–4397.
[19] M.H. Abdol Azis, F. Evrard, B. van Wachem, An immersed boundary method for incompressible flows in complex domains, J. Comput. Phys. 378 (2019) 770–795.
[20] K. Taira, T. Colonius, The immersed boundary method: A projection approach, J. Comput. Phys. 225 (2007) 2118–2137.
[21] J.H. Seo, R. Mittal, A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations, J. Comput. Phys. 230 (2011) 7347–7363.
[22] M. Berger, A. Giuliani, A state redistribution algorithm for finite volume schemes on cut cell meshes, J. Comput. Phys. 428 (2021).
[23] P.T. Brady, D. Livescu, Foundations for high-order, conservative cut-cell methods: Stable discretizations on degenerate meshes, J. Comput. Phys. 426 (2021).
[24] T. Ye, R. Mittal, H.S. Udaykumar, W. Shyy, An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries, J. Comput. Phys. 156 (1999) 209–240.
[25] P.G. Tucker, Z. Pan, A Cartesian cut cell method for incompressible viscous flow, Appl. Math. Model. 24 (2000) 591–606.
[26] M.H. Chung, Cartesian cut cell approach for simulating incompressible flows with rigid bodies of arbitrary shape, Comput. & Fluids 35 (2006) 607–623.
[27] H. Ji, F.S. Lien, E. Yee, Numerical simulation of detonation using an adaptive Cartesian cut-cell method combined with a cell-merging technique, Comput. & Fluids 39 (2010) 1041–1057.

[28] R. Yamashita, L. Wutschitz, N. Nikiforakis, A full-field simulation methodology for sonic boom modeling on adaptive Cartesian cut-cell meshes, J. Comput. Phys. 408 (2020).
[29] M.P. Kirkpatrick, S.W. Armfield, J.H. Kent, A representation of curved boundaries for the solution of the Navier–Stokes equations on a staggered three-dimensional Cartesian grid, J. Comput. Phys. 184 (2003) 1–36.
[30] D. Hartmann, M. Meinke, W. Schroder, A strictly conservative Cartesian cut-cell method for compressible viscous flows on adaptive grids, Comput. Methods Appl. Mech. Engrg. 200 (2011) 1038–1052.
[31] B. Muralidharan, S. Menon, A high-order adaptive Cartesian cut-cell method for simulation of compressible viscous flow over immersed bodies, J. Comput. Phys. 321 (2016) 342–368.
[32] M. Meyer, A. Devesa, S. Hickel, X.Y. Hu, N.A. Adams, A conservative immersed interface method for large-eddy simulation of incompressible flows, J. Comput. Phys. 229 (2010) 6300–6317.
[33] F. Gao, D.M. Ingram, D.M. Causon, C.G. Mingham, The development of a Cartesian cut cell method for incompressible viscous flows, Internat. J. Numer. Methods Fluids 54 (2007) 1033–1053.
[34] W. Bai, C.G. Mingham, D.M. Causon, L. Qian, Finite volume simulation of viscous free surface waves using the Cartesian cut cell approach, Internat. J. Numer. Methods Fluids 63 (2010) 69–95.
[35] T.Q. Li, P. Troch, J. De Rouck, Wave overtopping over a sea dike, J. Comput. Phys. 198 (2004) 686–726.
[36] Z. Xie, A two-phase flow model for three-dimensional breaking waves over complex topography, Proc. R. Soc. A Math. Phys. Eng. Sci. 471 (2015) 20150101.
[37] J.-Y. Lin, Y. Shen, H. Ding, N.-S. Liu, X.-Y. Lu, Simulation of compressible two-phase flows with topology change of fluid–fluid interface by a robust cut-cell method, J. Comput. Phys. 328 (2017) 140–159.
[38] S.T. Dang, E.A. Meese, J.C. Morud, S.T. Johansen, Numerical approach for generic three-phase flow based on cut-cell and ghost fluid methods, Internat. J. Numer. Methods Fluids 91 (2019) 419–447.
[39] Z. Xie, T. Stoesser, S. Yan, Q. Ma, P. Lin, A Cartesian cut-cell based multiphase flow model for large-eddy simulation of three-dimensional wave-structure interaction, Comput. & Fluids 213 (2020) 104747.
[40] H.S. Udaykumar, W. Shyy, M.M. Rao, ELAFINT: A Mixed Eulerian-Lagrangian method for fluid flows with complex and moving boundaries, Internat. J. Numer. Methods Fluids 22 (1996) 691–712.
[41] W.P. Bennett, N. Nikiforakis, R. Klein, A moving boundary flux stabilization method for Cartesian cut-cell grids using directional operator splitting, J. Comput. Phys. 368 (2018) 333–358.
