Journal of Computational Physics 374 (2018) 912–940

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

An efficient, high-order method for solving Poisson equation for immersed boundaries: Combination of compact difference and multiscale multigrid methods

Shirzad Hosseinverdi *, Hermann F. Fasel

Department of Aerospace and Mechanical Engineering, University of Arizona, Tucson, AZ 85721, USA

ARTICLE INFO

Article history:
Received 3 November 2017
Received in revised form 31 July 2018
Accepted 1 August 2018
Available online 6 August 2018

Keywords:
High-order
Poisson equation
Immersed boundary
Compact finite difference
Multiscale multigrid method

ABSTRACT

A new efficient and high-order accurate sharp-interface method for solving the Poisson equation on irregular domains and non-uniform meshes is presented. The approach is based on a combination of a fourth-order compact finite difference scheme and a multiscale multigrid (MSMG) method. The key aspect of the new method is that the regular compact finite difference stencil is modified at the irregular grid points near the immersed boundary to obtain a sharp interface solution while maintaining the formal fourth-order accuracy. The MSMG method is designed based on the standard multigrid V-cycle technique to solve the system of equations derived from the fourth-order compact discretization, while the corresponding multigrid relaxation, restriction and prolongation operators are properly constructed for non-uniform grids with immersed boundaries. The contribution of the present work is the design of a fourth-order-accurate Poisson solver whose accuracy, efficiency and computational cost are independent of the complexity of the geometry and the presence or not of an immersed boundary. The new method is demonstrated and validated for a number of problems including smooth and jagged boundaries. The test cases confirm that the new method is fourth-order accurate in the maximum norm whether an immersed boundary is present or not and on uniform or non-uniform meshes. Furthermore, the computational efficiency of the new method is demonstrated with regard to convergence rate and run time, which shows that the MSMG method is equally efficient for domains with immersed boundaries as for simple domains. The new compact difference method is evaluated by comparison with the standard fourth-order (non-compact) finite difference approximation in terms of both accuracy and computational efficiency. The new compact difference scheme yields indeed more accurate numerical solutions. The striking difference between the two schemes is the much higher computational efficiency: The number of V-cycles needed to reach the discretization error is significantly lower for the new compact method compared to the standard difference scheme. As a result, the new compact scheme requires only a fraction of the computer time for convergence in comparison to the standard fourth-order difference scheme.

* Corresponding author.
E-mail address: shirzadh@email.arizona.edu (S. Hosseinverdi).

https://doi.org/10.1016/j.jcp.2018.08.006
0021-9991/

1. Introduction

In this paper, we present a new numerical method for solving the Poisson (Helmholtz type) equation,

$$\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}-\beta\right)u(x,y)=\rho(x,y),\quad (x,y)\in\Omega, \tag{1}$$

with Dirichlet boundary conditions, $u(\gamma)=f(\gamma)$ where $\gamma\in\partial\Omega$. Here Ω is a bounded two-dimensional domain and $\partial\Omega$ is the boundary of Ω. The domain Ω can contain an arbitrary immersed body with boundary Γ, in which case the domain is divided by the curve Γ into two subdomains Ω^+ and Ω^- as shown in Fig. 1(a). Referring to Fig. 1(a), one would typically solve Eq. (1) defined on the open region Ω^+ with boundary conditions on $\partial\Omega$, the outer boundary which conforms to the computational boundary, and Γ, the immersed boundary. The solution in the region Ω^- may or may not be of interest. In either case, the immersed boundary Γ represents a singularity if one considers that Eq. (1) applies throughout the entire domain enclosed by $\partial\Omega$; field variables and/or their derivatives will be discontinuous across the immersed boundary.

Eq. (1) is an elliptic partial differential equation with broad applications in mechanical and aerospace engineering, theoretical physics and other fields. Moreover, Eq. (1) represents the Helmholtz type equation for nonzero β, which governs some important physical phenomena like wave propagation and scattering phenomena arising in many areas, as for example in fluid dynamics and aerodynamics. Therefore, there is great interest to develop accurate, fast and efficient methods for the numerical solution of Eq. (1) for complex geometries.

Traditional approaches in Computational Fluid Dynamics (CFD) use body-fitted structured or unstructured grids. However, generating high-quality grids is generally cumbersome and it becomes very laborious as the complexity of the geometry increases. The grid generation process becomes a very difficult task, even for the simplest geometries, when more than one body is located within the domain. Grids with poor qualities (smoothness, orthogonality, aspect ratio) could negatively impact the accuracy and convergence properties of the Poisson solver.

An alternative approach is to use the so-called immersed boundary method (IBM) where Eq. (1) is discretized on a fixed Cartesian grid which is allowed to intersect with Γ. As a result, the grid does not conform to the solid boundaries, as shown in Fig. 1(b), for example. The main advantage of the IBM lies in the use of Cartesian grids for which there is almost no grid generation cost. Moreover, many efficient numerical methods such as high-order methods, fast Poisson solvers, structured multigrid solvers, and many others have been developed for Cartesian grids. A disadvantage of the IBM is that the imposition of boundary conditions on the immersed boundaries is not straightforward as compared to traditional body fitted methods. Also, the influence of the boundary treatment on accuracy and conservation properties of the numerical schemes is not trivial. Several methods have been proposed in the past to handle the singularity associated with the immersed boundary. Generally, they can be classified as either continuous forcing (diffuse) approaches or discrete forcing (sharp) methods [1,2].

In the continuous methods, the boundary conditions are enforced through a smooth forcing term added to the equations. Peskin [3,4] pioneered this type of IBM to handle elastic boundaries for simulating blood flow in the heart. The method was later extended to simulate flow with rigid boundaries (see for instance [5,6]). One disadvantage of diffuse methods is that the effect of the boundary is distributed over a band of several grid points which causes a smearing effect near the boundary. This smearing has a detrimental effect on the accuracy of the numerical scheme.

In the other approach, the discrete IBM, the numerical discretization near the immersed boundary is modified such as to account directly for the presence of the boundary, so that the interface remains “sharp”. There are several different implementation strategies of the sharp-interface approach. One of them is the so-called “ghost cell” approach. Ghost cells are defined as cells in the solid region that have at least one neighbor in the fluid. Boundary conditions at the immersed boundary is enforced through the ghost cells. For each ghost cell, an interpolation scheme that implicitly incorporates the boundary condition on the immersed boundary is then devised. A number of options are available for constructing the interpolation scheme [7,8]. While the approach is well suited for achieving second-order accuracy, extension to higher-order formulations is problematic. Higher-order formulations require large interpolation stencils which could lead to robustness issues. Furthermore, obtaining the ghost cell value is not that obvious for more irregularly shaped geometries, e.g. for sharp corners.

Another method that falls into the sharp category is the so called “cut-cell” method, often used within the finite-volume framework [9,10]. Being based on the finite volume approach, this method is designed to provide better conservation properties, especially for cells at the immersed boundary. Grid cells cut by the immersed boundary are reshaped to conform to the boundary. This reshaping may in some cases result in very small grid cells with an adverse impact on numerical stability. To overcome this problem, cell-merging strategies have been successfully proposed. However, due to the many possible configurations of the irregular cut-cell, particularly in three dimensions, implementation becomes a tedious and non-trivial task.

An alternative to the IBM is the sharp immersed interface method (IIM) introduced by LeVeque and Li [11] to solve elliptic problems with discontinuous and non-smooth solutions. The basic idea behind the IIM approach, first discussed by LeVeque and Li [11], and later clarified by Wiegmann and Bube [12] is to recognize that standard finite difference techniques fail when applied to non-smooth functions because the underlying Taylor expansions upon which they are based are invalid. The key idea of the IIM is that correction terms are added to the finite difference schemes at the interface of the immersed boundary to account for any jump in the solution or its derivative in order to maintain the formal accuracy of the underlying numerical scheme.

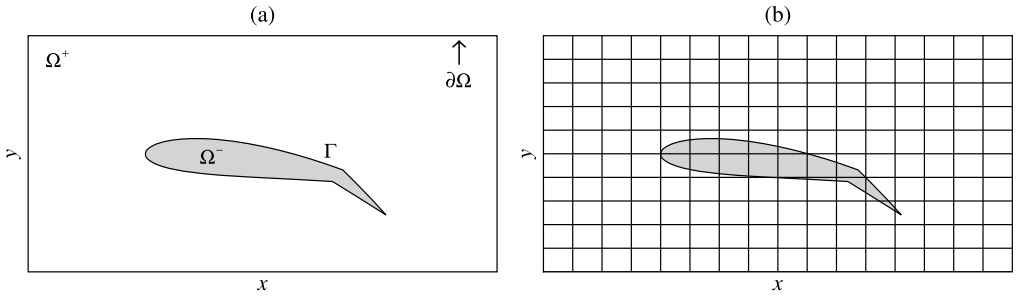

Fig. 1. (*a*) Schematic of arbitrary geometry immersed in the domain Ω. (*b*) Schematic of the immersed body in a Cartesian grid on which Eq. (1) is discretized.

An important application of sharp IIM is the treatment of problems defined in an irregular domain where the solution inside or the outside the interface is not of interest and is trivial. In such cases, higher order discretizations can be used near the interface. Linnick and Fasel [13] and Hosseinverdi and Fasel [14] developed a fourth-order accurate schemes based on the sharp IIM to solve the incompressible Navier-Stokes Equations in the stream-function-vorticity and velocity-vorticity formulations, respectively. They were able to accurately simulate Tollmien-Schlichting waves in a Blasius boundary-layer in addition to computing the flow past a circular cylinder and an elliptic airfoil. It is important to note, however, that the Poisson solver in Linnick and Fasel [13] IIM Navier-Stokes code required 60% to 70% more run time than the solver without immersed boundary [15].

Apart from the IBM and IIM, other approaches have been proposed in the literature to solve the elliptic equations with discontinuous coefficients or irregular boundaries such as the ghost fluid method of Gibou and Fedkiw [16], the matched interface method [17–19], [20, Ch. 7] and others [21,22]. However, what has not yet been addressed is the question of how to solve the Poisson equation highly accurately and still be as computationally efficient as for the case without immersed boundary. In fact, the Poisson equation with Dirichlet boundary conditions represents one of the major computational challenges when solving the incompressible Navier–Stokes Equations (INSE) in the stream function–vorticity and/or velocity–vorticity formulations as the majority of the computational time for solving the INSE is spent solving the Poisson equation(s) [23,24]. Thus, it is evident that solving the Poisson equation must be as computationally efficient as possible.

