Journal of Computational Physics 229 (2010) 4476–4500

Contents lists available at ScienceDirect

Journal of Computational Physics

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

A differentially interpolated direct forcing immersed boundary method for predicting incompressible Navier–Stokes equations in time-varying complex geometries

P.H. Chiu [a,1], R.K. Lin [a,2], Tony W.H. Sheu [a,b,c,*]

[a] Department of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, Taiwan, ROC
[b] Taida Institute of Mathematical Sciences (TIMS), National Taiwan University, Taiwan, ROC
[c] Center for Quantum Science and Engineering (CQSE), National Taiwan University, Taiwan, ROC

ARTICLE INFO

Article history:
Received 13 March 2009
Received in revised form 18 January 2010
Accepted 15 February 2010
Available online 1 March 2010

Keywords:
Dispersion-relation-preserving
Cartesian grids
Immersed boundary method
Differential-based
Time-varying domains
Momentum forcing term

ABSTRACT

A dispersion-relation-preserving dual-compact scheme developed in Cartesian grids is applied together with the immersed boundary method to solve the flow equations in irregular and time-varying domains. The artificial momentum forcing term applied at certain points in cells containing fluid and solid allows an imposition of velocity condition to account for the motion of solid body. We develop in this study a differential-based interpolation scheme which can be easily extended to three-dimensional simulation. The results simulated from the proposed immersed boundary method agree well with other numerical and experimental results for the chosen benchmark problems. The accuracy and fidelity of the IB flow solver developed to predict flows with irregular boundaries are therefore demonstrated.

1. Introduction

Flow and heat transfer problems of practical and academic importance often involve geometrically complex bodies which are stationary or in motion in high-Reynolds number fluid flows. These problems can be solved by the classical body-fitted grid-based methods, which discretize the governing equations in a curvilinear coordinate system that conforms to the boundaries, with re-gridding procedures at each time step. Grid generation is, however, entailed with a heavy cost in manpower as well as in computational time. In addition, the quality of grid generation can be a major simulation issue and very often one has to resort to either a multi-block structured or an unstructured approach in order to handle anything but the not-so-complicated geometries.

While methods underlying the unstructured grids generated in a multi-block setting, which can offer a greater flexibility, have been widely used in the past, they still suffer the drawback due to their inapplicability to the multigrid acceleration solver. A completely different methodology, which was firstly proposed by Peskin [1] in 1972, implemented normally in body non-conforming Cartesian grids retains most of the favorable properties of the structured grids but it can also provide a high-level flexibility in handling complex geometries. In short, the immersed boundary (IB) method becomes increasingly

* Corresponding author. Address: Department of Engineering Science and Ocean Engineering, National Taiwan University, Taipei, Taiwan, Republic of China. Tel.: +886 2 33665746; fax: +886 2 23929885.
E-mail addresses: f93525011@ntu.edu.tw (P.H. Chiu), rklin@ntu.edu.tw (R.K. Lin), twhsheu@ntu.edu.tw (T.W.H. Sheu).
[1] Tel.: +886 2 33665726.
[2] Tel.: +886 2 33664429.

0021-9991/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2010.02.013

popular nowadays since generation of grids becomes greatly simplified to tackle the flows with complex stationary or moving boundaries.

The immersed boundary method of Peskin [1] solves the transport equations of fluid flows both inside and outside of the stationary/moving solid objects. The underlying idea that makes this class of methods plausible to simulate flows over solid objects is to match the fluid and particle by the properly distributed force along the boundary. The resulting interaction between the fluid and the immersed solid can be well accounted for either by the prescribed penalty due to Peskin [1] or by the duality introduced in the fictions domain approach of Glowinski et al. [2]. Due to the penalty introduced in the immersed boundary method [1] and the Lagrange multipliers distributed in the fictions domain approach [2], penalty and duality terms make the resulting constrained equations stiff, thereby imposing restriction on the time step [3].

Since the pioneering work of Peskin [1], in the past four decades there have been various research efforts put into the improvement of the Peskin's immersed boundary method for increasing its predicted accuracy to the formally second order spatial accuracy [4], enforcing the volume conservation enclosed by an immersed boundary [5], increasing the resolution across the fluid–solid interface [6], and offering the stability analysis [7]. The immersed interface method (IIM) [8], which differs from the IB method in the way of approximating the singular force appearing in the continuous equations of motion, was developed to resolve the problem of lower (or first-) order accuracy issue for the problem with a sharp interface by incorporating the jump conditions for the Dirac function into the formulation. Both methods suffer, however, a severe time step restriction that is typically much more stringent than the time increment imposed by the explicit schemes for the advective terms [9]. A possible explanation for the unresolved time-step restriction is due to the accumulation of numerical errors for the case investigated at the flow incompressibility condition [10].

Within the context of immersed boundary methods, this class of methods can be broadly categorized into the continuous forcing and the discrete forcing two classes according to the review paper of Mittal and Iaccarino [11]. In the continuous forcing methods, a forcing term is explicitly added to the continuous Navier–Stokes equations before the discretization of them. Typical examples of them are the original method of Peskin [1], and its subsequent extensions by Goldstein et al. [12] and Saiki and Biringen [13] who introduced the feedback forcing to account for the effect of solid body so that the fluid velocity becomes zero at the desired points in the solid. The continuous forcing IB methods have the advantage of being formulated relatively independent of the chosen spatial discretization methods and can, as a result, be implemented in the existing Navier–Stokes solvers with relative ease. In the discrete forcing methods, the forcing term can be either explicitly or implicitly applied to the discretized Navier–Stokes equations [16–18,22,21,19,20,14,15]. In comparison with the first category of IB methods, the discrete forcing methods allow a sharper representation of the immersed boundary [23]. A number of other immersed boundary methods, known as the sharp interface method or the reshaped-cell method [24,17] developed to achieve the second-order or an even higher-order accuracy without momentum forcing [25] and the ghost fluid method (GFM) with momentum forcing [26–28], have been applied to simulate the problems with irregular geometry. One can find other variants of the immersed boundary methods in Cartesian grids, which avoid invoking grid generation, in [6].

This paper is organized as follows. We will describe in Section 2 the working equations for the incompressible Navier–Stokes equations cast in primitive variables. The immersed boundary method implemented with accurate interpolation scheme will be proposed in Section 3 for the evaluation of momentum forcing on the body surface (immersed boundary) or inside the body. Section 4 presents the dispersion-relation-preserving dual-compact scheme for calculating the Navier–Stokes solution in non-staggered grids without producing oscillatory pressures. Section 5 validates the proposed steady and transient Navier–Stokes solvers. For showing the fidelity of the proposed Navier–Stokes flow solver, two problems amenable to analytic solutions are investigated. In order to verify the immersed boundary method, we have investigated several benchmark problems in Section 6. In Section 7, some concluding remarks will be drawn.

2. Governing equations

In this study we will restrict our attention to the incompressible viscous flow equations, which are governed by the following continuity and Navier–Stokes equations

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

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

Given an initial divergence-free velocity field and the prescribed boundary velocity, the chosen primitive variables $\underline{u}$ and p will be sought in a region that is enclosed by the boundary $\partial\Omega$. The condition applied at $\partial\Omega$ with an outward normal vector $\underline{n}$ must satisfy the integral equation given by $\oint_{\partial\Omega} \underline{u} \cdot \underline{n}\, ds = 0$.

Subjected to the constraint Eq. (1), the differential system governing the viscous flow motion is not entirely evolutionary. Momentum equations can be solved along with the divergence-free Eq. (1). The need to satisfy this constraint condition for preserving mass (or incompressibility) can, however, lead to a matrix system which is normally not well-conditioned in cases of high Reynolds number. Under the circumstance, the convergent solutions for $(\underline{u}, p)$ become much difficult to be calculated using a computationally less expensive iterative solver [29]. In certain cases, the peripheral storage for these matrix equations may exceed the available computer power and disk space. These drawbacks prompted the derivation of a scalar equation for p to replace Eq. (1). This class of approaches is subjected to the rigorously derived integral boundary condition

[30] and is, therefore, computationally more difficult to be dealt with. Due to the above drawback existing in the mixed formulation and theoretically no boundary condition is needed for the pressure, Lin [31] proposed the sequential regularization method to solve the time-dependent incompressible Navier–Stokes equations without invoking specification of pressure boundary condition. In the present study, a computationally effective and novel method will be developed in Section 4.2 for the time-dependent Navier–Stokes equations.

3. Immersed boundary method

Fluid flow over a body can exert a force on the no-slip surface and will, in turn, apply a force with the magnitude and direction opposing to the local flow. As a result, the fluid flow can be brought to rest on the body surface [12]. This implies that such an externally applied body-force to the momentum equations at certain points in the flow can simulate the effect of the investigated body in the flow. This idea enables the IB method to mimic the complex body through a suitably introduced artificial force adding to the momentum equations. The resulting momentum equations with an appropriately prescribed forcing vector $\underline{f}$ at certain points in the vicinity of immersed boundary can be written as

$$\frac{\partial \underline{u}}{\partial t} + (\underline{u}\cdot\nabla)\underline{u} = -\nabla p + \frac{1}{Re}\nabla^2\underline{u} + \underline{f} \tag{3}$$

As the name of the discrete-time momentum forcing method indicates, the forcing vector $\underline{f}$ shown above can be directly computed from the following discrete-time momentum vector equation

$$\frac{\underline{u}^{n+1} - u^n}{\Delta t} + (\underline{u}\cdot\nabla)\underline{u} = -\nabla p + \frac{1}{Re}\nabla^2\underline{u} + \underline{f} \tag{4}$$

Note that the momentum forcing vector $\underline{f}$, which is applied only along the body surface or inside the body, should be calculated at each time step. This enables the velocity along an arbitrary immersed boundary surface Ω to be equal to the specified vector function $\underline{V}_\Omega$. When the grid line coincides with the immersed boundary, the momentum forcing term specified in this way can make the velocity magnitude to be equal to $\underline{V}_\Omega$ at this point. For the forcing points inside the body and nearest the boundary, the momentum forcing term needs to be specified so that the normal and tangential velocity components along the boundary can be equal in magnitude but opposite in direction to the velocity components at the corresponding point outside the body. When the boundary surface is not perfectly aligned with the grid plane, the prescribed momentum forcing term will act only on the points nearest the immersed boundary. Interpolation of the value of momentum forcing term is therefore required so that the prescribed forcing term can render the velocity with a magnitude approximately equal to $\underline{V}_\Omega$ at the immersed boundary

$$\underline{f} = -\underline{RHS} + \frac{\underline{V}_\Omega - \underline{u}^n}{\Delta t} \quad \text{on } \Omega \tag{5}$$

In the above, Δt denotes the time increment and $\underline{V}_\Omega$ represents the specified velocity along the immersed boundary. Note that the vector $\underline{RHS}$ shown above is composed of the pressure gradient, convection, and diffusion terms in the momentum equations. For the case with a stationary solid body, along which a no-slip boundary condition is applied, $\underline{V}_\Omega = 0$ will be specified along the boundary.

3.1. Algebraically-interpolated method for the direct forcing term $\underline{f}$

In the immersed boundary method, development of an interpolation scheme to accurately impose the no-slip condition on the immersed boundary is essential since grid lines are not necessarily aligned with the immersed boundary. Some interpolation techniques for evaluating the momentum forcing term on the body surface or inside the body have therefore been proposed by several authors in the past [14–22,32].

The interpolation procedures due to Li and Wang [32] will be described firstly. Let P shown in Fig. 1(a) be the point along the immersed boundary, along which no-slip boundary condition is prescribed. The nearest interior point A has only one neighboring fluid point (node Q). For this case, u_A can be calculated linearly from the nodal value of u_Q and the no-slip velocity at P as

$$u_A = -\frac{h_A}{h_Q}u_Q \tag{6}$$

In the above, h_A and h_Q are denoted as the distances between the points A, Q and P, respectively. In Fig. 1(b), the interior point A has fluid-neighbor points B, C and D. Let u be approximated as $a_1 + a_2x + a_3y + a_4xy$, then $u_A \sim u_P$ can be expressed as follows:

$$\begin{aligned} u_A &= a_1 + a_2x_A + a_3y_A + a_4x_Ay_A \\ u_B &= a_1 + a_2x_B + a_3y_B + a_4x_By_B \\ u_C &= a_1 + a_2x_C + a_3y_C + a_4x_Cy_C \\ u_D &= a_1 + a_2x_D + a_3y_D + a_4x_Dy_D \\ u_P &= a_1 + a_2x_P + a_3y_P + a_4x_Py_P \end{aligned} \tag{7}$$

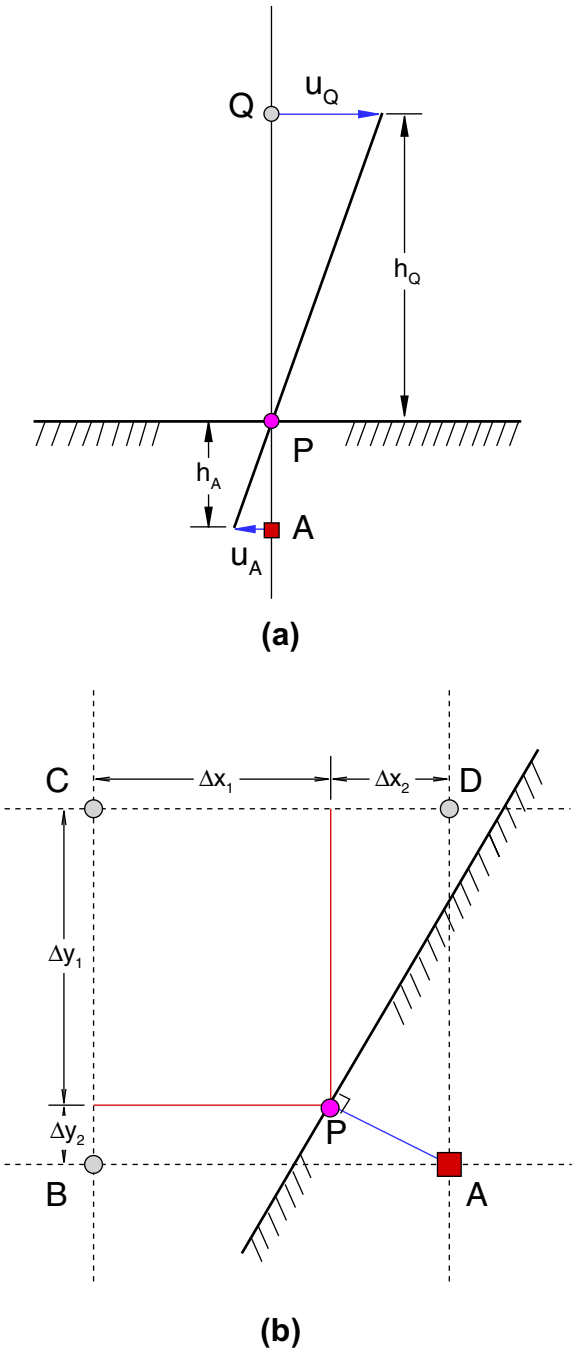

Fig. 1. Schematic of the algebraic interpolation schemes given in Section 3.1. (a) Linear interpolation; (b) bilinear interpolation.