[42] M. Meinke, L. Schneiders, C. Gunther, W. Schroder, A cut-cell method for sharp moving boundaries in Cartesian grids, Comput. & Fluids 85 (2013) 135–142.
[43] L. Schneiders, D. Hartmann, M. Meinke, W. Schroder, An accurate moving boundary formulation in cut-cell methods, J. Comput. Phys. 235 (2013) 786–809.
[44] G. Dechriste, L. Mieussens, A Cartesian cut cell method for rarefied flow simulations around moving obstacles, J. Comput. Phys. 314 (2016) 465–488.
[45] B. Muralidharan, S. Menon, Simulation of moving boundaries interacting with compressible reacting flows using a second-order adaptive Cartesian cut-cell method, J. Comput. Phys. 357 (2018) 230–262.
[46] Z. Xie, P. Lin, T. Stoesser, A conservative and consistent implicit Cartesian cut-cell method for moving geometries with reduced spurious pressure oscillations, J. Comput. Phys. 459 (2022) 111124.
[47] H.S. Udaykumar, H.C. Kan, W. Shyy, R. Tran-Son-Tay, Multiphase dynamics in arbitrary geometries on fixed Cartesian grids, J. Comput. Phys. 137 (1997) 366–405.
[48] K.M.T. Kleefsman, G. Fekken, A.E.P. Veldman, B. Iwanowski, B. Buchner, A volume-of-fluid based simulation method for wave impact problems, J. Comput. Phys. 206 (2005) 363–393.
[49] M.H. Chung, An adaptive Cartesian cut-cell/level-set method to simulate incompressible two-phase flows with embedded moving solid boundaries, Comput. & Fluids 71 (2013) 469–486.
[50] L. Qian, D.M. Causon, C.G. Mingham, D.M. Ingram, A free-surface capturing method for two fluid flows with moving bodies, Proc. R. Soc. A Math. Phys. Eng. Sci. 462 (2006) 21–42.
[51] W. Benguigui, A. Doradoux, J. Lavieville, S. Mimouni, E. Longatte, A discrete forcing method dedicated to moving bodies in two-phase flow, Internat. J. Numer. Methods Fluids 88 (2018) 315–333.
[52] Z. Xie, T. Stoesser, A three-dimensional Cartesian cut-cell/volume-of-fluid method for two-phase flows with moving bodies, J. Comput. Phys. 416 (2020) 109536.
[53] Q. Chen, J. Zang, A.S. Dimakopoulos, D.M. Kelly, C.J.K. Williams, A Cartesian cut cell based two-way strong fluid-solid coupling algorithm for 2d floating bodies, J. Fluids Struct. 62 (2016) 252–271.
[54] L. Schneiders, C. Gunther, M. Meinke, W. Schroder, An efficient conservative cut-cell method for rigid bodies interacting with viscous compressible flows, J. Comput. Phys. 311 (2016) 62–86.
[55] V. Pasquariello, G. Hammerl, F. Örley, S. Hickel, C. Danowski, A. Popp, W.A. Wall, N.A. Adams, A cut-cell finite volume finite element coupling approach for fluid-structure interaction in compressible flow, J. Comput. Phys. 307 (2016) 670–695.
[56] H. Ji, F.-S. Lien, E. Yee, A robust and efficient hybrid cut-cell/ghost-cell method with adaptive mesh refinement for moving boundaries on irregular domains, Comput. Methods Appl. Mech. Engrg. 198 (2008) 432–448.
[57] K. Fidkowski, D. Darmofal, A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier–Stokes equations, J. Comput. Phys. 225 (2007) 1653–1672.
[58] S. Claus, P. Kerfriden, A CutFEM method for two-phase flow problems, Comput. Methods Appl. Mech. Engrg. 348 (2019) 185–206.
[59] H.P. Bui, S. Tomar, S.P. Bordas, Corotational cut finite element method for real-time surgical simulation: Application to needle insertion simulation, Comput. Methods Appl. Mech. Engrg. 345 (2019) 183–211.

[60] Z. Han, S.K. Stoter, C.-T. Wu, C. Cheng, A. Mantzaflaris, S.G. Mogilevskaya, D. Schillinger, Consistent discretization of higher-order interface models for thin layers and elastic material surfaces, enabled by isogeometric cut-cell methods, Comput. Methods Appl. Mech. Engrg. 350 (2019) 245–267.
[61] C. Ager, B. Schott, M. Winter, W. Wall, A Nitsche-based cut finite element method for the coupling of incompressible fluid flow with poroelasticity, Comput. Methods Appl. Mech. Engrg. 351 (2019) 253–280.