The objective of the present work is to design a highly efficient fourth-order-accurate method to solve Eq. (1) in a domain with immersed boundaries on a non-uniform mesh. To achieve this goal, a fourth-order compact finite-difference scheme is combined with the multiscale multigrid (MSMG) method. Our new method falls into the sharp interface category. However, it distinguishes itself from other IIM in that the jump corrections are no longer required. The underlying feature of the new method is that the regular compact finite difference stencil is adjusted at the irregular grid points in the vicinity of the interfaces of the immersed bodies to obtain a solution that is sharp across the interface while keeping the fourth-order global accuracy.

The Multigrid Method is very effective for solving large scale sparse linear system of equations derived from the discretization of Eq. (1). It offers convergence rates independent of the grid size for a prescribed accuracy. However, more iterations are needed to achieve higher accuracy on the finer mesh. In the current work, the standard multigrid V-cycle technique is used to build the MSMG method with the objective to converge to the discretization error in a fixed number of iterations that is independent of the grid size. The performance and efficiency of the MSMG method depends on the proper design of its operators, namely, relaxation, restriction and prolongation. If these operators are not optimally designed, a deterioration of the convergence rate will occur. This is particularly the case when non-uniform grids with immersed boundaries are considered. In this regard, the present work is aimed to develop an efficient solution strategy for the Poisson equation whose computational performance and convergence rate do not deteriorate for irregular domains and for domains with immersed boundaries. The paper is organized as follows: In Section 2, the discretization of Eq. (1) on the non-uniform mesh is presented for regular grid points. The numerical procedure to construct a fourth-order accurate modified compact difference stencil for irregular grid points are discussed in Section 3. A formal proof for the order of accuracy and a spectral analysis of the resulting coefficient matrices for simple and irregular domains are provided in Section 4. The solution strategy of the discretized equations is explained in Section 5. Section 5 also introduces an efficient strategy to design and construct the different operators involved in the MSMG method. Then, in Section 6, the proposed method is validated for several test cases, which demonstrate the high efficiency and confirm the fourth-order convergence. A summary and conclusions are provided in Section 7.

2. Discretization for regular grid points on non-uniform grid: fourth-order compact difference scheme

A rectangular domain Ω defined on $[a, b] \times [c, d]$ is considered. The domain is divided into $M \times N$ uniform/non-uniform cells by the points $a = x_1 < x_2 < \cdots < x_{M-1} < x_M = b$ and $c = y_1 < y_2 < \cdots < y_{N-1} < y_N = d$. Before we proceed with the details of the discretization, different type of grid points need to be defined. A grid point is called a solid point if it lies inside the immersed body. A grid point is said to be regular if all the eight neighboring grid points are outside

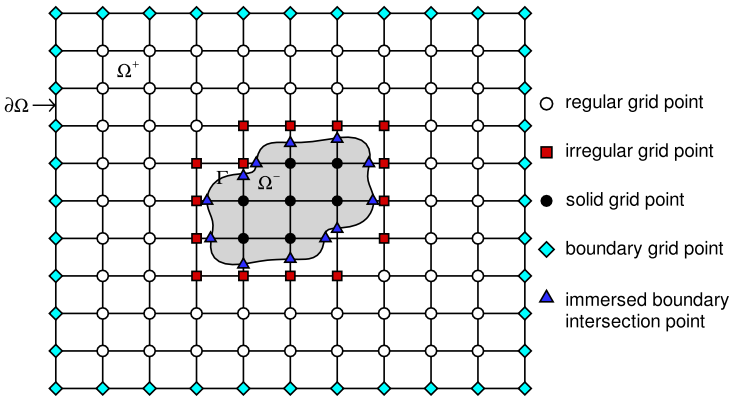

Fig. 2. Representation of the different type of grid points.

the immersed body, otherwise, it is defined as irregular point. Furthermore, the locations where the immersed boundary intersects with the grid are called immersed boundary intersection (IBI) points. The BI points are the locations where the boundary conditions can be enforced. All types of grid points are illustrated in Fig. 2.

In this section, the discretization for regular points is discussed so that the compact difference scheme is well-defined and valid. The discretization of Eq. (1) is based on two 1-D, fourth-order compact finite-difference stencils for second derivatives in x and y:

$$(L_{xx}u_{xx})_j = (R_{xx}u)_j \,, \tag{2}$$

$$(L_{yy}u_{yy})_i = (R_{yy}u)_i \,, \tag{3}$$

where the finite difference (FD) operators are given by

$$(L_{xx}u_{xx})_j = a_i^x\, u_{xx_{i-1,j}} + b_i^x\, u_{xx_{i,j}} + c_i^x\, u_{xx_{i+1,j}} \,, \tag{4}$$

$$(L_{yy}u_{yy})_i = a_j^y\, u_{yy_{i,j-1}} + b_j^y\, u_{yy_{i,j}} + c_j^y\, u_{yy_{i,j+1}} \,, \tag{5}$$

$$(R_{xx}u)_j = ar_i^x\, u_{i-1,j} + br_i^x\, u_{i,j} + cr_i^x\, u_{i+1,j} \,, \tag{6}$$

$$(R_{yy}u)_i = ar_j^y\, u_{i,j-1} + br_j^y\, u_{i,j} + cr_j^y\, u_{i,j+1} \,. \tag{7}$$

Here, u_{xx} and u_{yy} represent numerical approximations to the second partial derivatives in x and y directions, respectively. Matching the Taylor series coefficients about $u_{i,j}$ in the x-direction, the coefficients in Eq. (2) can be found from

$$\begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & -dx_b & 0 & dx_f \\ -1 & -1 & dx_b^2/2 & 0 & dx_f^2/2 \\ dx_b & -dx_f & -dx_b^3/6 & 0 & dx_f^3/6 \\ -dx_b^2/2 & -dx_f^2/2 & dx_b^4/24 & 0 & dx_f^4/24 \end{bmatrix} \begin{bmatrix} a_i^x \\ c_i^x \\ ar_i^x \\ br_i^x \\ cr_i^x \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} , \tag{8}$$

where $dx_f = x_{i+1} - x_i$ and $dx_b = x_i - x_{i-1}$. Solving the above system of equations leads to

$$a_i^x = \frac{dx_f\,(dx_b^2 + dx_b dx_f - dx_f^2)}{(dx_f^3 + 4dx_f^2 dx_b + 4dx_f dx_b^2 + dx_b^3)} \,, \tag{9}$$

$$b_i^x = 1 \,, \tag{10}$$

$$c_i^x = \frac{dx_b\,(dx_f^2 + dx_b dx_f - dx_b^2)}{(dx_f^3 + 4dx_f^2 dx_b + 4dx_f dx_b^2 + dx_b^3)} \,, \tag{11}$$

$$ar_i^x = \frac{12dx_f}{(dx_f^3 + 4dx_f^2 dx_b + 4dx_f dx_b^2 + dx_b^3)} \,, \tag{12}$$

$$br_i^x = \frac{-12}{(dx_f^2 + 3dx_f dx_b + dx_b^2)} \,, \tag{13}$$

$$cr_i^x = \frac{12dx_b}{(dx_f^3 + 4dx_f^2 dx_b + 4dx_f dx_b^2 + dx_b^3)} \,. \tag{14}$$

The coefficients for Eq. (3) are the same except that x is replaced with y. Combining Eqs. (2) and (3) at three consecutive x- and y-locations centered at point (i, j) leads to

$$\begin{aligned} &a_j^y\,(L_{xx}u_{xx} - R_{xx}u)_{j-1} + b_j^y\,(L_{xx}u_{xx} - R_{xx}u)_j \;+ c_j^y\,(L_{xx}u_{xx} - R_{xx}u)_{j+1} \;+ \\ &a_i^x\,(L_{yy}u_{yy} - R_{yy}u)_{i-1} + b_i^x\,(L_{yy}u_{yy} - R_{yy}u)_i + c_i^x\,(L_{yy}u_{yy} - R_{yy}u)_{i+1} = 0. \end{aligned} \tag{15}$$

Applying the FD operators in Eq. (15) and using Eq. (1) leads to a nine-points, fourth-order compact scheme at the regular grid point (i, j) inside the computational domain as follows:

$$\begin{aligned} &A_{i,j}^{NW}\,u_{i-1,j+1} \;+\; A_{i,j}^{N}\,u_{i,j+1} \;+\; A_{i,j}^{NE}\,u_{i+1,j+1} + \\ &\quad A_{i,j}^{W}\,u_{i-1,j} \;\;+\; A_{i,j}^{P}\,u_{i,j} \;\;\;+\; A_{i,j}^{E}\,u_{i+1,j} \;\;+ \\ &A_{i,j}^{SW}\,u_{i-1,j-1} \;+\; A_{i,j}^{S}\,u_{i,j-1} \;+\; A_{i,j}^{SE}\,u_{i+1,j-1} = Q_{i,j}\,, \end{aligned} \tag{16}$$

where RHS is given by

$$\begin{aligned} Q_{i,j} = \;& Q_{i,j}^{NW}\,\rho_{i-1,j+1} \;+\; Q_{i,j}^{N}\,\rho_{i,j+1} \;+\; Q_{i,j}^{NE}\,\rho_{i+1,j+1} \\ &+ Q_{i,j}^{W}\,\rho_{i-1,j} \;\;\;+\; Q_{i,j}^{P}\,\rho_{i,j} \;\;+\; Q_{i,j}^{E}\,\rho_{i+1,j} \\ &+ Q_{i,j}^{SW}\,\rho_{i-1,j-1} \;+\; Q_{i,j}^{S}\,\rho_{i,j-1} \;+\; Q_{i,j}^{SE}\,\rho_{i+1,j-1}\,. \end{aligned} \tag{17}$$