One can then solve the following matrix to get the values of coefficients $a_1 \sim a_4$ and, in turn, the value of u_P from

$$\begin{bmatrix} 1 & x_A & y_A & x_A y_A \\ 1 & x_B & y_B & x_B y_B \\ 1 & x_C & y_C & x_C y_C \\ 1 & x_D & y_D & x_D y_D \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} u_A \\ u_B \\ u_C \\ u_D \end{bmatrix}$$

In most of the problems, the no-slip velocity condition at P is known and we can always use the values of u_B, u_C, u_D and u_P to calculate u_A. In other words, the value of u_A can be calculated in a way to satisfy the no-slip condition on the immersed

boundary (point P). Based on the expression of u_A in Eq. (7) and the uniquely determined values of a_1, a_2, a_3 and a_4 from the above 4×4 matrix equation, u_P is derived as

$$u_P = \frac{[(u_D \cdot \Delta x_1 + u_C \cdot \Delta x_2)\Delta y_2 + (u_A \cdot \Delta x_1 + u_B \cdot \Delta x_2)\Delta y_1]}{[(\Delta x_1 + \Delta x_2)(\Delta y_1 + \Delta y_2)]} \tag{8}$$

where $\Delta x_1 = x_P - x_B, \Delta x_2 = x_A - x_P, \Delta y_1 = y_D - y_P, \Delta y_2 = y_P - y_A$ are shown in Fig. 1(b). By setting $u_P = 0$ in Eq. (8), u_A can be derived as

$$u_A = \frac{-(u_D \cdot \Delta x_1 + u_C \cdot \Delta x_2)\Delta y_2}{\Delta y_1 \cdot \Delta x_1} - \frac{u_B \cdot \Delta x_2}{\Delta x_1} \tag{9}$$

3.2. Differentially interpolated method for the direct forcing term $\underline{f}$

Imposition of nodal forces at the immersed boundaries is the key in developing the IB method. In general, the forcing points are not necessarily located on the immersed boundary. Very often, some of these points will be present inside the body. Thus, an interpolation procedure is required and the scheme employed to interpolate the velocity in the solid-fluid cells determines the degree of simulation accuracy. However, boundary treatment by the polynomials used in algebraic-based approaches may lead to some numerical instabilities. Taking Fig. 2 as an example, we can not prescribe u_A from Eq. (9) since only the velocities at fluid-points B and C are known. A remedy for this type of problem is to change the interpolation function to $b_1 + b_2 x + b_3 y$. Given the fluid-points B and C, the immersed boundary point P, and the solid point A, $u_A \sim u_P$ can be expressed as follows:

$$\begin{aligned} u_A &= b_1 + b_2 x_A + b_3 y_A \\ u_B &= b_1 + b_2 x_B + b_3 y_B \\ u_C &= b_1 + b_2 x_C + b_3 y_C \\ u_P &= b_1 + b_2 x_P + b_3 y_P \end{aligned} \tag{10}$$

One can then solve the following matrix to get the values of $b_1 \sim b_3$ from

$$\begin{bmatrix} 1 & x_A & y_A \\ 1 & x_B & y_B \\ 1 & x_C & y_C \end{bmatrix} \cdot \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} u_A \\ u_B \\ u_C \end{bmatrix}$$

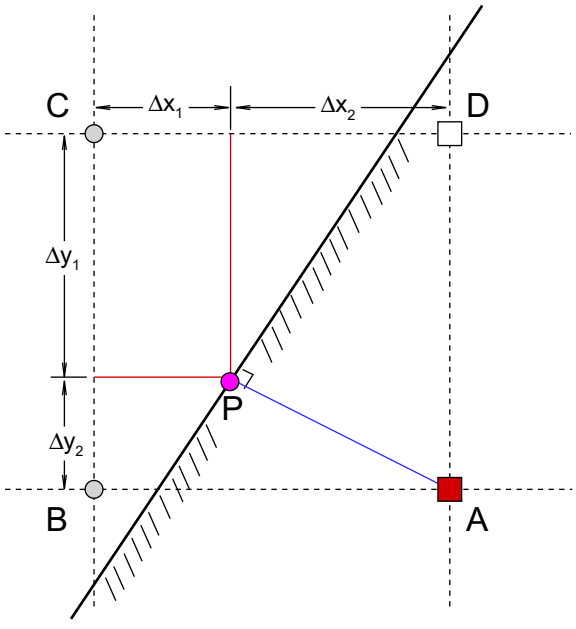

Fig. 2. Schematic of the bilinear algebraic interpolation scheme for the cell involving two fluid nodes.

and, in turn, the value of u_P given below

$$u_P = \frac{[\Delta x_1(\Delta y_1 + \Delta y_2)\cdot u_A + (\Delta x_2 \Delta y_1 - \Delta x_1 \Delta y_2)\cdot u_B + \Delta y_2(\Delta x_1 + \Delta x_2)\cdot u_C]}{[(\Delta x_1 + \Delta x_2)(\Delta y_1 + \Delta y_2)]} \tag{11}$$

By setting $u_P = 0$ in Eq. (11), u_A can be derived as

$$u_A = -\frac{[(\Delta x_2 \Delta y_1 - \Delta x_1 \Delta y_2)\cdot u_B]}{[\Delta x_1(\Delta y_1 + \Delta y_2)]} - \frac{[\Delta y_2(\Delta x_1 + \Delta x_2)\cdot u_C]}{[\Delta x_1(\Delta y_1 + \Delta y_2)]} \tag{12}$$

Numerical instabilities were, however, reported to occur when using the above interpolation [26,28]. As the immersed boundary point P is close to one of the fluid-points, the matrix equation will be ill-conditioned or singular. This will lead to an unphysically very large value of u_A and it will break-down the calculation at the solid point A. Thus, improvement on this type of methods is still in need [28,33]. In order to avoid this problem, we will propose a different way, that is applicable easily to three-dimensional problems, to impose the no-slip condition without the interpolated problems encountered in the algebraically-interpolated method.

Following the idea given in [34], we define the value u_Q at point Q, which is the image of the ghost point A through the boundary point P, as shown in Fig. 3. It follows that the length between points A and P (or $\overline{AP}$) is equal to the length between points P and Q (or $\overline{PQ}$). The value u_A can be derived as a function of u_Q and u_P by performing the Taylor series expansion along the direction that is normal to the immersed boundary

$$u_A = 2u_P - u_Q \tag{13}$$

Now, how to determine the value u_Q at point Q, which is usually not in alignment with the mesh point, is the key issue.

Instead of constructing the local functions shown above, the following advection equation will be employed to transport the values along the direction that is normal to the immersed boundary

$$\frac{\partial u}{\partial \tau} + \underline{n}\cdot\nabla u = 0 \tag{14}$$

In the above, $\tau$is the artificial time and $\underline{n}$ is the unit normal vector. Eq. (14) entails the desirable feature of advecting u along the characteristic direction of the immersed boundary. By solving this equation, one can transport the known value to the ghost point placed inside the solid. Here, we can evaluate u_Q using the differential Eq. (14) at the ghost point A by choosing $\Delta\tau = \overline{APQ} = 2\overline{AP}$. We can then expect that the value u_Q at point Q will transport to the point A. At point A, the transport Eq. (14) is discretized by the following first-order upwind scheme

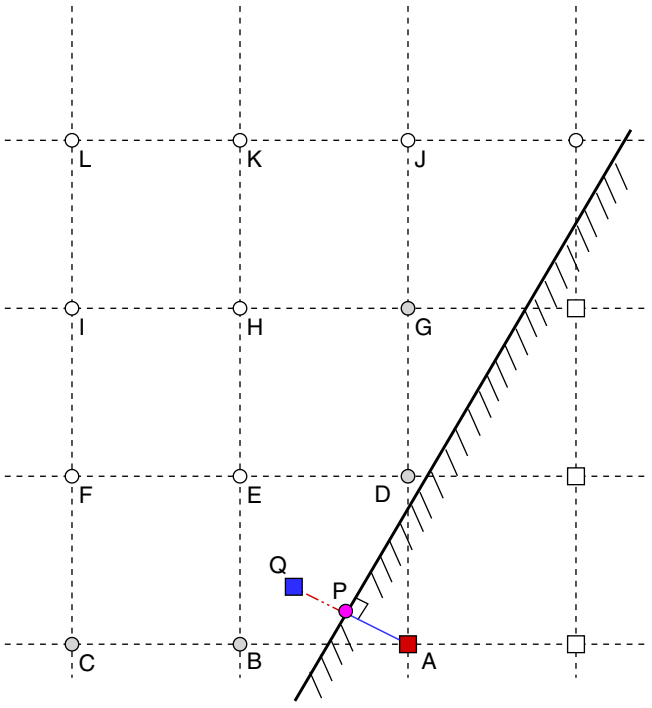

Fig. 3. Schematic of differential interpolation scheme for the cell involving three fluid nodes. Note that the values used for the interpolation are marked by the gray circles.

$$\left(\frac{u_A^{\tau+1}-u_A^{\tau}}{\Delta\tau}+O(\Delta\tau)\right)+\left(n_x\frac{u_A^{\tau}-u_B^{\tau}}{\Delta x}+O(\Delta x)\right)+\left(n_y\frac{u_D^{\tau}-u_A^{\tau}}{\Delta y}+O(\Delta y)\right)=0 \tag{15}$$

where Δx and Δy are the mesh sizes along the x- and y-direction, respectively. Define $\Delta\tau = c_1 h$, $\Delta x = c_2 h$ and $\Delta y = c_3 h$, where h is defined as $(\Delta x + \Delta y)/2$. Since the values of c_1 to c_3 have normally the order of $O(1)$, Eq. (17) can then be derived as

$$\left(\frac{u_A^{\tau+1}-u_A^{\tau}}{c_1 h}+O(h)\right)+\left(n_x\frac{u_A^{\tau}-u_B^{\tau}}{c_2 h}+O(h)\right)+\left(n_y\frac{u_D^{\tau}-u_A^{\tau}}{c_3 h}+O(h)\right)=0 \tag{16}$$

After multiplying h, we can derive the following equation:

$$u_A^{\tau+1}=u_A^{\tau}-c_1\left(n_x\frac{u_A^{\tau}-u_B^{\tau}}{c_2}+n_y\frac{u_D^{\tau}-u_A^{\tau}}{c_3}\right)+O(h^2) \tag{17}$$

The above equation shows that the value $u_A^{\tau+1}$ at point A should be equal to u_Q up to second-order accuracy. However, the value u_A^{τ} is unknown since point A is the solid ghost point.

Thanks to the multidimensional extrapolation idea of Aslam [35], the following two-dimensional extrapolation equation is proposed

$$\frac{\partial u}{\partial \tau^*}+n_x(\Delta x u'_{xx})+n_y(\Delta y u'_{yy})=0 \tag{18}$$

where u'_{xx} and u'_{yy} will be derived below. Define

$$u_{xx}(i,j)=(u_{i,j}-2u_{i-1,j}+u_{i+1,j})/\Delta x^2$$
$$u_{yy}(i,j)=(u_{i,j}-2u_{i,j-1}+u_{i,j+1})/\Delta y^2$$

we are led to derive

$$u'_{xx}(i,j)=u_{xx}(i-1,j)\,if\,n_x(i,j)\geqslant 0;\,else\,u'_{xx}(i,j)=-u_{xx}(i+1,j)$$
$$u'_{yy}(i,j)=u_{yy}(i,j-1)\,if\,n_y(i,j)\geqslant 0;\,else\,u'_{yy}(i,j)=-u_{yy}(i,j+1)$$

By solving Eq. (18) to reach the steady state, the second-order accurate extrapolation along the characteristic direction can be obtained on the ghost points. It is worth to note that if $n_x = 0$ or $n_y = 0$, the above derivation will be reduced to the one-dimensional second-order extrapolation along x or y direction. Numerical results confirm that the solution of second-order accuracy for $u_A^{\tau+1}$ can be calculated from Eq. (17) provided that u_A^{τ} is obtained by solving Eq. (18) to get the steady state solution.

The overall solution procedures of the present method are summarized below:

(I) Calculate the extrapolated-intermediate velocity u_A^{τ} at point A by solving Eq. (18) to get the steady state solution.
(II) Calculate $u_A^{\tau+1}$ at point A by solving Eq. (14) at the artificial time $\tau = 2\overline{AP}$.
(III) Set $u_Q = u_A^{\tau+1}$ and calculate the intermediate velocity u_A at point A by using Eq. (13).

For the sake of completeness, we will show the difference between the algebraically-interpolated method and the currently proposed differentially-interpolated method. The gray circles shown in Figs. 1–4 are the points which are used for calculating u_A. It can be seen that the stencil points used in the present method are not compact. However, implementation of the present method is straightforward and the problem encountered in the algebraically-interpolated method won't be happened in the presently proposed method.