[62] H.-G. Bui, D. Schillinger, G. Meschke, Efficient cut-cell quadrature based on moment fitting for materially nonlinear analysis, Comput. Methods Appl. Mech. Engrg. 366 (2020) 113050.
[63] S. Pande, P. Papadopoulos, I. Babuška, A cut-cell finite element method for Poisson's equation on arbitrary planar domains, Comput. Methods Appl. Mech. Engrg. 383 (2021) 113875.
[64] D.M. Causon, D.M. Ingram, C.G. Mingham, G. Yang, R.V. Pearson, Calculation of shallow water flows using a Cartesian cut cell approach, Adv. Water Resour. 23 (2000) 545–562.
[65] Z. Xie, Numerical study of breaking waves by a two-phase flow model, Internat. J. Numer. Methods Fluids 70 (2012) 246–268.
[66] R.I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, J. Comput. Phys. 62 (1986) 40–65.
[67] H.A. van der Vorst, Bi-CGSTAB - a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear-systems, SIAM J. Sci. Stat. Comput. 13 (1992) 631–644.
[68] B. Xie, F. Xiao, Accurate and robust PISO algorithm on hybrid unstructured grids using the multimoment finite volume method, Numer. Heat Transfer B 71 (2017) 146–172.
[69] C.M. Venier, C.I. Pairetti, S.M. Damian, N.M. Nigro, On the stability analysis of the PISO algorithm on collocated grids, Comput. & Fluids 147 (2017) 25–40.
[70] W.E. Lorensen, H.E. Cline, Marching cubes: A high resolution 3D surface construction algorithm, Comput. Graph. 21 (1987) 163–169.
[71] M. Berger, A note on the stability of cut cells and cell merging, Appl. Numer. Math. 96 (2015) 180–186.
[72] S. May, M. Berger, An explicit implicit scheme for cut cells in embedded boundary meshes, J. Sci. Comput. 71 (2016) 919–943.
[73] E. Burman, S. Claus, P. Hansbo, M. Larson, A. Massing, CutFEM: Discretizing geometry and partial differential equations, Internat. J. Numer. Methods Engrg. 104 (2015) 472–501.
[74] D.J. Tritton, Experiments on the flow past a circular cylinder at low Reynolds numbers, J. Fluid Mech. 6 (1959) 547–567.
[75] M. Coutanceau, R. Bouard, Experimental determination of the main features of the viscous flow in the wake of a circular cylinder in uniform translation. part 1. steady flow, J. Fluid Mech. 79 (1977) 231–256.
[76] B. Fornberg, A numerical study of steady viscous flow past a circular cylinder, J. Fluid Mech. 98 (1980) 819–855.
[77] M.N. Linnick, H.F. Fasel, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, J. Comput. Phys. 204 (2005) 157–192.
[78] Y. Cheny, O. Botella, The LS-STAG method: A new immersed boundary/level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties, J. Comput. Phys. 229 (2010) 1043–1076.
[79] 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.
[80] C. Liu, X. Zheng, C. Sung, Preconditioned multigrid methods for unsteady incompressible flows, J. Comput. Phys. 139 (1998) 35–57.
[81] M.-C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705–719.
[82] J. Yang, F. Stern, Sharp interface immersed-boundary/level-set method for wave-body interactions, J. Comput. Phys. 228 (2009) 6590–6616.
[83] D.F. Kurtulus, On the unsteady behavior of the flow around, NACA 0012 airfoil with steady external conditions at Re=1000, Int. J. Micro Air Veh. 7 (2015) 301–326.
[84] K. Menon, R. Mittal, Aerodynamic characteristics of canonical airfoils at low Reynolds numbers, AIAA J. 112 (2020) 977–980.
[85] S. Mittal, T.E. Tezduyar, Massively parallel finite element computation of incompressible flows involving fluid-body interaction, Comput. Methods Appl. Mech. Engrg. 112 (1994) 253–282.
[86] B. Fornberg, Steady viscous flow past a sphere at high Reynolds numbers, J. Fluid Mech. 190 (1988) 471–489.
[87] G.S. Constantinescu, K.D. Squires, LES And DES investigations of turbulent flow over a sphere at Re=10,000, Turbul. Combust. 70 (2003) 267–298.
[88] J. Jeong, F. Hussain, On the identification of a vortex, J. Fluid Mech. 285 (1995) 69–94.
[89] G. Taubin, An accurate algorithm for rasterizing algebraic curves and surfaces, IEEE Comput. Graphics Appl. 14 (1994) 14–23.
[90] M. Berger, M. Aftosmis, Progress towards a Cartesian cut-cell method for viscous compressible flow, in: 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2012.