The coefficients for LHS are given by

$$\begin{aligned} A_{i,j}^{SW} &= ar_i^x\,a_j^y + a_i^x\,ar_j^y - \beta\,a_i^x\,a_j^y\,, \\ A_{i,j}^{W} &= ar_i^x\,b_j^y + a_i^x\,br_j^y - \beta\,a_i^x\,b_j^y\,, \\ A_{i,j}^{NW} &= ar_i^x\,c_j^y + a_i^x\,cr_j^y - \beta\,a_i^x\,c_j^y\,, \\ A_{i,j}^{S} &= br_i^x\,a_j^y + b_i^x\,ar_j^y - \beta\,b_i^x\,a_j^y\,, \\ A_{i,j}^{P} &= br_i^x\,b_j^y + b_i^x\,br_j^y - \beta\,b_i^x\,b_j^y\,, \\ A_{i,j}^{N} &= br_i^x\,c_j^y + b_i^x\,cr_j^y - \beta\,b_i^x\,c_j^y\,, \\ A_{i,j}^{SE} &= cr_i^x\,a_j^y + c_i^x\,ar_j^y - \beta\,c_i^x\,a_j^y\,, \\ A_{i,j}^{E} &= cr_i^x\,b_j^y + c_i^x\,br_j^y - \beta\,c_i^x\,b_j^y\,, \\ A_{i,j}^{NE} &= cr_i^x\,c_j^y + c_i^x\,cr_j^y - \beta\,c_i^x\,c_j^y\,. \end{aligned} \tag{18}$$

For the RHS, the coefficients are

$$\begin{aligned} Q_{i,j}^{SW} &= a_i^x\,a_j^y\,, & Q_{i,j}^{S} &= b_i^x\,a_j^y\,, & Q_{i,j}^{SE} &= c_i^x\,a_j^y\,, \\ Q_{i,j}^{W} &= a_i^x\,b_j^y\,, & Q_{i,j}^{P} &= b_i^x\,b_j^y\,, & Q_{i,j}^{E} &= c_i^x\,b_j^y\,, \\ Q_{i,j}^{NW} &= a_i^x\,c_j^y\,, & Q_{i,j}^{N} &= b_i^x\,c_j^y\,, & Q_{i,j}^{NE} &= c_i^x\,c_j^y\,. \end{aligned} \tag{19}$$

3. Treatment of irregular points

In this section, the method to determine the coefficients of the compact scheme stencil at an irregular grid point is presented. The key aspect of the new method is to modify and adjust the compact finite-difference scheme, Eqs. (2) and (3), when they intersect an immersed boundary to obtain a solution that is sharp across the interface while keeping the fourth-order global accuracy.

The general process to construct a modified FD stencil for an irregular grid point centered at (i, j) is as follows: In the y-direction, in the range $y_{j-1} \le y \le y_{j+1}$, check if the grid lines passing through $x = x_{i-1}$, $x = x_i$ and $x = x_{i+1}$ have any intersection with the immersed boundary. If yes, modified FD operators have to be used along that line, otherwise, the standard FD operators in the y-direction given by Eqs. (5) and (7) are valid and can therefore be used. Similarly in the x-direction, in the range $x_{i-1} \le x \le x_{i+1}$, modified FD operators need to be employed along any mesh line $y = y_{j-1}$, $y = y_j$ and $y = y_{j+1}$ if there is an intersection with the immersed boundary. Otherwise, the standard FD operators in the x-direction, Eqs. (4) and (6), are valid. The modified compact FD stencil will be built based on the combination of the standard and modified FD operators. In the following sections, the treatment of irregular grid points for different situations is presented.

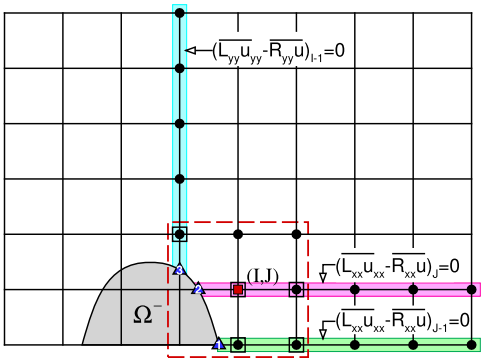

Fig. 3. Intersection of compact FD nine-point stencil at irregular grid point centered at (i, j) with an immersed boundary in the single intersection scenario. Colored rectangles correspond to the mesh lines where the modified FD operators are applied. All grid points inside the rectangles, as well as the points where the immersed boundary intersects grid lines, are used in the RHS of modified FD operators. Grid points enclosed by squares are the points used in the LHS of the modified FD operators. Immersed boundary intersection (IBI) points are numbered from 1 to 3. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

3.1. Single intersection with immersed boundary

In this scenario, the immersed boundary has only one intersection with each grid line in the range of $x_{i-1} \le x \le x_{i+1}$ and $y_{j-1} \le y \le y_{j+1}$. For the illustration, we consider the nine-point 2-D stencil centered at irregular grid point (i, j) as shown in Fig. 3. Lines $y = y_{j-1}$, $y = y_j$ and $x = x_{i-1}$ do intersect the immersed boundary and the FD operators along these lines need to be adjusted to take into account the immersed boundary while maintaining the formal fourth-order accuracy. Eq. (15) is rewritten as

$$\begin{aligned} &a_j^y\,(\overline{L_{xx}u_{xx}} - \overline{R_{xx}u})_{j-1} + b_j^y\,(\overline{L_{xx}u_{xx}} - \overline{R_{xx}u})_j \;+ c_j^y\,(L_{xx}u_{xx} - R_{xx}u)_{j+1} \;+ \\ &a_i^x\,(\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i-1} + b_i^x\,(L_{yy}u_{yy} - R_{yy}u)_i + c_i^x\,(L_{yy}u_{yy} - R_{yy}u)_{i+1} = 0\,. \end{aligned} \tag{20}$$

In the above equation, the FD operators with the overbar are modified, while the other operators are the same as those given in Eqs. (4)–(7). A key point is that the coefficients used in the modified LHS FD operators ($\overline{L_{xx}}$, $\overline{L_{yy}}$) in Eq. (20) must be kept the same as those used in Eqs. (4) and (5). However, the coefficients corresponding to the grid points inside the immersed boundary have to be dropped. Therefore, to maintain the formal fourth order accuracy, additional grid points are needed to determine the coefficients for the modified RHS FD operators ($\overline{R_{xx}}$, $\overline{R_{yy}}$) in Eq. (20). Along the line $y = y_{j-1}$, $(\overline{L_{xx}u_{xx}})_{j-1} = (\overline{R_{xx}u})_{j-1}$ where

$$(\overline{L_{xx}u_{xx}})_{j-1} = b_i^x\,u_{xx_{i,j-1}} + c_i^x\,u_{xx_{i+1,j-1}}\,, \tag{21}$$

$$(\overline{R_{xx}u})_{j-1} = \overline{br}_{i,1}^x\,u_{i,j-1} + \overline{cr}_{i,1}^x\,u_{i+1,j-1} + \phi_{1,1}\,u_{i+2,j-1} + \phi_{2,1}\,u_{i+3,j-1} + \phi_{3,1}\,u_{i+4,j-1} + \phi_{b1}\,u_{IBI_1}\,. \tag{22}$$

In Eq. (22), u_{IBI_1} is the known function value at the immersed boundary intersection location 1, IBI_1. It can be observed that the term $a_i^x\,u_{xx_{i-1,j-1}}$ is not included in Eq. (21) since it lies in the region Ω^-. For the right-hand side operator, Eq. (22), we use three additional points (see Fig. 3) to keep the fourth-order accuracy. Taylor series expansions about $u_{i,j-1}$ are used to find the coefficients by solving the following system of equations:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & h_0 & h_1 & h_2 & h_3 & h_b \\ 0 & h_0^2 & h_1^2 & h_2^2 & h_3^2 & h_b^2 \\ 0 & h_0^3 & h_1^3 & h_2^3 & h_3^3 & h_b^3 \\ 0 & h_0^4 & h_1^4 & h_2^4 & h_3^4 & h_b^4 \\ 0 & h_0^5 & h_1^5 & h_2^5 & h_3^5 & h_b^5 \end{bmatrix} \begin{bmatrix} \overline{br}_{i,1}^x \\ \overline{cr}_{i,1}^x \\ \phi_{1,1} \\ \phi_{2,1} \\ \phi_{3,1} \\ \phi_{b1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2(b_i^x + c_i^x) \\ 6\,h_0\,c_i^x \\ 12\,h_0^2\,c_i^x \\ 20\,h_0^3\,c_i^x \end{bmatrix}, \tag{23}$$

where $h_0 = x_{i+1} - x_i$, $h_1 = x_{i+2} - x_i$, $h_2 = x_{i+3} - x_i$, $h_3 = x_{i+4} - x_i$, and $h_b = x_{IBI_1} - x_i$. Note that the coefficients on the RHS in Eq. (23) are given by Eqs. (10) and (11). For the stencil passing through the $y = y_j$, $(\overline{L_{xx}u_{xx}})_j = (\overline{R_{xx}u})_j$ where the modified FD operators are given by

$$(\overline{L_{xx}u_{xx}})_j = b_i^x\,u_{xx_{i,j}} + c_i^x\,u_{xx_{i+1,j}}\,, \tag{24}$$

$$(\overline{R_{xx}u})_j = \overline{br}_{i,2}^x\,u_{i,j} + \overline{cr}_{i,2}^x\,u_{i+1,j} + \phi_{1,2}\,u_{i+2,j} + \phi_{2,2}\,u_{i+3,j} + \phi_{3,2}\,u_{i+4,j} + \phi_{b2}\,u_{IBI_2}\,, \tag{25}$$

where u_{IBI_2} is the known boundary value at IBI location 2, IBI_2. Matching the Taylor series coefficients about $u_{i,j}$, the coefficients in Eq. (25) can be found from