4. Discretization schemes

4.1. Dual-compact scheme for spatial flux terms

In a grid of uniform grid size $\Delta x = \Delta y = h$, the first-order derivative term $\frac{\partial\phi}{\partial x}$ and the second-order derivative term $\frac{\partial^2\phi}{\partial x^2}$ will be couplely approximated within the following three-point compact framework

$$a_1\left.\frac{\partial\phi}{\partial x}\right|_{i-1}+\left.\frac{\partial\phi}{\partial x}\right|_{i}=\frac{1}{h}(c_1\phi_{i-1}+c_2\phi_i+c_3\phi_{i+1})-h\left(b_1\left.\frac{\partial^2\phi}{\partial x^2}\right|_{i-1}+b_2\left.\frac{\partial^2\phi}{\partial x^2}\right|_{i}+b_3\left.\frac{\partial^2\phi}{\partial x^2}\right|_{i+1}\right) \tag{19}$$

$$\bar{b}_1\left.\frac{\partial^2\phi}{\partial x^2}\right|_{i-1}+\left.\frac{\partial^2\phi}{\partial x^2}\right|_{i}+\bar{b}_3\left.\frac{\partial^2\phi}{\partial x^2}\right|_{i+1}=\frac{1}{h^2}(\bar{c}_1\phi_{i-1}+\bar{c}_2\phi_i+\bar{c}_3\phi_{i+1})-\frac{1}{h}\left(\bar{a}_1\left.\frac{\partial\phi}{\partial x}\right|_{i-1}+\bar{a}_2\left.\frac{\partial\phi}{\partial x}\right|_{i}+\bar{a}_3\left.\frac{\partial\phi}{\partial x}\right|_{i+1}\right) \tag{20}$$

As for the terms $\frac{\partial\phi}{\partial y}$ and $\frac{\partial^2\phi}{\partial y^2}$, they can be similarly approximated along the y-direction. Note that the compact schemes for $\frac{\partial\phi}{\partial x}|_i$ and $\frac{\partial^2\phi}{\partial x^2}|_i$ are not independent of each other. They are rather coupled through the terms $\frac{\partial\phi}{\partial x}|_{i-1}, \frac{\partial\phi}{\partial x}|_i, \frac{\partial\phi}{\partial x}|_{i+1}, \frac{\partial^2\phi}{\partial x^2}|_{i-1}, \frac{\partial^2\phi}{\partial x^2}$

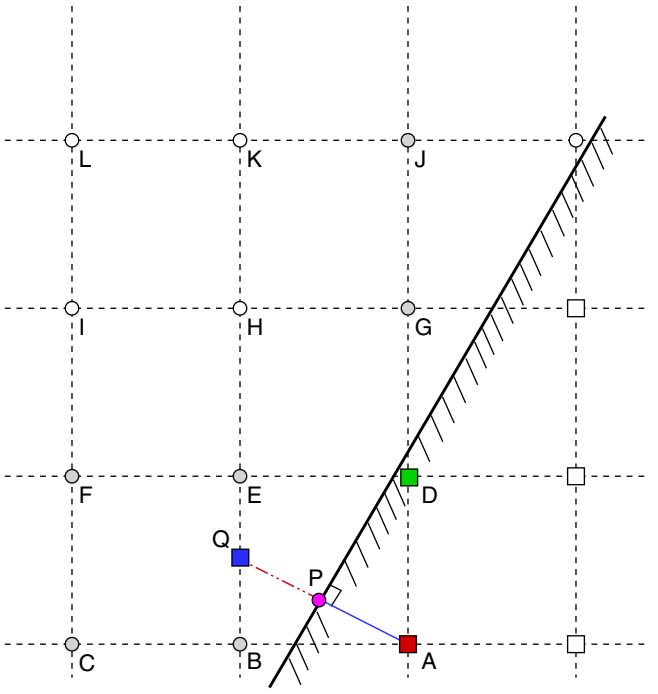

Fig. 4. Schematic of differential interpolation scheme for the cell involving two fluid nodes. Note that the values used for the interpolation are marked by the gray circles.

$|_i, \frac{\partial^2\phi}{\partial x^2}|_{i+1}, \phi_{i-1}, \phi_i$ and ϕ_{i+1}. For the sake of description, we only describe below for the case involving the positive convective coefficient. As for the negative convective coefficient, the derivation can be done in the similar way.

4.1.1. Compact scheme for the second-order derivative term

The second-order derivative term $\frac{\partial^2\phi}{\partial x^2}$ is normally approximated by the center-type scheme since its discretization error is prevailingly dissipative. For this reason, the weighting coefficients shown in Eq. (20) will be determined solely by the modified equation analysis so as to get a higher spatial accuracy. Derivation of the coefficients $\bar{a}_1, \bar{a}_2, \bar{a}_3, \bar{b}_1, \bar{b}_3, \bar{c}_1, \bar{c}_2$ and $\bar{c}_3$ is started by applying the Taylor series expansions for $\phi_{i\pm1}, \frac{\partial\phi}{\partial x}|_{i\pm1}$ and $\frac{\partial^2\phi}{\partial x^2}|_{i\pm1}$ with respect to $\phi_i, \frac{\partial\phi}{\partial x}|_i$ and $\frac{\partial^2\phi}{\partial x^2}|_i$ and, then, eliminating the leading eight error terms derived in the modified equation. Elimination of these leading error terms enables us to get the following set of algebraic equations for the coefficients shown in Eq. (20)

$$\bar{c}_1 + \bar{c}_2 + \bar{c}_3 = 0 \tag{21}$$

$$-\bar{a}_1 - \bar{a}_2 - \bar{a}_3 - \bar{c}_1 + \bar{c}_3 = 0 \tag{22}$$

$$\bar{a}_1 - \bar{a}_3 - \bar{b}_1 + \frac{\bar{c}_1}{2} - \bar{b}_3 + \frac{\bar{c}_3}{2} = 1 \tag{23}$$

$$-\frac{\bar{a}_1}{2} - \frac{\bar{a}_3}{2} + \bar{b}_1 - \bar{b}_3 - \frac{\bar{c}_1}{6} + \frac{\bar{c}_3}{6} = 0 \tag{24}$$

$$\frac{\bar{a}_1}{6} - \frac{\bar{a}_3}{6} - \frac{\bar{b}_1}{2} + \frac{\bar{c}_1}{24} - \frac{\bar{b}_3}{2} + \frac{\bar{c}_3}{24} = 0 \tag{25}$$

$$-\frac{\bar{a}_1}{24} - \frac{\bar{a}_3}{24} + \frac{\bar{b}_1}{6} - \frac{\bar{b}_3}{6} - \frac{\bar{c}_1}{120} + \frac{\bar{c}_3}{120} = 0 \tag{26}$$

$$\frac{\bar{a}_1}{120} - \frac{\bar{a}_3}{120} - \frac{\bar{b}_1}{24} - \frac{\bar{b}_3}{24} + \frac{\bar{c}_1}{720} + \frac{\bar{c}_2}{720} = 0 \tag{27}$$

$$-\frac{\bar{a}_1}{720} - \frac{\bar{a}_3}{720} + \frac{\bar{b}_1}{120} - \frac{\bar{b}_3}{120} - \frac{\bar{c}_1}{5040} + \frac{\bar{c}_3}{5040} = 0 \tag{28}$$

By solving Eqs. (21)–(28), we are led to get these coefficients as $\bar{a}_1 = -\frac{9}{8}, \bar{a}_2 = 0, \bar{a}_3 = \frac{9}{8}, \bar{b}_1 = -\frac{1}{8}, \bar{b}_3 = -\frac{1}{8}, \bar{c}_1 = 3, \bar{c}_2 = -6$ and $\bar{c}_3 = 3$, which are exactly the same as those derived by Chu and Fan [39]. By virtue of the following derived modified equation, we are led to know that the presently derived coefficients can render the spatial accuracy order of sixth for the approximated equation of $\frac{\partial^2\phi}{\partial x^2}$

$$\frac{\partial^2\phi}{\partial x^2} = \frac{\partial^2\phi}{\partial x^2}\bigg|_{exact} + \frac{h^6}{20160}\frac{\partial^8\phi}{\partial x^8} + \frac{h^8}{604800}\frac{\partial^{10}\phi}{\partial x^{10}} + O(h^{12}) + \cdots$$

4.1.2. Dispersion-relation-preserving compact scheme for the first-order derivative term

The coefficients a_1, b_1, b_2, b_3 , c_1, c_2 and c_3 will be partly determined by the Taylor series by expanding $\phi_{i\pm1}, \frac{\partial\phi}{\partial x}|_{i-1}$ and $\frac{\partial^2\phi}{\partial x^2}|_{i\pm1}$ with respect to ϕ_i, $\frac{\partial\phi}{\partial x}|_i$ and $\frac{\partial^2\phi}{\partial x^2}|_i$. By eliminating the leading six error terms derived in the modified equation, the following set of algebraic equations for Eq. (19) can be derived as

$$c_1 + c_2 + c_3 = 0 \tag{29}$$

$$-a_1 - c_1 + c_3 = 1 \tag{30}$$

$$-a_1 + b_1 + b_2 + b_3 - \frac{c_1}{2} - \frac{c_3}{2} = 0 \tag{31}$$

$$\frac{a_1}{2} - b_1 + b_3 + \frac{c_1}{6} - \frac{c_3}{6} = 0 \tag{32}$$

$$-\frac{a_1}{6} + \frac{b_1}{2} + \frac{b_3}{2} - \frac{c_1}{24} - \frac{c_3}{24} = 0 \tag{33}$$

$$\frac{a_1}{24} - \frac{b_1}{6} + \frac{b_3}{6} + \frac{c_1}{120} - \frac{c_3}{120} = 0 \tag{34}$$

One more algebraic equation is needed for us to uniquely determine all the seven introduced coefficients shown in Eq. (19).

It is well known that dispersion relation governs the relation between the angular frequency and the wavenumber of the first-order dispersive term [36]. Hence it is of primary importance for us to develop a scheme that accommodates the dispersion relation for the first-order derivative term. To preserve the dispersion relation, the Fourier transform and its inverse for ϕ given below will be applied

$$\tilde{\phi}(\alpha) = \frac{1}{2\pi}\int_{-\infty}^{+\infty}\phi(x)\exp(-\mathbf{i}\alpha x)dx, \tag{35}$$

$$\phi(x) = \int_{-\infty}^{+\infty}\tilde{\phi}(\alpha)\exp(\mathbf{i}\alpha x)d\alpha. \tag{36}$$

Note that $\mathbf{i}$ shown above is equal to $\sqrt{-1}$. Development of the advection scheme is followed by performing Fourier transform on each term shown in Eqs. (19) and (20). The actual wavenumber α for these two equations can be therefore derived as

$$\mathbf{i}\alpha h(a_1\exp(-\mathbf{i}\alpha h) + 1) \simeq c_1\exp(-\mathbf{i}\alpha h) + c_2 + c_3\exp(\mathbf{i}\alpha h) - (\mathbf{i}\alpha h)^2(b_1\exp(-\mathbf{i}\alpha h) + b_2 + b_3\exp(\mathbf{i}\alpha h)) \tag{37}$$

$$(\mathbf{i}\alpha h)^2\left(-\frac{1}{8}\exp(-\mathbf{i}\alpha h) + 1 - \frac{1}{8}\exp(\mathbf{i}\alpha h)\right) \simeq 3\exp(-\mathbf{i}\alpha h) - 6 + 3\exp(\mathbf{i}\alpha h) - \mathbf{i}\alpha h\left(-\frac{8}{9}\exp(-\mathbf{i}\alpha h) + \frac{8}{9}\exp(\mathbf{i}\alpha h)\right) \tag{38}$$

In an approximation sense, the effective wavenumbers α' and α'' should have the same expressions as those shown in the right-hand sides of Eqs. (37) and (38) [36]. In other words, it is desirable to express $\alpha\prime$ and α'' as follows :

$$\mathbf{i}\alpha' h(a_1\exp(-\mathbf{i}\alpha h) + 1) = c_1\exp(-\mathbf{i}\alpha h) + c_2 + c_3\exp(\mathbf{i}\alpha h) - (\mathbf{i}\alpha'' h)^2(b_1\exp(-\mathbf{i}\alpha h) + b_2 + b_3\exp(\mathbf{i}\alpha h)) \tag{39}$$

$$\mathbf{i}\alpha' h\left(-\frac{8}{9}\exp(-\mathbf{i}\alpha h) + \frac{8}{9}\exp(\mathbf{i}\alpha h)\right) = 3\exp(-\mathbf{i}\alpha h) - 6 + 3\exp(\mathbf{i}\alpha h) - (\mathbf{i}\alpha'' h)^2\left(-\frac{1}{8}\exp(-\mathbf{i}\alpha h) + 1 - \frac{1}{8}\exp(\mathbf{i}\alpha h)\right) \tag{40}$$

By solving Eqs. (39) and (40), the expressions for α' and α'' can be derived as follows:

$$\begin{aligned}\alpha' h = -\mathbf{i}(&24b_1\exp(-2\mathbf{i}\alpha h) + c_1\exp(-2\mathbf{i}\alpha h) + c_3 + c1 + 24b_1 + c_2\exp(-\mathbf{i}\alpha h) + 24b_2\exp(-\mathbf{i}\alpha h) + 24b_3 - 48b_1\\ &\times\exp(-\mathbf{i}\alpha h) - 8c_1\exp(-\mathbf{i}\alpha h) - 48b_3\exp(\mathbf{i}\alpha h) + 24b_2\exp(\mathbf{i}\alpha h) + 24b_3\exp(2\mathbf{i}\alpha h) - 48b_2 + c_2\exp(\mathbf{i}\alpha h)\\ &+ c_3\exp(2\mathbf{i}\alpha h) - 8c_3\exp(\mathbf{i}\alpha h) - 8c_2)/(-8 + \exp(\mathbf{i}\alpha h) - 8a_1\exp(-\mathbf{i}\alpha h) + a_1\exp(-2\mathbf{i}\alpha h) - 9b_1\exp(-2\mathbf{i}\alpha h)\\ &- 9b_2\exp(-\mathbf{i}\alpha h) + 9b_2\exp(\mathbf{i}\alpha h) + 9b_3\exp(2\mathbf{i}\alpha h) + a_1 + 9b_1 - 9b_3 + \exp(\mathbf{i}\alpha h))\end{aligned} \tag{41}$$

$$\alpha'' h = \sqrt{-\frac{3\exp(-\mathbf{i}\alpha h) - 6 + 3\exp(\mathbf{i}\alpha h) - \mathbf{i}\alpha' h\left(-\frac{8}{9}\exp(-\mathbf{i}\alpha h) + \frac{8}{9}\exp(\mathbf{i}\alpha h)\right)}{-\frac{1}{8}\exp(-\mathbf{i}\alpha h) + 1 - \frac{1}{8}\exp(\mathbf{i}\alpha h)}} \tag{42}$$

It is demanded that $\alpha h \approx \Re[\alpha' h]$, where $\Re[\alpha' h]$ denotes the real part of $\alpha' h$, to get a better dispersive accuracy for α'. This implies that $E(\alpha)$ defined below should be a very small and positive value

$$E(\alpha) = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}}[W(\alpha h - \Re[\alpha' h])]^2 d(\alpha h) = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}}[W(\gamma - \Re[\gamma'])]^2 d\gamma \tag{43}$$

where $\gamma = \alpha h$ and $\gamma' = \alpha' h$. Note that Eq. (43) can be analytically integrable provided that the weighting function W, which is the denominator of $(\gamma - \Re[\gamma'])$, is chosen as

$$\begin{aligned} W = & -16 + 72b_3 + 72b_1 - 81b_1^2 - 81b_3^2 - 81b_2^2 - 162b_2b_3cos(\gamma) - 144a_1b_3cos(\gamma) - 162b_1b_2cos(\gamma) - a_1^2cos(\gamma)^2 \\ & + 8a_1^2cos(\gamma) - 18b_3cos(\gamma)^3 + 18b_1cos(\gamma)^3 + 81b_2^2cos(\gamma)^2 + 162b_1b_3 - 72b_1cos(\gamma)^2 + 81b_3^2cos(\gamma)^2 \\ & + 81b_1^2cos(\gamma)^2 - 72a_1b_2 - 18b_1cos(\gamma) + 16a_1cos(\gamma)^2 - 2a_1cos(\gamma)^3 + 72b_3cos(\gamma)^2 + 18b_3cos(\gamma) - 32a_1cos(\gamma) \\ & - 36a_1b_3cos(\gamma)^4 - 18a_1b_2cos(\gamma)^3 + 162b_2b_3cos(\gamma)^3 + 162b_1b_2cos(\gamma)^3 + 324b_1b_3cos(\gamma)^4 + 72a_1b_2cos(\gamma)^2 \\ & + 144a_1b_3cos(\gamma)^3 - 486b_1b_3cos(\gamma)^2 + 36a_1b_3cos(\gamma)^2 + 18a_1b_2cos(\gamma) + 8cos(\gamma) - 16a_1^2 - cos(\gamma)^2 \end{aligned} \tag{44}$$

It is worth pointing out that the integration interval shown in Eq. (43) should cover a complete period of the sine (or cosine) wave. To make E defined in Eq. (43) to be positive and minimum, the following extreme condition is enforced

$$\frac{\partial E}{\partial c_3} = 0 \tag{45}$$

The above equation, which is enforced to preserve the dispersion relation, will be used together with another six previously derived algebraic equations by way of the modified equation analysis to get a higher dissipation accuracy as well as a dispersion accuracy. The resulting seven introduced unknowns can be uniquely determined as

$$a_1 = 0.875 \tag{46}$$

$$b_1 = 0.12512823415990895606 \tag{47}$$

$$b_2 = -0.24871765840091043936 \tag{48}$$

$$b_3 = 0.000128234159908956 0636 \tag{49}$$

$$c_1 = -1.9359611900810925272 \tag{50}$$

$$c_2 = 1.9969223801621850545 \tag{51}$$

$$c_3 = -0.060961190081092527237 \tag{52}$$

We remark here that the upwinding scheme developed to approximate $\frac{\partial \phi}{\partial x}$ can be easily shown to have the spatial accuracy order of fifth thanks to the following modified equation:

$$\begin{aligned} \frac{\partial \phi}{\partial x} = & \frac{\partial \phi}{\partial x}|_{exact} - 0.00070085615243989224 75h^5\frac{\partial^6 \phi}{\partial x^6} + 0.00019841269841269841 27h^6\frac{\partial^7 \phi}{\partial x^7} \\ & - 0.00004988305074583303 90h^7\frac{\partial^8 \phi}{\partial x^8} + O(h^8) + \cdots \end{aligned} \tag{53}$$

4.2. Divergence-free-condition compensated solution algorithm

Following the idea of projection method, we can use the predicted pressure p^* to calculate the intermediate velocity $\underline{u}^*$, which does not satisfy the divergence-free constraint condition, from the following momentum vector equation

$$\frac{\underline{u}^* - \underline{u}^n}{\Delta t} + (\underline{u}^* \cdot \nabla)\underline{u}^* - \frac{1}{Re}\nabla^2 \underline{u}^* + \nabla p^* = \underline{f} \tag{54}$$

The velocity is then projected to the divergence-free space to update the pressure value as

$$\frac{\underline{u}^{n+1} - \underline{u}^*}{\Delta t} = -\nabla p' \tag{55}$$

$$p^{n+1} = p^* + p' \tag{56}$$

$$\nabla \cdot \underline{u}^{n+1} = 0 \tag{57}$$

Substitution of Eq. (57) to the semi-discrete equation $\frac{\underline{u}^{n+1} - \underline{u}^n}{\Delta t} + (\underline{u}^{n+1} \cdot \nabla)\underline{u}^{n+1} - \frac{1}{Re}\nabla^2 \underline{u}^{n+1} + \nabla p^{n+1} = \underline{f}$ yields the following equation:

$$\frac{\underline{u}^{n+1} - \underline{u}^n}{\Delta t} + (\underline{u}^* \cdot \nabla)\underline{u}^* - \frac{1}{Re}\nabla^2 \underline{u}^* + \nabla p^* = -\nabla p' + \underline{f} + M_1 + M_2 \tag{58}$$

where

$$M_1 = \Delta t[(\underline{u}^* \cdot \nabla)\nabla p' + (\nabla p' \cdot \nabla)\underline{u}^* - \frac{1}{Re}\nabla^2(\nabla p')] \tag{59}$$

$$M_2 = -\Delta t^2(\nabla p' \cdot \nabla)\nabla p') \tag{60}$$

By performing the divergence operator on Eq. (57) and imposing the constraint equation $\nabla \cdot \underline{u}^{n+1} = 0$, we can rewrite Eq. (60) as

$$M_1 = \Delta t[(\underline{u}^* \cdot \nabla)\nabla p' + (\nabla p' \cdot \nabla)\underline{u}^*] - \frac{1}{Re}\nabla(\nabla \cdot \underline{u}^*) \tag{61}$$

By setting $M_{\mathrm{DFC}} = -\nabla p' + M_1 + M_2$, which is shown in Eq. (58), one is led to know that the introduced term M_{DFC} replaces the divergence-free condition. We can therefore call M_{DFC} as the divergence-free-condition (DFC) compensated momentum sourcing term derived to replace the divergence-free constraint condition with the momentum sourcing term.

The computational procedures of the proposed IB method, combined with the DFC compensated method (IB-DFC), are summarized as follows:

Given the predicted values of $\underline{u}_0^*$ and p_0^* for $s = 1, 2, \ldots$

(I) Calculate the intermediate velocity $\underline{u}_s^{n+1}$ by solving the following equation at the mesh points in the fluid-domain.

$$\frac{\underline{u}_s^{n+1} - \underline{u}^n}{\Delta t} + \underline{u}_s^* \cdot \nabla \underline{u}_s^* - \frac{1}{Re}\nabla^2 \underline{u}_s^* = -\nabla p_{s-1}^* + \underline{f}_{s-1} + M_{DFC_{s-1}}$$

(II) Calculate the extrapolated-intermediate velocity $\underline{u}_{A_s}^{n+1}$ at the near-fluid solid points by the present IB method, which is described in Section 3.2.

(III) Calculate p_s^* and $\underline{u}_{s+1}^*$ by

$$p_s^* = p_{s-1}^* + p_s'$$
$$\underline{u}_{s+1}^* = \underline{u}_s^{n+1}$$

(IV) When reaching convergence, set $u_{s+1}^{n+1} = u^{n+1}$ and $p_s^* = p^{n+1}$ and goto the next time step; else goto (I).

The above algorithm allows us to employ the explicit scheme to calculate the solution iteratively from the momentum equations. With the explicit formulation for pressure, this algorithm turns out to be fully explicit. Also, the non-linear term can be linearized in the iterative sequence. Note that if the divergence-free constraint condition is satisfied, both M_{DFC} and $\nabla p'$ will disappear, thereby yielding $u^* = u^{n+1}$ and $p^* = p^{n+1}$.

The remaining issue left for the derivation is to derive p'. By performing the divergence operator on Eq. (57), for example, we can derive

$$\nabla \cdot \underline{u}^{n+1} = \nabla \cdot \underline{u}^* - \Delta t \nabla^2 p' \tag{62}$$

Enforcement of $\nabla \cdot \underline{u}^{n+1} = 0$ can yield

$$\nabla^2 p' = \frac{\nabla \cdot \underline{u}^*}{\Delta t} \tag{63}$$

To solve the Poisson equation for p', one needs to specify its associated boundary condition. This necessity is, however, unphysical since no boundary condition is available for the pressure. As the above Poisson equation indicates, the pressure correction has association with the defect of divergence-free condition. It is more natural to explicitly relate the pressure with the defect divergence-free constraint. Therefore, at an interior point (i, j) the central approximation of Eq. (63) leads to

$$2\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)p'_{i,j} = \frac{\nabla \cdot \underline{u}_{i,j}^*}{\Delta t} - \frac{1}{\Delta x^2}(p'_{i-1,j} + p'_{i+1,j}) - \frac{1}{\Delta y^2}(p'_{i,j-1} + p'_{i,j+1}) \tag{64}$$

By omitting the term $\frac{1}{\Delta x^2}(p'_{i-1,j} + p'_{i+1,j}) + \frac{1}{\Delta y^2}(p'_{i,j-1} + p'_{i,j+1})$, we can get the following equation:

$$p'_{i,j} = -\frac{\Delta x^2 \Delta y^2}{2(\Delta x^2 + \Delta y^2)\Delta t}\nabla \cdot \underline{u}_{i,j^*} \tag{65}$$

The need to solve the pressure Poisson equation, which is subjected to the unphysical specification of pressure boundary condition, is thus avoided.

The above formulation for p' usually leads to an over-estimated pressure due to the omitted term. We can, therefore, employ the following way to compensate the omission term. First, we use (65) to calculate the predicted pressure correction p'^*

$$p'^*_{i,j} = -\frac{\Delta x^2 \Delta y^2}{2(\Delta x^2 + \Delta y^2)\Delta t}\nabla \cdot \underline{u}_{i,j}^* \tag{66}$$

This is followed by calculating the pressure correction p' from p'^*

$$p'_{i,j} = p'^*_{i,j} + \frac{\Delta y^2}{2(\Delta x^2 + \Delta y^2)}(p'^*_{i-1,j} + p'^*_{i+1,j}) + \frac{\Delta x^2}{2(\Delta x^2 + \Delta y^2)}(p'^*_{i,j-1} + p'^*_{i,j+1}) \tag{67}$$

We can therefore relate the pressure with the defect divergence-free constraint condition without relaxation of p'.

4.3. Combined compact difference scheme for calculating the pressure gradient in collocated grids

Application of staggered grid approaches to solve the incompressible flow equations has long been known to be able to suppress pressure oscillations arising from the even–odd decoupling. This type of approaches can, however, increase programming complexity. Discretization of the differential equations in a domain where velocities and pressure are stored at the same point will be employed in this study. The pressure gradient term ∇p in the momentum equations must be carefully approximated in the non-staggered mesh system. Otherwise, spurious pressure oscillations will be inevitable.

Our underlying idea of avoiding the notorious even–odd decoupling solutions is to employ $p_{i,j}$ when approximating ∇p at an interior node (i,j). Instead of explicitly approximating $\frac{\partial p}{\partial x}|_{i,j}$, its value can be calculated implicitly with the two adjacent terms $\frac{\partial p}{\partial x}|_{i\pm1,j}$ using the compact scheme which has been proposed previously by Sheu and his co-authors in [37,38]. Following the above idea, the combined compact difference (CCD) scheme given by Chu and Fan [39] is used for computing ∇p in present study.

5. Validation of IB-DFC Navier–Stokes solver

To verify the proposed IB-DFC Navier–Stokes solver, two problems amenable to their analytic solutions are investigated firstly. Both problems are solved in the domain defined by $\Omega = [0,1] \times [0,1]$, within which there is a circular body centered at $(x_c, y_c) = (1/2, 1/2)$ with the radius 1/5 shown in Fig. 5.

5.1. Steady-state analytic validation problem

The steady-state Navier–Stokes equations with the exact boundary velocities prescribed in [40] will be solved at $Re = 100$

$$u(x,y) = \frac{-2(1+y)}{(1+x)^2 + (1+y)^2} \tag{68}$$

$$v(x,y) = \frac{2(1+x)}{(1+x)^2 + (1+y)^2} \tag{69}$$

The resulting exact pressure can be derived as

$$p(x,y) = -\frac{2}{(1+x)^2 + (1+y)^2} \tag{70}$$

In Table 1, we tabulate the predicted L_2-error norms and the corresponding rates of convergence based on the solutions (u), (v), (p) obtained at three different uniform grids $(21^2, 41^2, 81^2)$. It can be found from the computed errors that the predicted

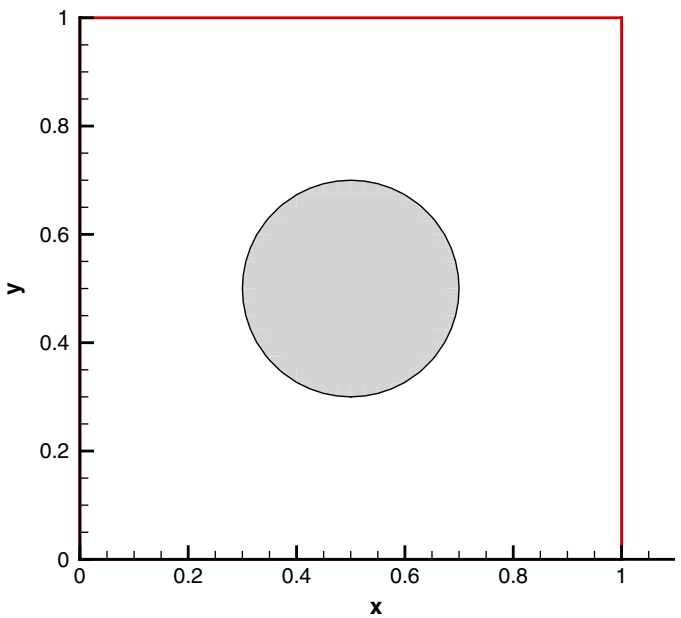

Fig. 5. The physical domain for the problem investigated in Section 5.

Table 1
The predicted L_2-error norms and the corresponding rates of convergence (R.O.C.) for the problem, given in Section 5.1, carried out at three chosen mesh sizes.

Mesh size	u	R.O.C.	v	R.O.C.	p	R.O.C.
$h = 1/20$	7.386E−5	–	7.039E−5	–	1.453E−4	–
$h = 1/40$	1.374E−5	2.426	1.429E−5	2.299	3.742E−5	1.957
$h = 1/80$	3.248E−6	2.080	3.362E−6	2.088	1.052E−5	1.830

Table 2
The predicted L_2-error norms and the corresponding rates of convergence (R.O.C.) for the problem, given in Section 5.2, carried out at three chosen mesh sizes.