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & h_0 & h_1 & h_2 & h_3 & h_b \\ 0 & h_0^2 & h_1^2 & h_2^2 & h_3^2 & h_b^2 \\ 0 & h_0^3 & h_1^3 & h_2^3 & h_3^3 & h_b^3 \\ 0 & h_0^4 & h_1^4 & h_2^4 & h_3^4 & h_b^4 \\ 0 & h_0^5 & h_1^5 & h_2^5 & h_3^5 & h_b^5 \end{bmatrix} \begin{bmatrix} \overline{br_{i,2}^x} \\ \overline{cr_{i,2}^x} \\ \phi_{1,2} \\ \phi_{2,2} \\ \phi_{3,2} \\ \phi_{b2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2(b_i^x + c_i^x) \\ 6\, h_0\, c_i^x \\ 12\, h_0^2\, c_i^x \\ 20\, h_0^3\, c_i^x \end{bmatrix}, \tag{26}$$

where $h_0 = x_{i+1} - x_i$, $h_1 = x_{i+2} - x_i$, $h_2 = x_{i+3} - x_i$, $h_3 = x_{i+4} - x_i$, and $h_b = x_{IBI_2} - x_i$. Finally, for the stencil along the $x = x_{i-1}$, the FD scheme takes the form $(\overline{L_{yy}u_{yy}})_{i-1} = (\overline{R_{yy}u})_{i-1}$ where

$$(\overline{L_{yy}u_{yy}})_{i-1} = c_j^y\, u_{yy_{i-1,j+1}}\,, \tag{27}$$

$$(\overline{R_{yy}u})_{i-1} = \overline{cr_j^y}\, u_{i-1,j+1} + \psi_1\, u_{i-1,j+2} + \psi_2\, u_{i-1,j+3} + \psi_3\, u_{i-1,j+4} + \psi_4\, u_{i-1,j+5} + \psi_{b3}\, u_{IBI_3}\,, \tag{28}$$

u_{IBI_3} denoting the boundary value at IBI location 3, IBI_3. One should note that the Taylor expansion about $u_{i-1,j+1}$ is used to find the coefficients in Eq. (28) as follows

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & h_0 & h_1 & h_2 & h_3 & h_b \\ 0 & h_0^2 & h_1^2 & h_2^2 & h_3^2 & h_b^2 \\ 0 & h_0^3 & h_1^3 & h_2^3 & h_3^3 & h_b^3 \\ 0 & h_0^4 & h_1^4 & h_2^4 & h_3^4 & h_b^4 \\ 0 & h_0^5 & h_1^5 & h_2^5 & h_3^5 & h_b^5 \end{bmatrix} \begin{bmatrix} \overline{cr_j^y} \\ \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \psi_{b3} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2\, c_j^y \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{29}$$

where $h_0 = y_{j+2} - y_{j+1}$, $h_1 = y_{j+3} - y_{j+1}$, $h_2 = y_{j+4} - y_{j+1}$, $h_3 = y_{j+5} - y_{j+1}$, and $h_b = y_{IBI_3} - y_{j+1}$. Applying the above equations to Eq. (20), we get the modified compact scheme stencil at the irregular grid point (i, j)

$$\overline{A_{i,j}^S}\, u_{i,j-1} + \overline{A_{i,j}^{SE}}\, u_{i+1,j-1} + \overline{A_{i,j}^P}\, u_{i,j} + \overline{A_{i,j}^E}\, u_{i+1,j} + \overline{A_{i,j}^{NW}}\, u_{i-1,j+1} + A_{i,j}^N\, u_{i,j+1} + A_{i,j}^{NE}\, u_{i+1,j+1} + \mathcal{C}_{i,j} = \overline{Q_{i,j}} + \mathcal{B}_{i,j}\,, \tag{30}$$

with the modified coefficients

$$\begin{aligned} \overline{A_{i,j}^S} &= \overline{br_{i,1}^x}\, a_j^y + b_i^x\, ar_j^y - \beta\, b_i^x\, a_j^y\,, \\ \overline{A_{i,j}^{SE}} &= \overline{cr_{i,1}^x}\, a_j^y + c_i^x\, ar_j^y - \beta\, c_i^x\, a_j^y\,, \\ \overline{A_{i,j}^P} &= \overline{br_{i,2}^x}\, b_j^y + b_i^x\, br_j^y - \beta\, b_i^x\, b_j^y\,, \\ \overline{A_{i,j}^E} &= \overline{cr_{i,2}^x}\, b_j^y + c_i^x\, br_j^y - \beta\, c_i^x\, b_j^y\,, \\ \overline{A_{i,j}^{NW}} &= ar_i^x\, c_j^y + a_i^x\, \overline{cr_j^y} - \beta\, a_i^x\, c_j^y\,. \end{aligned} \tag{31}$$

The modified $\overline{Q_{i,j}}$ in Eq. (30) is the same as the one in Eq. (17) except that all the coefficients corresponding to the solid grid points are set to zero. In Eq. (30), $\mathcal{C}_{i,j}$ contains the additional points used to keep the fourth-order formal accuracy

$$\begin{aligned} \mathcal{C}_{i,j} = {} & a_j^y \left(\phi_{1,1}\, u_{i+2,j-1} + \phi_{2,1}\, u_{i+3,j-1} + \phi_{3,1}\, u_{i+4,j-1}\right) \\ & + b_j^y \left(\phi_{1,2}\, u_{i+2,j} + \phi_{2,2}\, u_{i+3,j} + \phi_{3,2}\, u_{i+4,j}\right) \\ & + a_i^x \left(\psi_1\, u_{i-1,j+2} + \psi_2\, u_{i-1,j+3} + \psi_3\, u_{i-1,j+4} + \psi_4\, u_{i-1,j+5}\right), \end{aligned} \tag{32}$$

and $\mathcal{B}_{i,j}$ includes the known function values at the immersed boundary intersection points,

$$\mathcal{B}_{i,j} = -(a_j^y\, \phi_{b1}\, u_{IBI_1} + b_j^y\, \phi_{b2}\, u_{IBI_2} + a_i^x\, \psi_{b3}\, u_{IBI_3})\,. \tag{33}$$

The last point to note is that the coefficients without the overbar in Eq. (30) are the same as the coefficients obtained for the regular grid points. It should be noted that the coefficients for the solid grid points are not used in Eq. (30) since the function value is zero at those grid points.

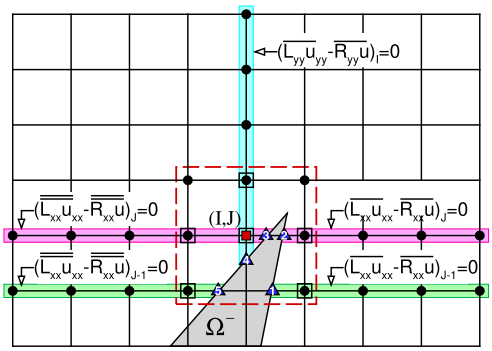

Fig. 4. Schematic of double intersections with immersed boundary for irregular grid point centered at (i, j). For the descriptions regarding the colored rectangles and marked points, see caption of Fig. 3. Points of intersection with the immersed boundary are numbered from 1 to 5.

3.2. Multiple intersections with immersed boundary

In many practical applications, however, the immersed boundary Γ can cross the grid lines more than once as shown in Fig. 4, for example. In this case, the finite difference operator centered at the irregular grid point (i, j) takes the following form:

$$a_j^y\,(\widehat{L_{xx}u_{xx}} - \widehat{R_{xx}u})_{j-1} + b_j^y\,(\widehat{L_{xx}u_{xx}} - \widehat{R_{xx}u})_j \;+ c_j^y\,(L_{xx}u_{xx} - R_{xx}u)_{j+1} \;+$$
$$a_i^x\,(L_{yy}u_{yy} - R_{yy}u)_{i-1} + b_i^x\,(\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_i + c_i^x\,(L_{yy}u_{yy} - R_{yy}u)_{i+1} = 0\,. \tag{34}$$

In the y-direction, the grid line passing through $x = x_i$ crosses the immersed boundary at the IBI_4. Along this line, $(\overline{L_{yy}u_{yy}})_i = (\overline{R_{yy}u})_i$ where

$$(\overline{L_{yy}u_{yy}})_i = b_j^y\,u_{yy_{i,j}} + c_j^y\,u_{yy_{i,j+1}}\,, \tag{35}$$
$$(\overline{R_{yy}u})_i = \overline{br_j^y}\,u_{i,j} + \overline{cr_j^y}\,u_{i,j+1} + \psi_1\,u_{i,j+2} + \psi_2\,u_{i,j+3} + \psi_3\,u_{i,j+4} + \psi_{b4}\,u_{IBI_4}\,. \tag{36}$$

The coefficients in Eq. (36) can be found from matching the Taylor series coefficients about $u_{i,j}$ in the y-direction. In the x-direction, however, the lines $y = y_{j-1}$ and $y = y_j$ intersect the immersed boundary at more than one location, namely IBI_1 & IBI_5 along $y = y_{j-1}$ and IBI_2 & IBI_3 along $y = y_j$. Therefore, special care is needed here to modify FD operators along these lines. For the stencil passing through $y = y_{j-1}$, $(\widehat{L_{xx}u_{xx}})_{j-1} = (\widehat{R_{xx}u})_{j-1}$ where $(\widehat{L_{xx}u_{xx}})_{j-1} = (\overline{L_{xx}u_{xx}})_{j-1} + (\overline{\overline{L_{xx}u_{xx}}})_{j-1}$ and $(\widehat{R_{xx}u})_{j-1} = (\overline{R_{xx}u})_{j-1} + (\overline{\overline{R_{xx}u}})_{j-1}$. The modified FD operators are given by

$$(\overline{L_{xx}u_{xx}})_{j-1} = c_i^x\,u_{xx_{i+1,j-1}}\,, \tag{37}$$
$$(\overline{R_{xx}u})_{j-1} = \overline{cr_{i,1}^x}\,u_{i+1,j-1} + \phi_{1,1}\,u_{i+2,j-1} + \phi_{2,1}\,u_{i+3,j-1} + \phi_{3,1}\,u_{i+4,j-1} + \phi_{b1}\,u_{IBI_1}\,, \tag{38}$$

and

$$(\overline{\overline{L_{xx}u_{xx}}})_{j-1} = a_i^x\,u_{xx_{i-1,j-1}}\,, \tag{39}$$
$$(\overline{\overline{R_{xx}u}})_{j-1} = \overline{\overline{ar_{i,1}^x}}\,u_{i-1,j-1} + \phi_{4,1}\,u_{i-2,j-1} + \phi_{5,1}\,u_{i-3,j-1} + \phi_{6,1}\,u_{i-4,j-1} + \phi_{b5}\,u_{IBI_5}\,. \tag{40}$$