Mesh size	u	R.O.C.	v	R.O.C.	p	R.O.C.
$h = 1/20$	1.565E−3	–	1.511E−3	–	2.878E-3	–
$h = 1/40$	3.797E−4	2.043	3.557E−4	2.086	7.090E−4	2.021
$h = 1/80$	8.498E−5	2.159	8.118E−5	2.131	1.671E−4	2.084

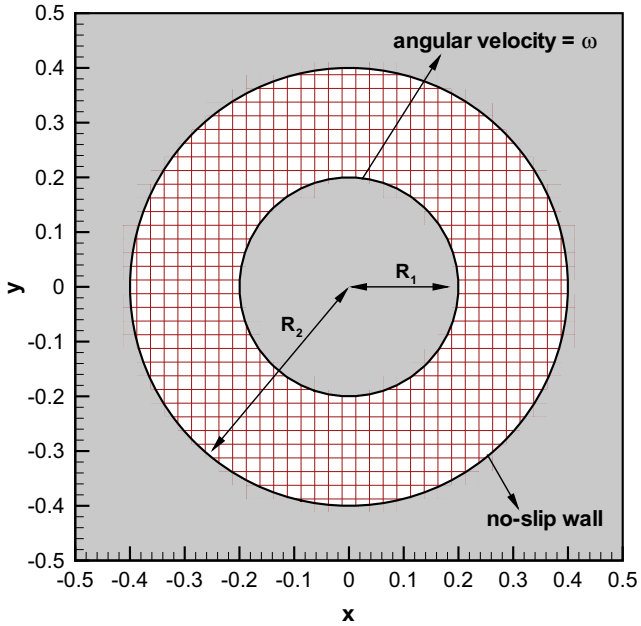

Fig. 6. Schematic of the specified boundary conditions for the Taylor–Couette problem considered in Section 5.3.

Table 3
The computed L_2-error norms and the corresponding spatial rates of convergence (s.r.c.) for the problem, given in Section 5.3, carried out at four chosen meshes for $Re = 200$.

Mesh size	u	R.O.C.	v	R.O.C.	p	R.O.C.
$h = 1/10$	4.431E−2	–	4.431E−2	–	3.723E−2	–
$h = 1/20$	7.990E−3	2.471	7.990E−3	2.471	8.837E−3	2.075
$h = 1/40$	1.620E−3	2.301	1.620E−3	2.301	1.755E−3	2.332
$h = 1/80$	3.576E−4	2.180	3.576E−4	2.180	5.556E−4	1.659

solutions agree very well with the exact solutions. Moreover, the global second order convergence can be obtained using the proposed scheme.

5.2. Transient analytic validation problem

The transient Navier–Stokes equations with the following exact solutions are also solved at $t = 1$ for the validation sake

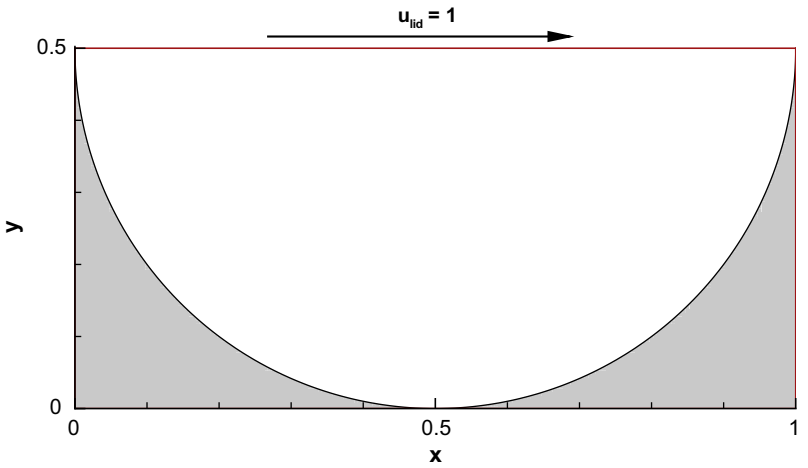

Fig. 7. Schematic of the semi-circular lid-driven cavity problem investigated in Section 6.1.

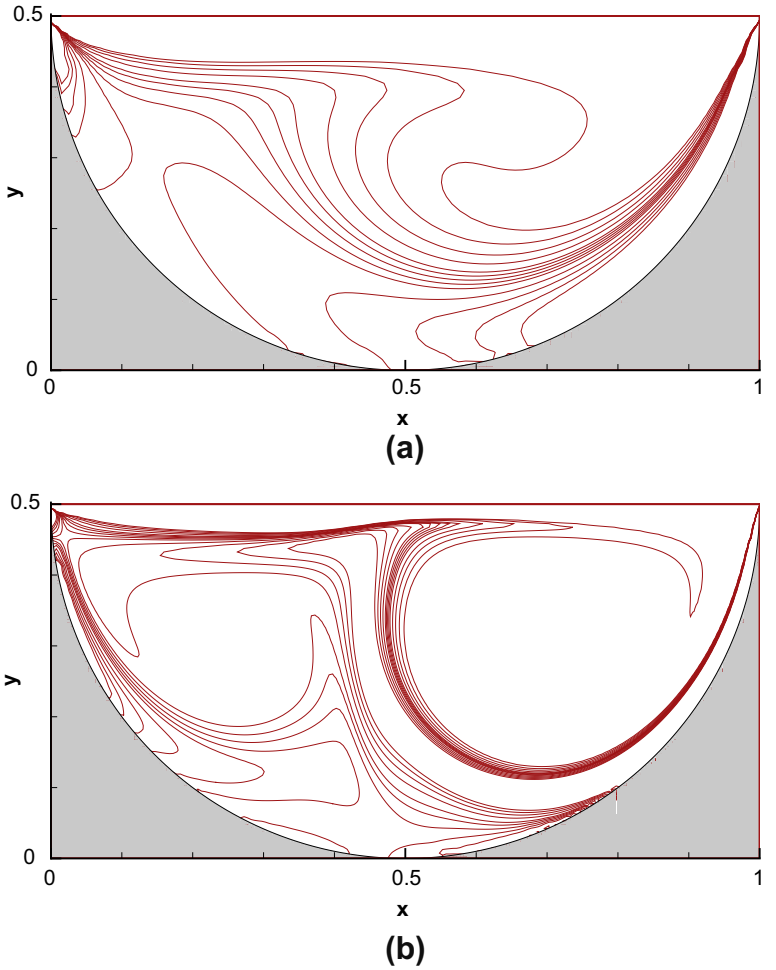

Fig. 8. The predicted vorticity contours for the flow investigated at two Reynolds numbers in a semi-circular lid-driven cavity. (a)*Re* = 500; (b)*Re* = 5000.

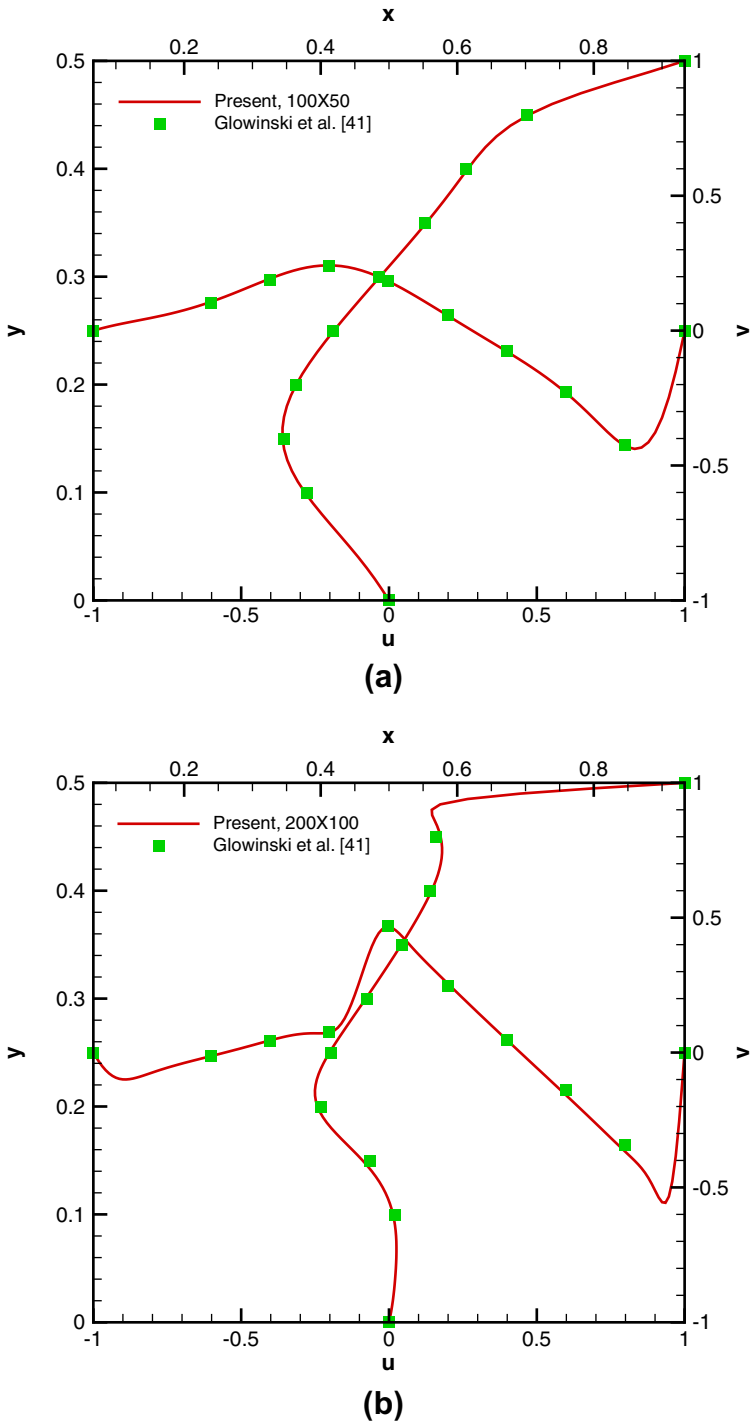

Fig. 9. Comparison of the mid-plane velocity profiles $u(x, 0.5)$ and $v(0.5, y)$ predicted at two Reynolds numbers. (a) $Re = 500$; (b) $Re = 5000$.

$$u(x,y,t) = -\cos(\pi x)\sin(\pi y)\exp(-2\pi^2 t/Re) \tag{71}$$

$$v(x,y,t) = \sin(\pi x)\cos(\pi y)\exp(-2\pi^2 t/Re) \tag{72}$$

$$p(x,y,t) = -\frac{1}{4}(\cos(2\pi x) + \cos(2\pi y))\exp(-4\pi^2 t/Re) \tag{73}$$

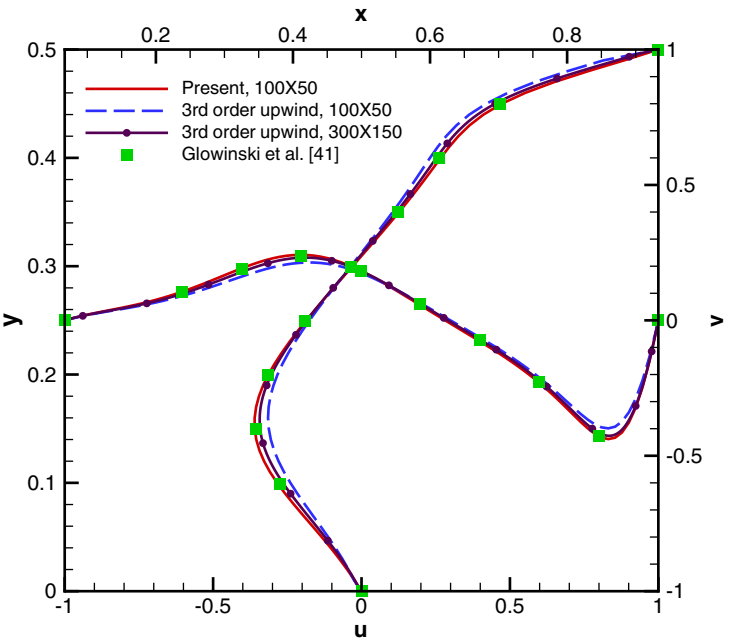

Fig. 10. Comparison of the predicted solution computed by the dual-compact scheme and third-order upwind scheme at $Re = 500$.

All the solutions will be calculated at $Re = 100, \Delta x = \Delta y = \frac{1}{20}, \frac{1}{40}, \frac{1}{80}$ and $\Delta t = 10^{-3}$. The proposed method is justified by the predicted L_2-norm errors and the rates of convergence tabulated in Table 2. The global second order convergence can be also confirmed through this test problem.

5.3. Taylor–Couette problem

The flow field driven by the inner cylinder with a constant angular velocity is investigated in the cylinder cavity. In the domain $(x, y) = (-0.5 \sim 0.5, -0.5 \sim 0.5)$, there are a inner cylinder, moving with the angular velocity ω, with the radius $R_1 (= 0.2)$, and a outer cylinder is with the radius $R_2 (= 0.4)$ and no-slip condition shown in Fig. 6.

The exact solutions for velocity pressure can be derived as

$$u(x,y) = -K\left(\frac{R_2^2}{r^2} - 1\right)y \tag{74}$$

$$v(x,y) = -K\left(\frac{R_2^2}{r^2} - 1\right)x \tag{75}$$

$$p(x,y) = K^2\left(\frac{r^2}{2} - \frac{R_2^4}{2r^2} - R_2^2 \log(r^2)\right) \tag{76}$$

where $K = \frac{\omega R_1^2}{R_2^2 - R_1^2}$ and $r = \sqrt{x^2 - y^2}$. The solutions will be calculated at $Re = 200, \Delta x = \Delta y = \frac{1}{10}, \frac{1}{20}, \frac{1}{40}$ and $\frac{1}{80}$. The proposed method is justified by the predicted L_2-norm errors and the rates of convergence tabulated in Table 3. Again, the global second order convergence can be confirmed through this test problem (see Table 3)

6. Numerical results

6.1. Flow in a semi-circular lid-driven cavity

The flow field driven by a constant upper lid velocity u_{lid} is then investigated in the semi-circular cavity, shown in Fig. 7, at $Re = 500$ and 5000. Note that $L(=1)$ is chosen as the characteristic length and $u_{lid}(=1)$ is the characteristic velocity. The simulated vorticity contours and the corresponding mid-plane velocity profiles $u(0.5, y)$ and $v(x, 0.5)$ predicted at the mid-planes are plotted in Figs. 8 and 9. Good agreement with the predicted solutions of Glowinski et al. [41] shown in Fig. 9 confirms the fidelity of applying the proposed scheme to simulate the incompressible viscous fluid flow in irregular domain. In order to show the performance of the dual-compact scheme, we compare the solution of semi-circular lid-driven cavity problem for

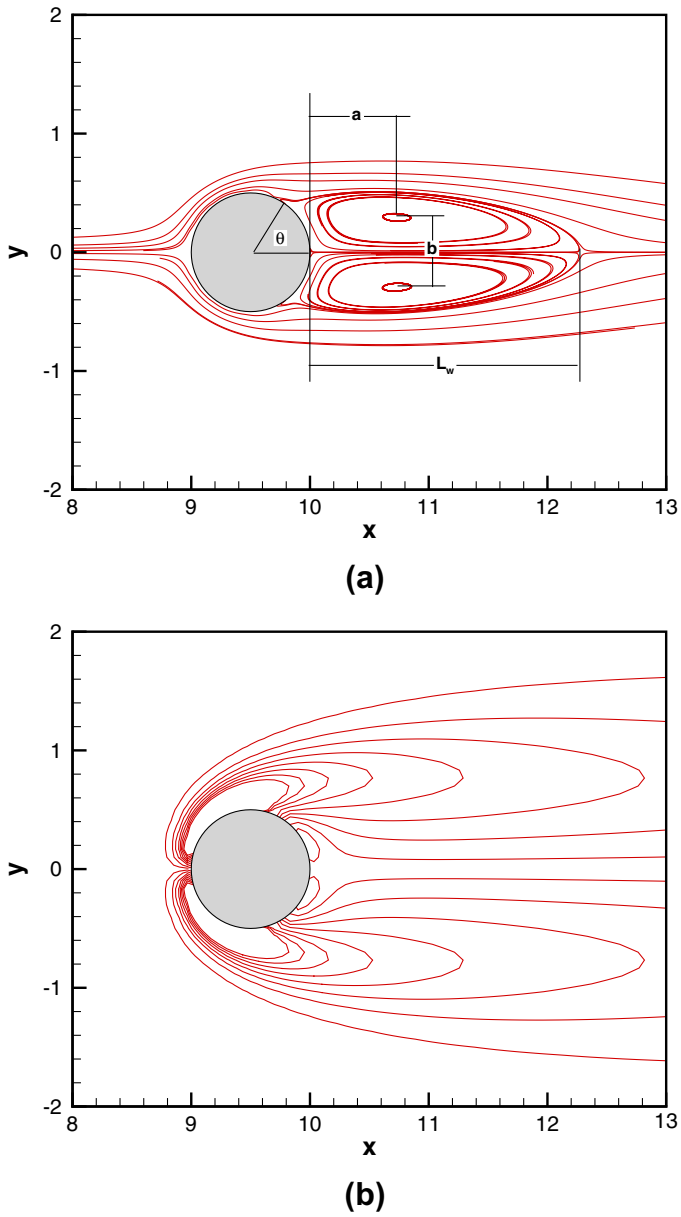

Fig. 11. The predicted results in the near wake of the circular cylinder. (a) Streamline contours and the chosen characteristic lengths; (b) vorticity contours.

Re = 500 with dual-compact scheme and third-order upwind scheme. From Fig. 10, it can be easily seen that our proposed dual-compact scheme is much better than third-order upwind scheme. This also shows the need of using the proposed dual-compact scheme.

6.2. Flow over a circular cylinder

Flow over a circular cylinder has been extensively studied for verifying the IB solvers. At a lower value of Re, the resulting flow is of the diffusion-dominated type and is called the creeping flow. At a somewhat higher value of Re (up to Re = 40), two symmetrical vortices will be stationarily attached behind the cylinder. By increasing the value of Re, the vortices become stretched and the flow will be distorted and broken apart, leading to an alternating vortex shedding (or Kármán vortex street) in the wake.

In this study, drag and lift coefficients and Strouhal number will be calculated for the sake of making a comparison with other numerical results. Drag coefficient is defined as $C_D = \frac{F_x}{\frac{1}{2}u_\infty^2 D}$, and lift coefficient is defined as $C_L = \frac{F_y}{\frac{1}{2}u_\infty^2 D}$, where

$\underline{F} = \int(\underline{n} \cdot \underline{\underline{\sigma}})dA$. The total stress $\underline{\underline{\sigma}}$ can be decomposed as a sum of the pressure and the shear stress contributions ($\underline{\underline{\sigma}} = P\underline{\underline{I}} + \underline{\underline{\tau}}$). When the flow becomes unstable, the stationary vortices behind the cylinder will be evolved to develop a shedding frequency f_q. This dimensionless vortex shedding frequency is called the Strouhal number and is defined as $S_t = \frac{f_q}{u_\infty D}$. Denote the dimensionless time period as T_p, one can express S_t as $S_t = \frac{1}{u_\infty T_p D}$.

In a rectangular domain, flow over a stationary cylinder will be simulated in Cartesian meshes. A constant velocity profile $u_\infty = 1$ was specified at the inlet, convective boundary was specified at the outlet, and the Neumann boundary condition was prescribed along the lateral boundaries. One circular cylinder of diameter $D = 1$ was placed inside the flow domain ($0 \leqslant x \leqslant 40D, -10D \leqslant y \leqslant 10D$) with its center located at $x = 9.5D$ and $y = 0$. Our simulations were performed at three Reynolds numbers ($Re = \frac{\rho u_\infty D}{\mu} = 40, 100, 200$), where the characteristic length and velocity are the diameter of the cylinder (D) and the constant velocity at the inlet (u_∞), respectively. The dimensionless time was defined as $T = \frac{u_\infty t}{D}$. The meshes with 600×300 nodal points are chosen respectively for $Re = 40$, and 700×350 nodal points for $Re = 100$ and $Re = 200$, in the streamwise (x) and transverse (y) directions.

For the case with $Re = 40$, the wake predicted behind the cylinder was seen to be symmetric and steady. This predicted result is in good agreement with the well-established results [42,43,17,28,44–46], which were obtained by the linear stability theory [42,43]. Fig. 11 plots the streamline vector and the corresponding vorticity at $T = 200$ for $Re = 40$. Comparison was

Table 4
Comparison of the results, predicted at the characteristic lengths defined in Fig. 11, with other numerical and experimental results at $Re = 40$.

Authors	C_D	L_w	a	b	θ
Tritton [42]†	1.59	–	–	–	–
Coutanceau and Bouard † [43]	–	2.13	0.76	0.59	53.8°
Ye et al. [17]	1.52	2.27	–	–	–
Tseng and Ferziger [28]	1.53	2.21	–	–	–
Linnick and Fasel [44]	1.54	2.28	0.72	0.60	53.6°
Choi et al. [45]	1.52	2.25	–	–	51.0°
Taira and Colonius [46]	1.54	2.30	0.73	0.60	53.7°
Present	1.52	2.27	0.73	0.60	53.6°

† indicate the experimental data.

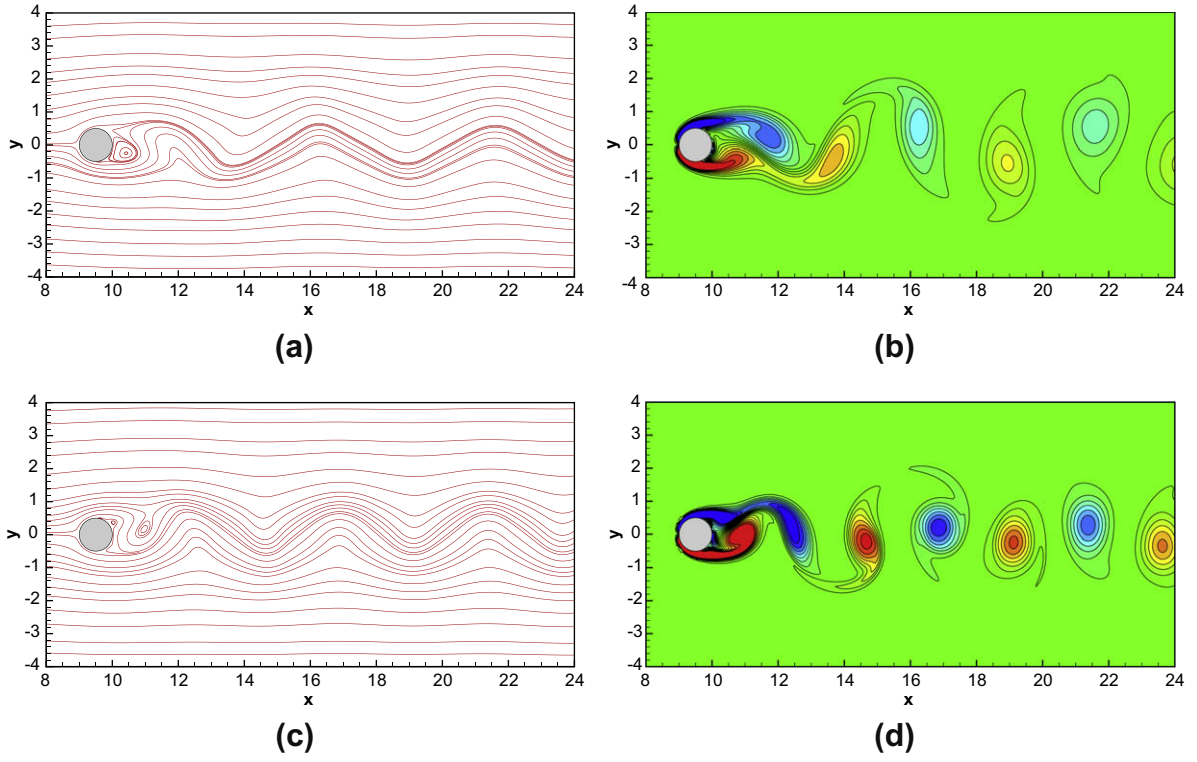

Fig. 12. The predicted results in the near wake of the investigated circular cylinder at $T = 200$. (a) Streamline contours, $Re = 100$; (b) vorticity contours, $Re = 100$; (c) streamline contours, $Re = 200$; (d) vorticity contours, $Re = 200$.

made for the drag coefficient, the length of bubble recirculation (L_w), distance from the cylinder to the center of the wake vortex (a), the gap between the centers of wake vortex (b), and the separation angle (θ) measured from the x-axis, schematic in Fig. 11. Table 4 shows that the results obtained from the currently proposed method agree quite well with other formerly cited results given in [42,43,17,28,44–46].

The cylinder wake instabilities become observed as $Re \geqslant 47$. This is indeed what we predict from the simulation carried out at $Re = 100$ and 200. Fig. 12 plots the streamline vector and the corresponding vorticity at $T = 200$ for $Re = 100$ and 200, respectively. The predicted vortex shedding in Fig. 12 confirms that the proposed method can predict the unsteady flow well.

Comparison of the predicted drag coefficient, lift coefficient, and Strouhal number with other established results in [47–52,45,46] at $Re = 100$ and $Re = 200$ is presented in Table 5. The results obtained from the currently proposed method are in good agreement with the formerly cited results. For the sake of completeness, we also plot the Strouhal numbers for the cases investigated at $Re = 60$, 80, 100, 200. Good agreement with the solutions of Williamson and Brown [53] and Stålberg et al. [52], shown in Fig. 13, confirms the fidelity of the proposed scheme.

6.3. Flow over two cylinders in tandem

Flow over two cylinders in tandem will be simulated next in the Cartesian mesh. The distance between the two cylinders is denoted by the dimensionless quantity $g^* = \frac{G}{D}$, where G is the minimum distance between the cylinders and D is the cylinder diameter. It has been known from the experimental result of Zdravkovich [54] that when the cylinder spacing g^* becomes larger than 3.8, eddy from the upstream cylinder will be shed in synchronization with the downstream one. To

Table 5
Comparison of the predicted drag coefficients, lift coefficients, and Strouhal numbers (frequency) with other numerical results carried out at $Re = 100$ and 200.

	$Re = 100$			$Re = 200$		
Authors	C_D	C_L	S_t	C_D	C_L	S_t
Rosenfeld et al. [47]	–	–	–	1.31 ± 0.04	±0.65	0.20
Liu et al. [48]	1.35 ± 0.012	±0.339	0.164	1.31 ± 0.049	±0.69	0.192
Wright and Smith [49]	–	–	–	1.33 ± 0.04	±0.68	0.196
Calhoun [50]	1.35 ± 0.014	±0.3	0.175	1.17 ± 0.058	±0.67	0.202
Russell and Wang [51]	1.38 ± 0.007	±0.322	0.169	1.29 ± 0.022	±0.50	0.195
Stålberg et al. [52]	1.32 ± 0.009	±0.33	0.166	–	–	–
Choi et al. [45]	1.34 ± 0.011	± 0.315	0.164	1.36 ± 0.048	± 0.64	0.191
Taira and Colonius [46]	–	–	–	1.34 ± 0.047	±0.68	0.195
Present	1.35 ± 0.012	±0.303	0.167	1.37 ± 0.051	±0.71	0.198

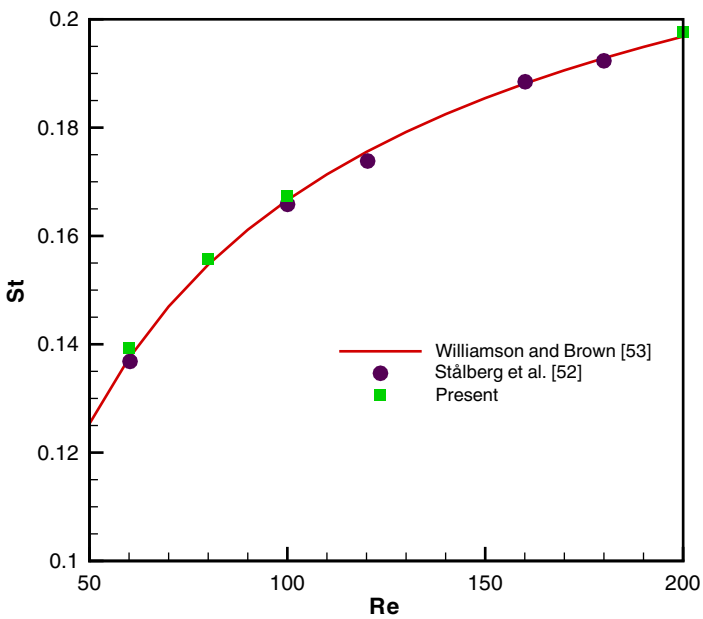

Fig. 13. Comparison of the predicted Strouhal numbers at different Reynolds numbers Re for the flow over a cylinder.

demonstrate that the current method can be used to predict the flow in complex geometry, flow over two cylinders in tandem will be analyzed at $Re = 200$ and $g^* = 4.0$.