Here, the Taylor series expansions in the x-direction about $u_{i+1,j-1}$ and $u_{i-1,j-1}$ are used to find the coefficients in Eqs. (38) and (40), respectively. Finally, for the stencil along the $y = y_j$, $(\widehat{L_{xx}u_{xx}})_j = (\widehat{R_{xx}u})_j$ where $(\widehat{L_{xx}u_{xx}})_j = (\overline{L_{xx}u_{xx}})_j + (\overline{\overline{L_{xx}u_{xx}}})_j$ and $(\widehat{R_{xx}u})_j = (\overline{R_{xx}u})_j + (\overline{\overline{R_{xx}u}})_j$ with the modified FD operators

$$(\overline{L_{xx}u_{xx}})_j = c_i^x\,u_{xx_{i+1,j}}\,, \tag{41}$$
$$(\overline{R_{xx}u})_j = \overline{cr_{i,2}^x}\,u_{i+1,j} + \phi_{1,2}\,u_{i+2,j} + \phi_{2,2}\,u_{i+3,j} + \phi_{3,2}\,u_{i+4,j} + \phi_{b2}\,u_{IBI_2}\,, \tag{42}$$

and

$$(\overline{\overline{L_{xx}u_{xx}}})_j = a_i^x\,u_{xx_{i-1,j}} + b_i^x\,u_{xx_{i,j}}\,, \tag{43}$$
$$(\overline{\overline{R_{xx}u}})_j = \overline{\overline{ar_{i,2}^x}}\,u_{i-1,j} + \overline{\overline{br_i^x}}\,u_{i,j} + \phi_{4,2}\,u_{i-2,j} + \phi_{5,2}\,u_{i-3,j} + \phi_{6,2}\,u_{i-4,j} + \phi_{b3}\,u_{IBI_3}\,. \tag{44}$$

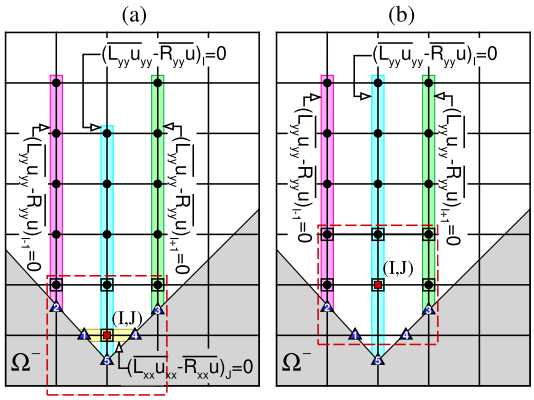

Fig. 5. Treatment of irregular grid points for a convex corner. For the descriptions regarding the colored rectangles and marked points, see caption of Fig. 3. Immersed boundary intersection points are numbered from 1 to 5. (*a*) Case *A*, (*b*) case *B*.

Using the original FD operators (Eqs. (9)–(14)) along the lines $x = x_{i-1}$, $x = x_{i+1}$ and $y = y_{j+1}$ and applying the modified FD operators, Eqs. (35)–(44), to Eq. (34), the modified compact scheme stencil at the irregular grid point (i, j) can be obtained

$$\begin{aligned}&\overline{A^{SW}_{i,j}}\, u_{i-1,j-1} + \overline{A^{SE}_{i,j}}\, u_{i+1,j-1} + \overline{A^{W}_{i,j}}\, u_{i-1,j} + \overline{A^{P}_{i,j}}\, u_{i,j} + \overline{A^{E}_{i,j}}\, u_{i+1,j} + A^{NW}_{i,j}\, u_{i-1,j+1} + \overline{A^{N}_{i,j}}\, u_{i,j+1} + A^{NE}_{i,j}\, u_{i+1,j+1}\\ &\quad + \mathcal{C}_{i,j} = \overline{Q_{i,j}} + \mathcal{B}_{i,j}\,. \end{aligned} \tag{45}$$

The modified coefficients are

$$\begin{aligned}
\overline{A^{SW}_{i,j}} &= \overline{\overline{ar^{x}_{i,1}}}\, a^{y}_{j} + a^{x}_{i}\, ar^{y}_{j} - \beta\, a^{x}_{i}\, a^{y}_{j}\,,\\
\overline{A^{SE}_{i,j}} &= \overline{cr^{x}_{i,1}}\, a^{y}_{j} + c^{x}_{i}\, ar^{y}_{j} - \beta\, c^{x}_{i}\, a^{y}_{j}\,,\\
\overline{A^{W}_{i,j}} &= \overline{\overline{ar^{x}_{i,2}}}\, b^{y}_{j} + a^{x}_{i}\, br^{y}_{j} - \beta\, a^{x}_{i}\, b^{y}_{j}\,,\\
\overline{A^{P}_{i,j}} &= \overline{\overline{br^{x}_{i}}}\, b^{y}_{j} + b^{x}_{i}\, \overline{br^{y}_{j}} - \beta\, b^{x}_{i}\, b^{y}_{j}\,,\\
\overline{A^{E}_{i,j}} &= \overline{cr^{x}_{i,2}}\, b^{y}_{j} + c^{x}_{i}\, br^{y}_{j} - \beta\, c^{x}_{i}\, b^{y}_{j}\,,\\
\overline{A^{N}_{i,j}} &= br^{x}_{i}\, c^{y}_{j} + b^{x}_{i}\, \overline{cr^{y}_{j}} - \beta\, b^{x}_{i}\, c^{y}_{j}\,.
\end{aligned} \tag{46}$$

The correction term, $\mathcal{C}_{i,j}$, is given by

$$\begin{aligned}\mathcal{C}_{i,j} = {}& a^{y}_{j}\left(\phi_{1,1}\, u_{i+2,j-1} + \phi_{2,1}\, u_{i+3,j-1} + \phi_{3,1}\, u_{i+4,j-1} + \phi_{4,1}\, u_{i-2,j-1} + \phi_{5,1}\, u_{i-3,j-1} + \phi_{6,1}\, u_{i-4,j-1}\right)\\ &+ b^{y}_{j}\left(\phi_{1,2}\, u_{i+2,j} + \phi_{2,2}\, u_{i+3,j} + \phi_{3,2}\, u_{i+4,j} + \phi_{4,2}\, u_{i-2,j} + \phi_{5,2}\, u_{i-3,j} + \phi_{6,2}\, u_{i-4,j}\right)\\ &+ b^{x}_{i}\left(\psi_{1}\, u_{i,j+2} + \psi_{2}\, u_{i,j+3} + \psi_{3}\, u_{i,j+4}\right).\end{aligned} \tag{47}$$

The known function values at the points of intersection with the immersed boundary are included in $\mathcal{B}_{i,j}$,

$$\mathcal{B}_{i,j} = -\left[a^{y}_{j}\left(\phi_{b1}\, u_{IBI_1} + \phi_{b5}\, u_{IBI_5}\right) + b^{y}_{j}\left(\phi_{b2}\, u_{IBI_2} + \phi_{b3}\, u_{IBI_3}\right) + b^{x}_{i}\, \psi_{b4}\, u_{IBI_4}\right]. \tag{48}$$

It is worth noting that the modified RHS, $\overline{Q_{i,j}}$, is the same as the one in Eq. (17) except that all the coefficients corresponding to the solid grid points are set to zero.

3.3. Treatment of irregular grid points in a convex corner

In dealing with a complex geometry, there are cases where an irregular grid point is located in a convex corner in which building a fourth-order accurate FD stencil is not as straightforward as was the case for the preceding sections. For example, in the situation illustrated in Fig. 5(a), the irregular point at (i, j) is surrounded by two boundary intersection points, IBI_1 & IBI_4, along the grid line $y = y_j$ which leads to a second-order operator. Similarly, in Fig. 5(b), the FD operator will be second-order accurate along $y = y_{j-1}$. Therefore, special treatment is required for irregular grid points associated with a convex corner in order to construct a fourth-order accurate stencil. Two cases are considered in this section as shown in Fig. 5.

3.3.1. Case A (see Fig. 5a)

For this case, the FD stencil centered at (i, j) takes the form

$$+ b_j^y\,(\overline{L_{xx}u_{xx}} - \overline{R_{xx}u})_j \;+ c_j^y\,(L_{xx}u_{xx} - R_{xx}u)_{j+1} \;+$$
$$a_i^x\,(\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i-1} + b_i^x\,(\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_i + c_i^x\,(\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i+1} = 0\,. \tag{49}$$

Along the line $x = x_k$, $(\overline{L_{yy}u_{yy}})_k = (\overline{R_{yy}u})_k$ with k being $k = i-1$, $k = i$ and $k = i+1$. For $k = i$, the modified FD operators are

$$(\overline{L_{yy}u_{yy}})_i = b_j^y\,u_{yy_{i,j}} + c_j^y\,u_{yy_{i,j+1}}\,, \tag{50}$$
$$(\overline{R_{yy}u})_i = \overline{br_j^y}\,u_{i,j} + \overline{cr_{j,1}^y}\,u_{i,j+1} + \psi_{1,1}\,u_{i,j+2} + \psi_{2,1}\,u_{i,j+3} + \psi_{3,1}\,u_{i,j+4} + \psi_{b5}\,u_{IBI_5}\,. \tag{51}$$

In Eq. (51) the coefficients are obtained by matching the Taylor series expansion about $u_{i,j}$ in the y-direction. The modified operators for $k = i-1$ and $k = i+1$ are given by

$$(\overline{L_{yy}u_{yy}})_i = c_j^y\,u_{yy_{k,j+1}}\,, \tag{52}$$
$$(\overline{R_{yy}u})_i = \overline{cr_{j,\alpha}^y}\,u_{k,j+1} + \psi_{1,\alpha}\,u_{i,j+2} + \psi_{2,\alpha}\,u_{k,j+3} + \psi_{3,\alpha}\,u_{i,j+4} + \psi_{4,\alpha}\,u_{i,j+5} + \psi_{b\alpha}\,u_{IBI_\alpha}\,, \tag{53}$$

with $\alpha = 2$ and $\alpha = 3$ for $k = i-1$ and $k = i+1$, respectively. Matching the Taylor series coefficients about $u_{k,j+1}$ is employed to find the coefficients in Eq. (53).