A constant velocity $u_\infty = 1$ was specified at the inlet, convective boundary was specified at the outlet, and a Neumann boundary condition was prescribed along the lateral boundaries. Two circular cylinders of diameter $D = 1$ were placed inside the domain with their centers located respectively at $(x, y) = (9.5D, 0)$ and $(14.5D, 0)$. Our simulations have been carried out in the domain of $0 \leqslant x \leqslant 45D$, $-10D \leqslant y \leqslant 10D$. The resulting truncated boundaries are sufficiently apart from the cylinders and can minimize the boundary effect on the flow development. The mesh is chosen to have 810×360 nodal points in the streamwise (x) and transverse (y) directions, respectively.

Table 6
Comparison of the predicted average drag coefficients, lift coefficients, and Strouhal numbers for two cylinders in tandem investigated at $g^* = 4.0$ and $Re = 200$. Index 1 refers to the upstream circular cylinder and index 2 indicates the downstream circular cylinder.

Authors	C_{D_1}	C_{L_1}	S_{t_1}	C_{D_2}	C_{L_2}	S_{t_2}
Farrant et al. [55]	1.25	0.71	0.179	0.38	1.59	0.179
Meneghini et al. [56]	1.18	–	0.174	0.38	–	0.174
Present	1.27	0.65	0.180	0.35	1.23	0.180

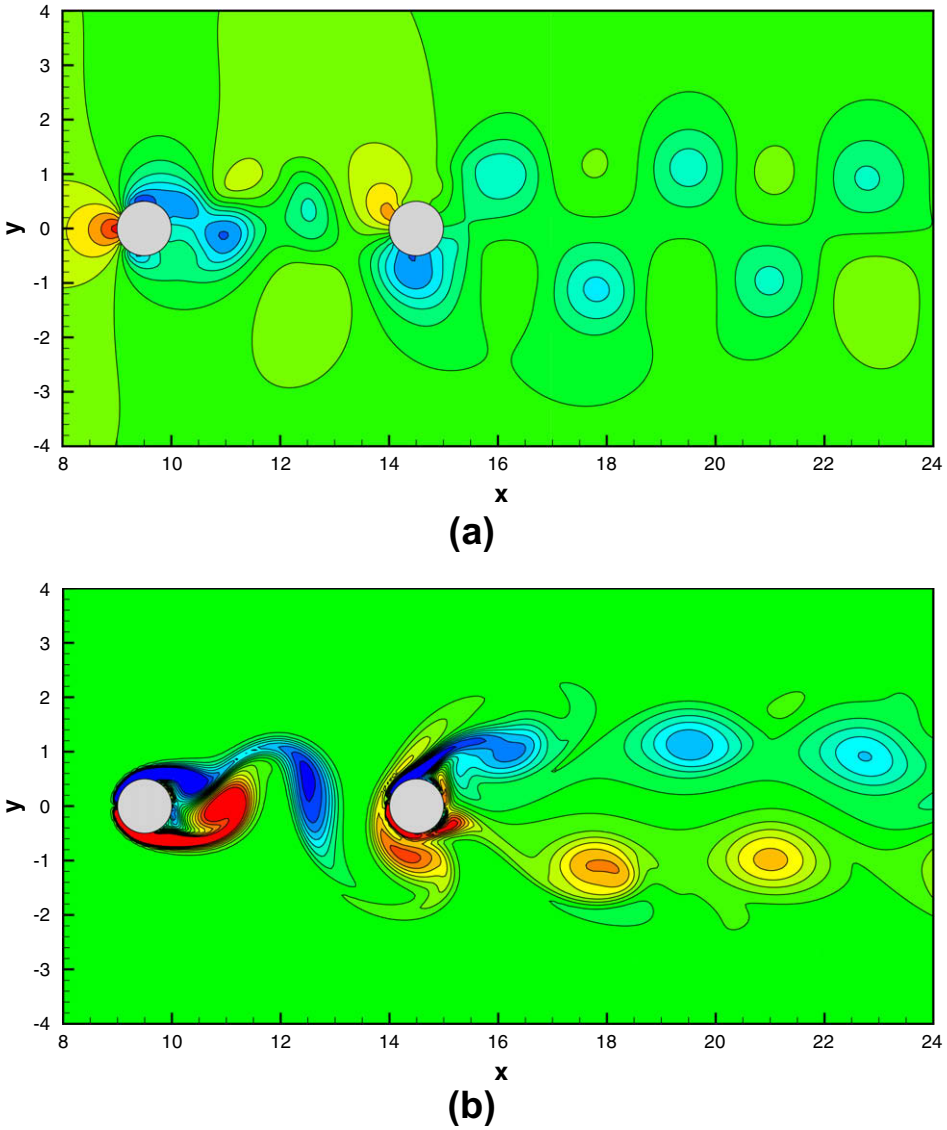

Fig. 14. The predicted results at $T = 400$ in the near wake of two circular cylinders. (a) Pressure contours; (b) vorticity contours.

For the two investigated cylinders in tandem, calculation will be carried out at $Re = 200$ and $g^* = 4.0$, which were investigated before by Farrant et al. [55] and Meneghini et al. [56]. Comparison of the predicted time-averaged drag coefficient, lift coefficient, and Strouhal number with the established results can be seen in Table 6. Good agreement with other non-Cartesian numerical simulations confirms that the proposed method implemented in Cartesian grids can be applied to predict the flows in complex geometries. Also, the Strouhal number, which is corresponding to the dominant frequency of the lift variation, for both cylinders has the same value (that is 0.180) shown in Table 6. The predicted identical Strouhal numbers confirm that the shedding is synchronized and their values are in agreement with the experimental data. In order to reveal

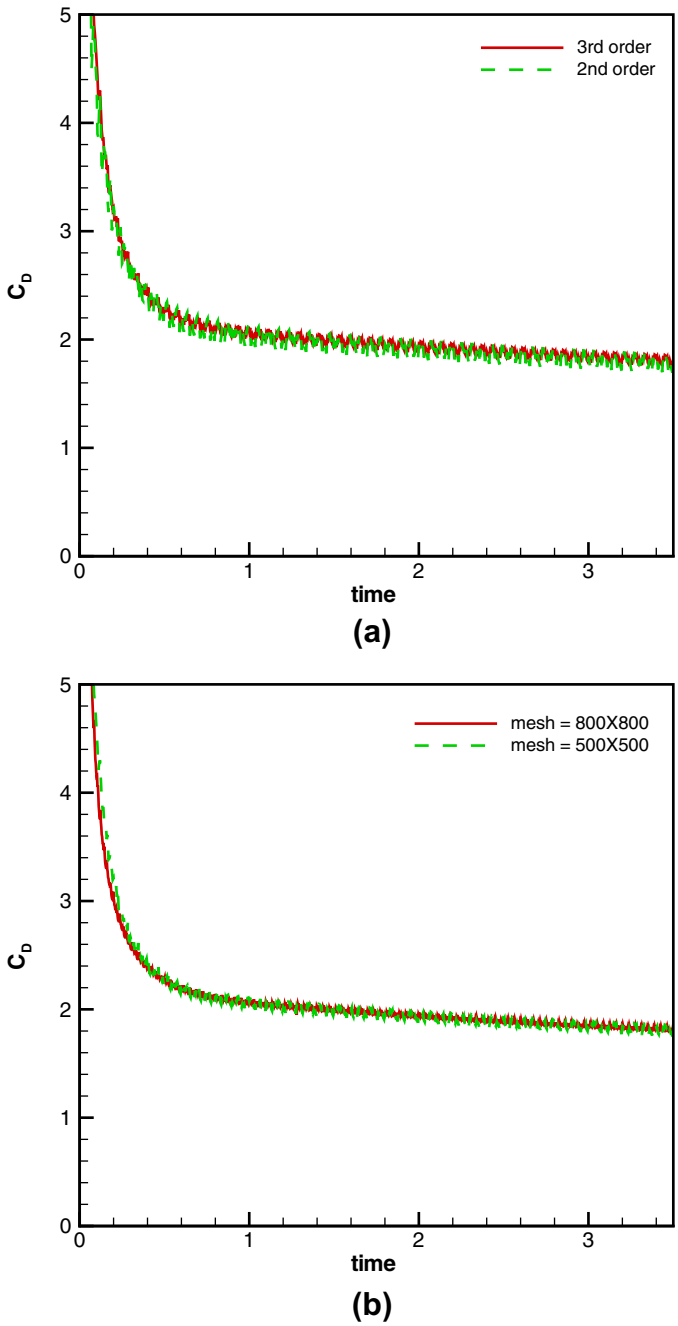

Fig. 15. Comparison of the predicted time evolving drag coefficients (C_D) at different meshes using different interpolation schemes. (a) Third-order and second-order extrapolations in a domain of 500×500 mesh points; (b) third-order order extrapolations in a domain of 500×500 and 800×800 mesh points.

the behavior of the cylinders in tandem, the streamline vector, vorticity contours, and the corresponding pressure and vorticity predicted at $T = 400$ are also plotted in Fig. 14.

6.4. Flow around a moving cylinder

We will then simulate flow over a constantly moving circular cylinder. Initially, the cylinder of unit diameter (it D = 1) is stationarily placed at the origin. The flow is then set into motion towards the left with a constant velocity of $u_{body} = 1$. The Reynolds number under investigation is $Re = u_{body}D/v = 40$. In the present study, our simulations have been carried out in a computational domain $(-16.5D \leqslant x \leqslant 13.5D, -15D \leqslant y \leqslant 15D)$ with the no-slip condition being applied along the boundary.

The field-extension procedure of Yang and Balaras [57] has been reported to have the ability to eliminate the spurious values near the moving boundary. The predicted drag coefficients are, however, seen to oscillate with respect to time if the second-order extrapolation (Eq. (18)) is used for $\underline{u}$. In order to suppress the oscillation, we propose the third-order extrapolation given by

$$\frac{\partial u}{\partial \tau'} + n_x(\Delta x^2 u'_{xxx}) + n_y(\Delta y^2 u'_{yyy}) = 0 \tag{77}$$

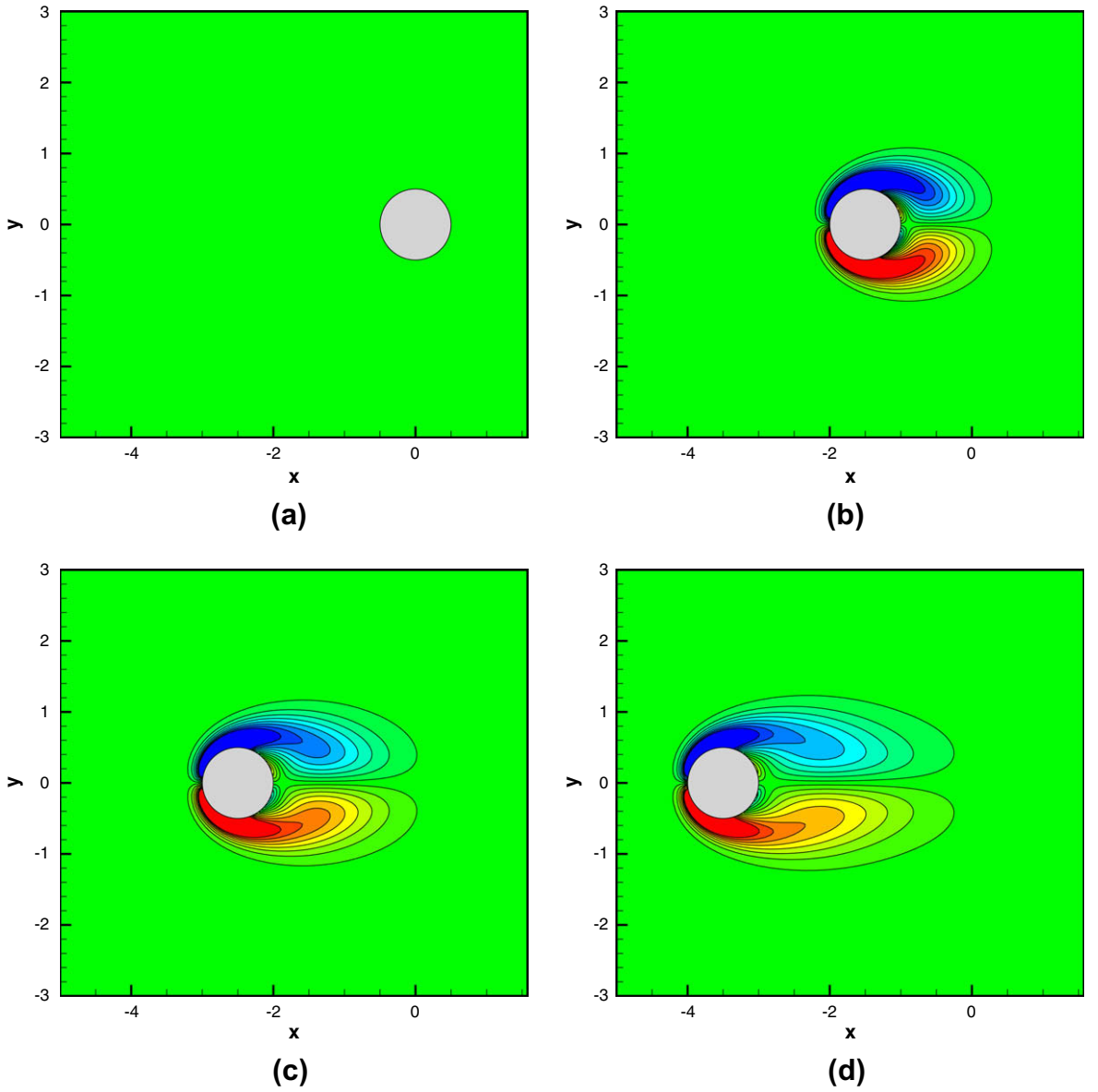

Fig. 16. The predicted instantaneous vorticity contours at $Re = 40$ for the flow around a moving cylinder. (a) $t = 0$; (b) $t = 1.5$; (c) $t = 2.5$; (d) $t = 3.5$.