For the stencil along $y = y_j$, the FD scheme takes the form $(\overline{L_{xx}u_{xx}})_j = (\overline{R_{xx}u})_j$ where

$$(\overline{L_{xx}u_{xx}})_j = b_i^x\,u_{xx_{i,j}}\,, \tag{54}$$
$$(\overline{R_{xx}u})_j = \overline{br_i^x}\,u_{i,j} + \phi_{b1}\,u_{IBI_1} + \phi_{b4}\,u_{IBI_4}\,. \tag{55}$$

The modified FD operator in Eq. (55) is second-order accurate which leads to the second-order accurate FD stencil for the irregular point at (i, j). Of particular interest in this case is the observation that the additional grid points that are used along $x = x_{i-1}$ and $x = x_{i+1}$ for the modified FD operators in the y-direction, Eq. (53), can be used to augment the accuracy to fourth-order for the FD operator along $y = y_j$. To achieve this, the modified FD operators along $y = y_j$ are rewritten as

$$(\overline{L_{xx}u_{xx}})_j = b_i^x\,u_{xx_{i,j}}\,, \tag{56}$$
$$(\overline{R_{xx}u})_j = \overline{br_i^x}\,u_{i,j} + \phi_1\,\hat{u}_{i-1,j} + \phi_2\,\hat{u}_{i+1,j} + \phi_{b1}\,u_{IBI_1} + \phi_{b4}\,u_{IBI_4}\,, \tag{57}$$

where $\hat{u}_{i-1,j}$ and $\hat{u}_{i+1,j}$ are the fictitious points that lie inside the region Ω^-. However, they can be evaluated using directional extrapolation in the y-direction as follows:

$$\hat{u}_{k,j} = \theta_{1,\alpha}\,u_{IBI_m} + \theta_{2,\alpha}\,u_{k,j+1} + \theta_{3,\alpha}\,u_{k,j+2} + \theta_{4,\alpha}\,u_{k,j+3} + \theta_{5,\alpha}\,u_{k,j+4}\,, \tag{58}$$

with $\alpha = 1$ & $m = 2$ for $k = i-1$ and $\alpha = 2$ & $m = 3$ for $k = i+1$. Matching the Taylor series coefficients about $\hat{u}_{k,j}$, the coefficients in Eq. (58) can be found from

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ h_b & h_1 & h_2 & h_3 & h_4 \\ h_b^2 & h_1^2 & h_2^2 & h_3^2 & h_4^2 \\ h_b^3 & h_1^3 & h_2^3 & h_3^3 & h_4^3 \\ h_b^4 & h_1^4 & h_2^4 & h_3^4 & h_4^4 \end{bmatrix} \begin{bmatrix} \theta_{1,\alpha} \\ \theta_{2,\alpha} \\ \theta_{3,\alpha} \\ \theta_{4,\alpha} \\ \theta_{5,\alpha} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{59}$$

where $h_b = y_{IBI_m} - y_j$, $h_1 = y_{j+1} - y_j$, $h_2 = y_{j+2} - y_j$, $h_3 = y_{j+3} - y_j$, and $h_4 = y_{j+4} - y_j$. Finally, the modified compact stencil at the irregular grid point (i, j) becomes

$$\overline{A_{i,j}^P}\,u_{i,j} + \overline{A_{i,j}^{NW}}\,u_{i-1,j+1} + \overline{A_{i,j}^N}\,u_{i,j+1} + \overline{A_{i,j}^{NE}}\,u_{i+1,j+1} + \mathcal{C}_{i,j} = \overline{Q_{i,j}} + \mathcal{B}_{i,j}\,, \tag{60}$$

with the modified coefficients

$$\begin{aligned} \overline{A_{i,j}^P} &= \overline{br_i^x}\,b_j^y + b_i^x\,\overline{br_j^y} - \beta\,b_i^x\,b_j^y\,, \\ \overline{A_{i,j}^{NW}} &= ar_i^x\,c_j^y + a_i^x\,\overline{cr_{j,2}^y} - \beta\,a_i^x\,c_j^y + b_j^y\,\phi_1\,\theta_{2,1}\,, \\ \overline{A_{i,j}^N} &= br_i^x\,c_j^y + b_i^x\,\overline{cr_{j,1}^y} - \beta\,b_i^x\,c_j^y\,, \\ \overline{A_{i,j}^{NE}} &= cr_i^x\,c_j^y + c_i^x\,\overline{cr_{j,3}^y} - \beta\,c_i^x\,c_j^y + b_j^y\,\phi_2\,\theta_{2,2}\,. \end{aligned} \tag{61}$$

The correction term $\mathcal{C}_{i,j}$ and $\mathcal{B}_{i,j}$ in Eq. (60) are given by

$$\begin{aligned}\mathcal{C}_{i,j} = {} & a_i^x \left(\psi_{1,2}\, u_{i-1,j+2} + \psi_{2,2}\, u_{i-1,j+3} + \psi_{3,2}\, u_{i-1,j+4} + \psi_{4,2}\, u_{i-1,j+5}\right) \\ & + b_i^x \left(\psi_{1,1}\, u_{i,j+2} + \psi_{2,1}\, u_{i,j+3} + \psi_{3,1}\, u_{i,j+4}\right) \\ & + c_i^x \left(\psi_{1,3}\, u_{i+1,j+2} + \psi_{2,3}\, u_{i+1,j+3} + \psi_{3,3}\, u_{i+1,j+4} + \psi_{4,3}\, u_{i+1,j+5}\right) \\ & + b_j^y\, \phi_1 \left(\theta_{3,1}\, u_{i-1,j+2} + \theta_{4,1}\, u_{i-1,j+3} + \theta_{5,1}\, u_{i-1,j+4}\right) \\ & + b_j^y\, \phi_2 \left(\theta_{3,2}\, u_{i+1,j+2} + \theta_{4,2}\, u_{i+1,j+3} + \theta_{5,2}\, u_{i+1,j+4}\right), \end{aligned} \tag{62}$$

$$\mathcal{B}_{i,j} = -\left[a_i^x\, \psi_{b2}\, u_{IBI_2} + b_i^x\, \psi_{b5}\, u_{IBI_5} + c_i^x\, \psi_{b3}\, u_{IBI_3} + b_j^y \left(\phi_{b1}\, u_{IBI_1} + \phi_{b4}\, u_{IBI_4} + \phi_1\, \theta_{1,1}\, u_{IBI_2} + \phi_2\, \theta_{1,2}\, u_{IBI_3}\right)\right]. \tag{63}$$

This stencil is fourth-order accurate as will be formally demonstrated in Section 4.1.

3.3.2. Case B (see Fig. 5b)

For this case, the FD stencil centered at (i, j) takes the form

$$\begin{aligned} & a_j^y\, (\overline{L_{xx}u_{xx}} - \overline{R_{xx}u})_{j-1} + b_j^y\, (L_{xx}u_{xx} - R_{xx}u)_j \; + c_j^y\, (L_{xx}u_{xx} - R_{xx}u)_{j+1} \; + \\ & a_i^x\, (\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i-1} + b_i^x\, (L_{yy}u_{yy} - R_{yy}u)_i + c_i^x\, (\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i+1} = 0\,. \end{aligned} \tag{64}$$

For stencil along the $y = y_{j-1}$, $(\overline{L_{yy}u_{yy}})_{j-1} = (\overline{R_{yy}u})_{j-1}$ where

$$(\overline{L_{xx}u_{xx}})_{j-1} = b_i^x\, u_{xx_{i,j-1}}\,, \tag{65}$$

$$(\overline{R_{xx}u})_{j-1} = \overline{br_i^x}\, u_{i,j-1} + \phi_{b1}\, u_{IBI_1} + \phi_{b4}\, u_{IBI_4}\,. \tag{66}$$

The finite difference formula in Eq. (66) leads to the second-order accurate FD stencil for the irregular point at (i, j). To remedy this, the compact FD stencil centered at (i, j) is recast as

$$\begin{aligned} & + b_j^y\, (L_{xx}u_{xx} - R_{xx}u)_j \; + c_j^y\, (L_{xx}u_{xx} - R_{xx}u)_{j+1} \; + \\ & a_i^x\, (\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i-1} + b_i^x\, (\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_i + c_i^x\, (\overline{L_{yy}u_{yy}} - \overline{R_{yy}u})_{i+1} = 0\,. \end{aligned} \tag{67}$$

Compared to Eq. (64), the modified FD operators at $y = y_{j-1}$ are removed and the FD operators acting along $x = x_i$ are treated as the modified ones. Special treatment is used for the FD operators acting at $x = x_i$, $(\overline{L_{yy}u_{yy}})_i = (\overline{R_{yy}u})_i$ as follows:

$$(\overline{L_{yy}u_{yy}})_i = b_j^y\, u_{yy_{i,j}} + c_j^y\, u_{yy_{i,j+1}}\,, \tag{68}$$

$$(\overline{R_{yy}u})_i = \overline{br_{j,1}^y}\, u_{i,j} + \overline{cr_{j,1}^y}\, u_{i,j+1} + \psi_{1,1}\, u_{i,j+2} + \psi_{2,1}\, u_{i,j+3} + \psi_{3,1}\, u_{i,j+4} + \psi_{b5}\, u_{IBI_5}\,. \tag{69}$$

The key point is that while the grid point at $(i, j-1)$ lies in the region Ω^+, $u_{i,j-1}$ is not used in Eq. (69). The coefficients in Eq. (69) are obtained from matching Taylor series expansions about $u_{i,j}$ in the y-direction. Along the line $x = x_k$, $(\overline{L_{yy}u_{yy}})_k = (\overline{R_{yy}u})_k$ with k being $k = i-1$ and $k = i+1$,

$$(\overline{L_{yy}u_{yy}})_k = b_j^y\, u_{yy_{k,j}} + c_j^y\, u_{yy_{k,j+1}}\,, \tag{70}$$

$$(\overline{R_{yy}u})_k = \overline{br_{j,\alpha}^y}\, u_{k,j} + \overline{cr_{j,\alpha}^y}\, u_{k,j+1} + \psi_{1,\alpha}\, u_{k,j+2} + \psi_{2,\alpha}\, u_{k,j+3} + \psi_{3,\alpha}\, u_{k,j+4} + \psi_{b\alpha}\, u_{IBI_\alpha}\,, \tag{71}$$

with $\alpha = 2$ for $k = i-1$ and $\alpha = 3$ for $k = i+1$. Matching the Taylor series coefficients about $u_{k,j}$ is employed to find the coefficients in Eq. (71). The modified compact stencil at the irregular grid point (i, j) reads then

$$\overline{A_{i,j}^W}\, u_{i-1,j} + \overline{A_{i,j}^P}\, u_{i,j} + \overline{A_{i,j}^E}\, u_{i+1,j} + \overline{A_{i,j}^{NW}}\, u_{i-1,j+1} + \overline{A_{i,j}^N}\, u_{i,j+1} + \overline{A_{i,j}^{NE}}\, u_{i+1,j+1} + \mathcal{C}_{i,j} = \overline{Q_{i,j}} + \mathcal{B}_{i,j}\,, \tag{72}$$

with the modified coefficients

$$\begin{aligned} \overline{A_{i,j}^W} &= ar_i^x\, b_j^y + a_i^x\, \overline{br_{j,2}^y} - \beta\, a_i^x\, b_j^y\,, \\ \overline{A_{i,j}^P} &= br_i^x\, b_j^y + b_i^x\, \overline{br_{j,1}^y} - \beta\, b_i^x\, b_j^y\,, \\ \overline{A_{i,j}^E} &= cr_i^x\, b_j^y + c_i^x\, \overline{br_{j,3}^y} - \beta\, c_i^x\, b_j^y\,, \\ \overline{A_{i,j}^{NW}} &= ar_i^x\, c_j^y + a_i^x\, \overline{cr_{j,2}^y} - \beta\, a_i^x\, c_j^y\,, \\ \overline{A_{i,j}^N} &= br_i^x\, c_j^y + b_i^x\, \overline{cr_{j,1}^y} - \beta\, b_i^x\, c_j^y\,, \\ \overline{A_{i,j}^{NE}} &= cr_i^x\, c_j^y + c_i^x\, \overline{cr_{j,3}^y} - \beta\, c_i^x\, c_j^y\,. \end{aligned} \tag{73}$$

Table 1
Order of discrete compact difference stencils for regular and irregular grid points.

Grid point type	Error
Regular grid point	$\varepsilon = \frac{1}{100}(k^6dx^4 + l^6dy^4) - \frac{1}{1200}k^2l^2(k^4dx^2 + l^4dy^2)dx^2dy^2 + O(dx^4dy^4)$
Single intersection	$\varepsilon = \frac{1}{2400}(13k^6dx^4 + 23l^6dy^4) - \frac{1}{1600}k^2l^2(k^4dx^2 + l^4dy^2)dx^2dy^2 + O(dx^4dy^4)$
Double intersection	$\varepsilon = \frac{1}{1200}(12k^6dx^4 + 7l^6dy^4) - \frac{1}{1200}k^2l^2(k^4dx^2 + l^4dy^2)dx^2dy^2 + O(dx^4dy^4)$
Corner grid point: Case A	$\varepsilon = \frac{1}{1200}(k^6dx^4 + 6l^6dy^4) - \frac{1}{2400}k^2l^2(k^4dx^2 + l^4dy^2)dx^2dy^2 + O(dx^4dy^4)$
Corner grid point: Case B	$\varepsilon = \frac{1}{1200}(11k^6dx^4 + 6l^6dy^4) - \frac{1}{2400}k^2l^2(k^4dx^2 + l^4dy^2)dx^2dy^2 + O(dx^4dy^4)$

The correction term $\mathcal{C}_{i,j}$ and $\mathcal{B}_{i,j}$ in Eq. (72) are given by

$$\begin{aligned}\mathcal{C}_{i,j} = {} & a_i^x\left(\psi_{1,2}\,u_{i-1,j+2} + \psi_{2,2}\,u_{i-1,j+3} + \psi_{3,2}\,u_{i-1,j+4}\right)\\ & + b_i^x\left(\psi_{1,1}\,u_{i,j+2} + \psi_{2,1}\,u_{i,j+3} + \psi_{3,1}\,u_{i,j+4}\right)\\ & + c_i^x\left(\psi_{1,3}\,u_{i+1,j+2} + \psi_{2,3}\,u_{i+1,j+3} + \psi_{3,3}\,u_{i+1,j+4}\right),\end{aligned} \tag{74}$$

$$\mathcal{B}_{i,j} = -\left(a_i^x\,\psi_{b2}\,u_{IBI_2} + b_i^x\,\psi_{b5}\,u_{IBI_5} + c_i^x\,\psi_{b3}\,u_{IBI_3}\right). \tag{75}$$

The fourth-order accuracy of the modified FD stencil in Eq. (72) will be formally demonstrated in Section 4.1.

4. Compact finite-difference operator: order and eigenvalues

In this section, the order of accuracy of the original and the modified compact finite-difference stencils is formally verified. Furthermore, the eigenvalue spectra of the resulting coefficient matrices for simple and irregular domains and a domain with an immersed boundary is investigated.

4.1. Order of compact finite-difference approximation for regular and irregular grid points

A formal derivation of fourth-order-accurate compact FD stencils for regular and irregular points was given in Section 2 and Section 3. Although all the FD operators are based on the desired fourth-order compact finite difference approximations, a formal confirmation for this fact needs to be provided. Using the fact that the eigenfunctions of the exact Laplace operator are known to be $\phi_{k,l} = cos(kx)\ cos(ly)$ with corresponding eigenvalues $\lambda = -(k^2 + l^2)$, one can estimate the order of discrete FD operators where eigenvalues change according to the order of the accuracy of finite-difference approximations [25,26]. In the derivation of the compact FD stencils, one operator acts on the field u, and one operator operates on the source term ρ, i.e. $A_h\ \boldsymbol{u} = Q_h\ \boldsymbol{\rho}$, where A_h and Q_h are discrete operators, see Eqs. (16)–(17) for example. Therefore, a generalized eigenvalue problem,

$$A_h\ \boldsymbol{u} = \lambda\ Q_h\ \boldsymbol{u}\,, \tag{76}$$

has to be solved. As pointed out by Sutmann and Steffen [25], both A_h and Q_h have the desired property that eigenfunctions are sampled continuous eigenvectors. As a result, only the eigenvalues need to be considered. For practical purposes one may insert the eigenfunctions of the exact operator and estimate the order of FD approximations in $h = max(dx, dy)$, i.e.,

$$\left[A_h\ \phi_{k,l} + (k^2 + l^2)\ Q_h\ \phi_{k,l}\right]\phi_{k,l}^{-1} = O(h^n)\,, \tag{77}$$

where n is the order of the difference approximation. Without loss of generality, it is assumed that the grid is uniform in each direction, however, the grid spacing in each direction could be different, i.e. $dx \neq dy$. Furthermore, the locations of the points of intersection with the immersed boundary are estimated according to Figs. 3–5. Applying the FD operators developed in Section 2 and Section 3, Eqs. (16), (30), (45), (60) and (72), provides the order of the compact FD stencils as presented in Table 1. For all the FD stencils developed for the regular and irregular grid points, the expected fourth-order accuracy is recovered.

4.2. Spectral analysis

The discretization of Eq. (1) for all grid points inside the computational domain, $2 \leq i \leq nx - 1$ and $2 \leq j \leq ny - 1$, leads to a set of $(nx - 2) \times (ny - 2)$ linear equations which can be written in vector form as

$$\mathcal{A}\,\boldsymbol{u} = \boldsymbol{b}\,, \tag{78}$$

where $\boldsymbol{b}$ is the source term, see Eq. (17). The coefficient matrix $\mathcal{A}$ has the following structure for a simple domain (without immersed boundary):

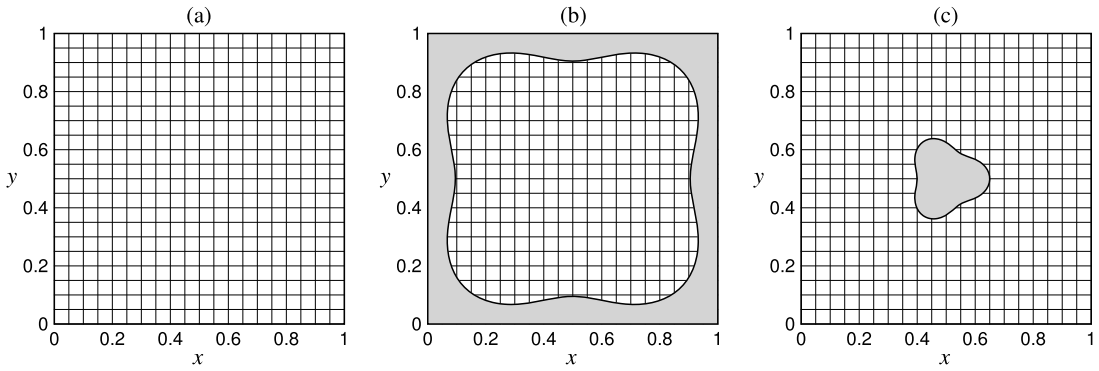

Fig. 6. Schematics of different cases considered for the evaluation of eigenvalue spectra. (*a*) Simple domain, (*b*) irregular domain boundary, and (*c*) domain with an immersed boundary. For all cases, a uniform grid with 41×41 points are used. For clarity, only every second point in x and y is shown.

$$\mathcal{A} = \begin{bmatrix} B_2 & U_2 & 0 & 0 & 0 & 0 & 0 \\ L_3 & B_3 & U_3 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & L_j & B_j & C_j & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & L_{ny-2} & B_{ny-2} & U_{ny-2} \\ 0 & 0 & 0 & 0 & 0 & L_{ny-1} & B_{ny-1} \end{bmatrix}, \tag{79}$$

with tridiagonal sub-matrices L_j, B_j, and U_j of size $(nx-2) \times (nx-2)$ where $j = 2, \cdots, ny-1$, sweeping in i, leading to a nine-diagonal matrix. In the presence of an immersed boundary, additional grid points are used to keep the fourth-order formal accuracy. Consequently, the discretization matrix will lose its nine-diagonal structure. It is worth noting that the matrix $\mathcal{A}$ is not symmetric in general. The only situation where the matrix $\mathcal{A}$ is symmetric is for a simple domain with uniform grid spacing.