where u'_{xxx} and u'_{yyy} are as follows:

$$u'_{xxx}(i,j) = u^{+}_{xxx}(i,j) if n_x(i,j) >= 0;\ \text{else}\ u'_{xxx}(i,j) = u^{-}_{xxx}(i,j)$$

$$u'_{yyy}(i,j) = u^{+}_{yyy}(i,j) if n_y(i,j) >= 0;\ \text{else}\ u'_{yyy}(i,j) = u^{-}_{yyy}(i,j)$$

and

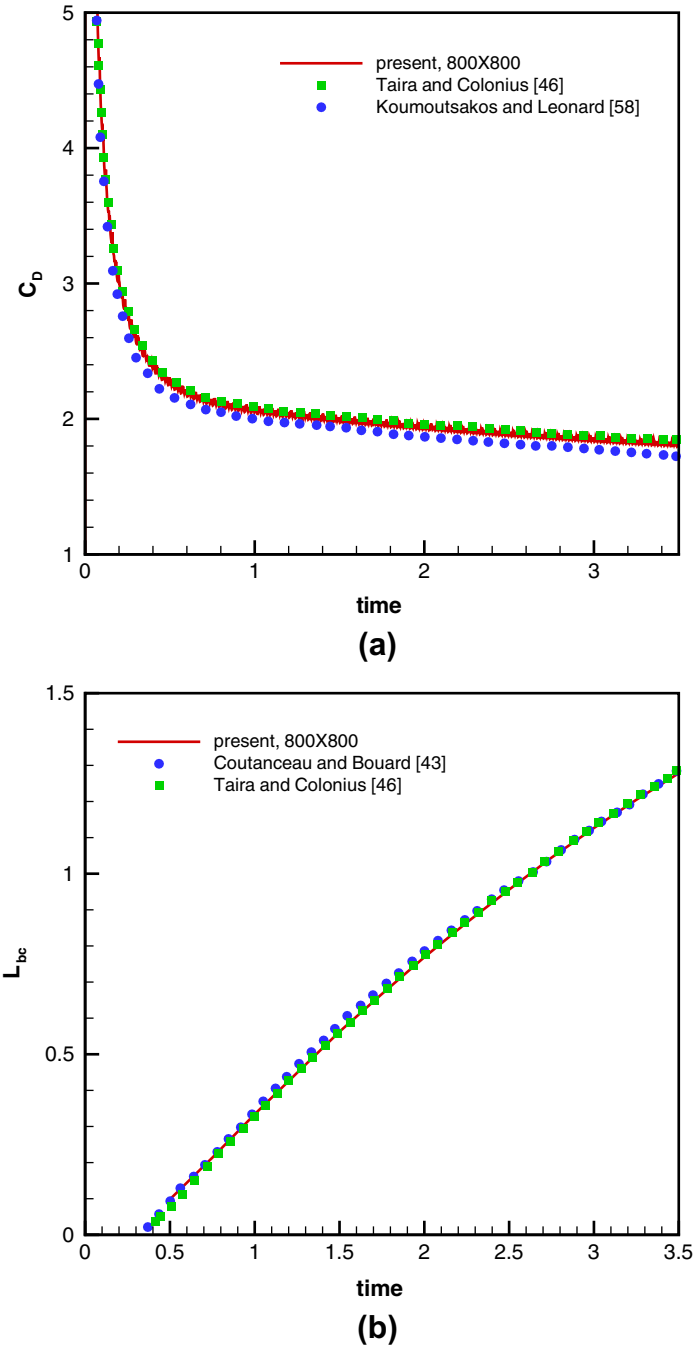

Fig. 17. Comparison of the predicted time evolving drag coefficients (C_D) and the lengths of recirculation bubble (L_{bc}). Note that L_{bc} is evaluated by an observer sitting on the moving cylinder. (a) C_D; (b) L_{bc}.

$$u^{+}_{xxx}(i,j) = (u_{i,j} - 3u_{i-1,j} + 3u_{i-2,j} - u_{i-3,j})/\Delta x^3$$
$$u^{-}_{xxx}(i,j) = (u_{i+3,j} - 3u_{i+2,j} + 3u_{i+1,j} - u_{i,j})/\Delta x^3$$
$$u^{+}_{yyy}(i,j) = (u_{i,j} - 3u_{i,j-1} + 3u_{i,j-2} - u_{i,j-3})/\Delta y^3$$
$$u^{-}_{yyy}(i,j) = (u_{i,j+3} - 3u_{i,j+2} + 3u_{i,j+1} - u_{i,j})/\Delta y^3$$

It can be seen from Fig. 15 that application of the proposed third-order extrapolation helps to suppress the oscillation. In addition, when the grid is refined to have 800×800 nodal points, the oscillation can be well suppressed. Comparisons are made in the time interval ranging from the non-dimensional time $t = 0$ to 3.5 for the C_D coefficient and the bubble recirculation length defined in Section 6.2 in the frame of reference of the cylinder $(u - u_{body}, v)$. The results shown in Figs. 16 and 17 are seen to agree well with the results predicted in [46,43,58].

7. Concluding remarks

In this study, an immersed boundary method is proposed for the simulation of steady incompressible viscous semi-circular lid-driven cavity problem, and time-dependent (unsteady) incompressible viscous flows over a circular cylinder, two cylinders in tandem, and around a moving cylinder in Cartesian grids. For the sake of accuracy, a dispersion-relation-preserving dual-compact scheme is developed to approximate the first-order derivative and the second derivative terms. The differentially-based interpolation is proposed to improve the classical algebraic-based interpolation. All the results compare favorably with the experimental and other numerical results. Moreover, the present method can be easily extended to simulate the three-dimensional problem since there is no need to reconstruct the 3D algebraic-based interpolation.

Acknowledgments

The financial supports provided by the National Science Council under grant NSC96-2221-E-002-004 and NTU-CQSE (97R0066-69) are gratefully acknowledged.

References

[1] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (1972) 252–271.
[2] R. Glowinski, T.W. Pan, T. Hesla, D.D. Joseph, J. Periaux, A fictious domain approach to the direct numerical simulation of incompressible viscous flows past moving rigid bodies: application to particulate flow, J. Comput. Phys. 169 (2001) 363–426.
[3] R. Vikhansky, A new modification of the immersed boundaries method for fluid–solid flows: modernate Reynolds numbers, J. Comput. Phys. 191 (2003) 328–339.
[4] R. Cortez, M. Minion, The blob projection method for immersed boundary problems, J. Comput. Phys. 161 (2000) 428–453.
[5] C.S. Peskin, B.F. Printz, Improved volume conservation in the computation of flows with immersed elastic boundaries, J. Comput. Phys. 105 (1993) 33–46.
[6] S. Xu, Z.J. Wang, An immersed interface method for simulating the interaction of a fluid with moving boundaries, J. Comput. Phys. 216 (2006) 454–493.
[7] C. Tu, C.S. Peskin, Stability and instability in the computation of flows with moving immersed boundaries, SIAM J. Sci. Statist. Comput. 13 (1992) 1361–1376.
[8] Z. Li, R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (2003) 832–856.
[9] E.P. Newren, A.L. Fogelson, R.D. Guy, R.M. Kirby, Unconditionally stable discretizations of the immersed boundary equations, SIAM J. Sci. Comput. 25 (2003) 832–856.
[10] J.M. Stockie, B.R. Wetton, Analysis of stiffness in the immersed boundary method and implications for time-stepping schemes, J. Comput. Phys. 154 (1999) 41–64.
[11] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[12] D. Goldstein, R. Haandler, L. Sirovich, Modeling a no-slip flow boundary with an external force field, J. Comput. Phys. 105 (1993) 336–354.
[13] E.M. Saiki, S. Biringen, Numerical simulation of a cylinder in uniform flow: application of a virtual boundary method, J. Comput. Phys. 123 (1996) 450–465.
[14] A.S. Almgren, J.B. Bell, P. Colella, T. Marthaler, A Cartesian grid projection method for the incompressible Euler equations in complex geometries, SIAM J. Sci. Comput. 18 (1997) 1289–1309.
[15] J. Mohd-Yusof, Combined immersed boundary/B-spline method for simulations of flows in complex geometries, CTR Annual Research Briefs, NASA Ames/Stanford University, 1997, pp. 317-327.
[16] H.S. Udaykumar, R. Mittal, W. Shyy, Computation of solid–liquid phase fronts in the sharp interface limit on fixed grids, J. Comput. Phys. 153 (1999) 535–574.
[17] 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.
[18] E.A. Fadlun, R. Verzicco, P. Orlandi, J. Mohd-Yusof, Combined immersed-boundary methods for three dimensional complex flow simulations, J. Comput. Phys. 161 (2000) 30–60.
[19] J. Kim, D. Kim, H. Choi, An immersed boundary finite-volume method for simulations of flows in complex geometries, J. Comput. Phys. 171 (2001) 132–150.
[20] F. Gibou, R.J. Fedkiw, L.T. Cheng, M. Kang, A second-order-accurate symmetric discretization of the Poisson equation on irregular domains, J. Comput. Phys. 176 (2002) 205–227.
[21] D. You, R. Mittal, M. Wang, P. Moin, Study of rotor tip-clearance flow using large-eddy simulation, AIAA J. 0838 (2003).
[22] E. Balaras, Modeling complex boundaries using an external force field on fixed Cartesian grids in large-eddy simulations, Comput. Fluids 33 (2004) 375–404.
[23] R. Ghias, R. Mittal, H. Dong, A sharp interface immersed boundary method for compressible viscous flows, J. Comput. Phys. 225 (2007) 528–553.
[24] H.S. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface Cartesian grid method for simulating flows with complex moving boundaries, J. Comput. Phys. 174 (2001) 345–380.
[25] L. Lee, R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (2003) 832–856.

[26] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), J. Comput. Phys. 152 (1999) 457–492.
[27] Y. Hao, A. Prosperetti, A numerical method for three-dimensional gas–liquid flow computations, J. Comput. Phys. 196 (2004) 126–144.
[28] Y.H. Tseng, J.H. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comput. Phys. 192 (2003) 593–623.
[29] Morten M.T. Wang, Tony W.H. Sheu, An element-by-element BICGSTAB iterative method for three-dimensional steady Navier–Stokes equations, J. Comput. Appl. Math. 79 (1997) 147–165.
[30] L. Quartapelle, M. Napolitano, Integral conditions for the pressure in the computation of incompressible viscous flows, J. Comput. Phys. 62 (1986) 340–348.
[31] P. Lin, A sequential regularization method for time-dependent incompressible Navier–Stokes equations, SIAM J. Numer. Anal. 34 (1997) 1051–1071.
[32] C.W. Li, L.L. Wang, An immersed boundary finite difference method for LES of flow around bluff shapes, Int. J. Numer. Methods Fluids 46 (2004) 85–107.
[33] T. Gao, Y.H. Tseng, X.Y. Lu, An improved hybrid Cartesian/immersed boundary method for fluid–solid flows, Int. J. Numer. Methods Fluids 55 (2007) 1189–1211.
[34] S. Majumdar, G. Iaccarino, P. Durbin, RANS solvers with adaptive structured boundary non-conforming grids, Annual Research Briefs. NASA Ames Research Center/Stanford University Center for Turbulence Research, Stanford, CA, 2001, pp. 353–366.
[35] T.D. Aslam, A partial differential equation approach to multidimensional extrapolation, J. Comput. Phys. 193 (2003) 349–355.
[36] C.K.W. Tam, J.C. Webb, Dispersion-relation-preserving finite difference schemes for computational acoustics, J. Comput. Phys. 107 (2003) 262–281.
[37] Tony W.H. Sheu, R.K. Lin, Newton linearization on the incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 44 (2004) 297–312.
[38] Tony W.H. Sheu, R.K. Lin, An incompressible Navier–Stokes model implemented on non-staggered grids, Numer. Heat Transf., B Fundam. 44 (2003) 277–294.
[39] P.C. Chu, C. Fan, A three-point combined compact difference scheme, J. Comput. Phys. 140 (1998) 370–399.
[40] P.H. Chiu, Tony W.H. Sheu, R.K. Lin, An effective explicit pressure gradient scheme implemented in the two-level non-staggered grids for incompressible Navier–Stokes equations, J. Comput. Phys. 227 (2008) 4018–4037.
[41] R. Glowinski, G. Guidoboni, T.W. Pan, Wall-driven incompressible viscous flow in a two-dimensional semi-circular cavity, J. Comput. Phys. 216 (2006) 79–91.
[42] D.J. Tritton, Experiments on the flow past a circular cylinder at low Reynolds numbers, J. Fluid Mech. 6 (1959) 547–567.
[43] 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.
[44] 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.
[45] J.I. Choi, R.C. Oberoi, J.R. Edwards, J.A. Rosati, An immersed boundary method for complex incompressible flows, J. Comput. Phys. 224 (2007) 757–784.
[46] K. Taira, T. Colonius, The immersed boundary method: a projection approach, J. Comput. Phys. 225 (2007) 2118–2137.
[47] M. Rosenfeld, D. Kwak, M. Vinokur, A fractional step solution method for the unsteady incompressible Navier–Stokes equations in generalized coordinate systems, J. Comput. Phys. 94 (1991) 102–137.
[48] C. Liu, X. Sheng, C.H. Sung, Preconditioned multigrid methods for unsteady incompressible flows, J. Comput. Phys. 139 (1998) 35–57.
[49] J.A. Wright, R.W. Smith, An edge-based method for the incompressible Navier–Stokes equations on polygonal meshes, J. Comput. Phys. 169 (2001) 24–43.
[50] D. Calhoun, A Cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions, J. Comput. Phys. 176 (2002) 231–275.
[51] D. Russell, Z.J. Wang, A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow, J. Comput. Phys. 191 (2003) 177–205.
[52] E. Stålberg, A. Brüger, P. Lötstedt, A.V. Johansson, D.S. Henningson, High order accurate solution of flow past a circular cylinder, J. Sci. Comput. 27 (2006) 431–441.
[53] C.H.K. Williamson, G.L. Brown, A series in $1/\sqrt{Re}$ to represent the Strouhal–Reynolds number relationship of the cylinder wake, J. Fluids Struct. 12 (1998) 1073–1085.
[54] M.M. Zdravkovich, Review of flow interference between two circular cylinders in various arrangements, J. Fluids Eng. 99 (1977) 618–633.
[55] T. Farrant, M. Tan, W.G. Tan, A cell boundary element method applied to laminar vortex shedding from circular cylinders, Comput. Fluids 30 (2001) 211–236.
[56] J.R. Meneghini, F. Saltara, C.L.R. Siqueira, J.A. Ferrari, Numerical simulation of flow interference between two circular cylinders in tandem and side-by-side arrangements, J. Fluids Struct. 15 (2001) 327–350.
[57] 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.
[58] P. Koumoutsakos, A. Leonard, High-resolution simulations of the flow around an impulsively started cylinder using vortex methods, J. Fluid Mech. 296 (1995) 1–38.