The numerical stability involving a finite difference approximation for Eq. (1) requires that all eigenvalues (λ) of $\mathcal{A}$ satisfies $real(\lambda) < 0$ [19]. The eigenvalues of the discrete finite difference operator will depend upon the geometry and grid resolution of the problem. Therefore, it is necessary to evaluate the effect of the extended stencil due to the presence of an immersed boundary on the eigenvalue spectra of matrix $\mathcal{A}$. In the following, the matrix structures and the eigenvalue spectra of the fourth-order-accurate compact difference scheme for three different cases are investigated: (i) A simple domain, (ii) a domain with irregular domain boundary, and (iii) a domain with an immersed body as shown in Fig. 6. For case (ii), the irregular domain boundary Γ is defined by the parametric equations,

$$\begin{aligned} x_\Gamma(\theta) &= \left[0.154 sin^2(2\theta) + 0.031 cos^2(2\theta) + 0.374\right] cos(\theta)\,, \\ y_\Gamma(\theta) &= \left[0.154 sin^2(2\theta) + 0.031 cos^2(2\theta) + 0.374\right] sin(\theta)\,, \end{aligned} \tag{80}$$

for $0 \le \theta \le 2\pi$. Whereas for case (ii) the domain Ω^+ is the region inside the boundary Γ, the reverse is true for case (iii) where Γ is given by

$$\begin{aligned} x_\Gamma(\theta) &= \left[0.125 + 0.025 cos(3\theta)\right] cos(\theta)\,, \\ y_\Gamma(\theta) &= \left[0.125 + 0.025 cos(3\theta)\right] sin(\theta)\,. \end{aligned} \tag{81}$$

Fig. 7 shows the sparse patterns of the discretization matrices for the three cases as obtained for a 11×11 uniform grid (for illustration purposes). This demonstrates the effect of the extended stencil for irregular grid points on the nine-diagonal matrix. Fig. 8 presents the eigenvalue spectra of the resulting discretization matrix for the three cases when computed for a uniform 41×41 grid. For the simple domain, all eigenvalues are real and negative as the coefficient matrix is symmetric. As can be seen the irregular boundaries introduce complex eigenvalues. All the eigenvalues, however, have negative real components and therefore satisfy the stability condition. The eigenvalues shown in Fig. 8 are normalized with dx^2.

5. Solution strategy: multiscale multigrid method

The compact finite difference discretization of Eq. (1) at all grid points forms a large sparse algebraic system. Multigrid methods [27] are among the fastest and most efficient iterative methods for solving systems of linear equations arising from the discretization of partial differential equations. Multigrid methods typically converge at a rate independent of the

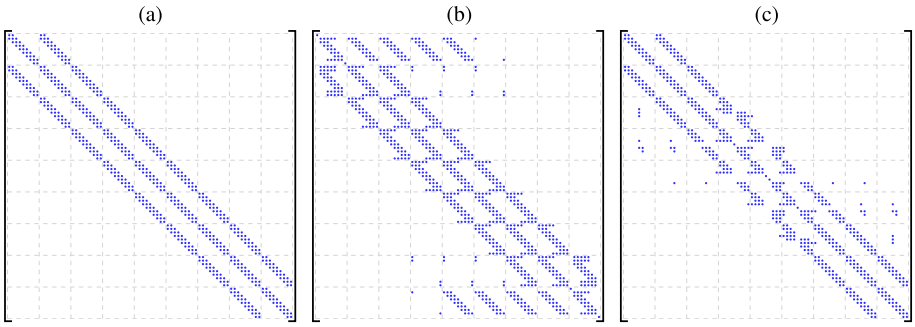

Fig. 7. Sparse patterns of discretization matrix obtained on a uniform 11×11 grid. (*a*) Simple domain, (*b*) irregular domain boundary, and (*c*) domain with an immersed boundary.

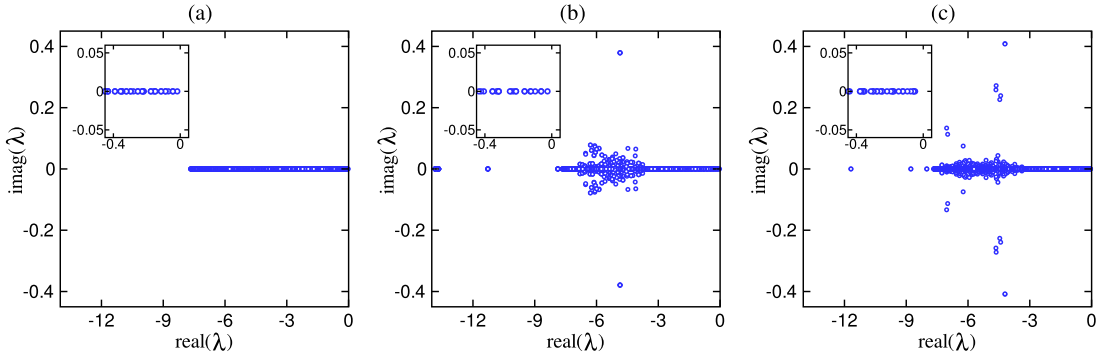

Fig. 8. Eigenvalue spectra of the discretization matrices obtained for a uniform grid with 41×41 points. (*a*) Simple domain, (*b*) irregular domain boundary, and (*c*) domain with an immersed boundary. Inserts are close-up views of the spectra near the origin.

grid size for a prescribed accuracy for elliptic problems. They will thus provide a solution for a specified accuracy in a fixed number of iterations, independent of how fine the mesh is. However, more iterations are needed to exploit the higher accuracy expected for the finer mesh.

One approach to improve the performance of the multigrid method is by providing a better initial guess to start the multigrid iteration. This can be achieved by a *nested iteration*, where low-cost computations on the coarser level is used to provide an initial guess on the finest grid. In this technique, the system is first solved (iteratively or directly) on the coarsest grid, then the results are interpolated to the next finer grid, where a few inexpensive multigrid V-cycles are applied and so on, until the finest grid is reached. The combination of the nested iteration and the multigrid is called full multigrid (FMG) method [27].

In this paper, a modified version of an FMG method is employed, the so-called multiscale multigrid method as proposed by Wang and Zhang [28]. They used the MSMG method on a simple rectangular domain with a uniform mesh to obtain higher-order accurate solution from two lower-order solutions by using a multiscale strategy in combination with a Richardson extrapolation. Here, our goal is to obtain a solution, where the accuracy is only limited by the discretization error, in a fixed number of iterations (independent of the grid size) by using the MSMG method for irregular boundaries on uniform and/or non-uniform meshes. Fig. 9 illustrates the structure of the MSMG method. It is similar to the FMG technique, but the computation does not start from the coarsest grid. In this example, six different grid levels and four different multigrid V-cycles are applied. The MSMG method starts with a standard multigrid V-cycle on $\Omega_{8\Delta}$ and then on $\Omega_{4\Delta}$ and so on until it reaches the finest level Ω_{Δ}.

In this technique, the solution from $\Omega_{8\Delta}$ is interpolated on $\Omega_{4\Delta}$ and used as an initial approximation; the solution on $\Omega_{4\Delta}$ is used as the initial guess for $\Omega_{2\Delta}$; and finally, the solution from $\Omega_{2\Delta}$ provides the initial approximation for Ω_{Δ}. Typically, just one or two multigrid V-cycles are necessary at each level of the hierarchy. Here Ω_{Δ} denotes the finest grid level and $\Omega_{2\Delta}$ corresponds to a grid one level coarser than grid Ω_{Δ} and so on. Grid $\Omega_{2\Delta}$ is formed by the standard coarsening strategy in which every second grid point of grid Ω_{Δ} is considered.

Each of the standard multigrid V-cycles in the MSMG method consists of smoothing the errors using a relaxation technique, restricting the residuals to the next coarser grid and prolongating the corrections back to the finer grid. Therefore, the performance and efficiency of multigrid depends on these three operations, relaxation, restriction and prolongation. The optimal efficiency of the MSMG method can be exploited only when they are suitably designed and constructed. This is particularly the case when non-uniform grids with immersed boundaries are considered. If these operators are not optimally designed, a deterioration of the convergence rate will occur. Note also that a one-dimensional, fourth-order finite difference

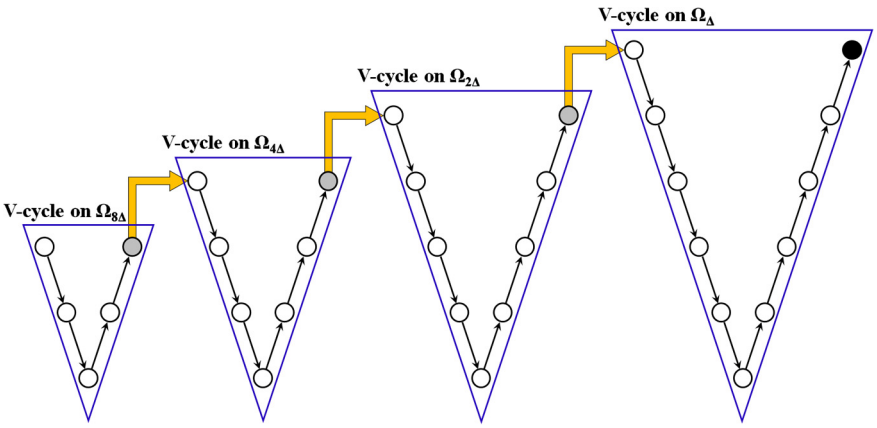

Fig. 9. Representation of multiscale multigrid method. Down arrows represent the restriction operator, up arrows correspond to the prolongation operator and right bent arrows interpolate the solution to the next multigrid V-cycle.

interpolation is used to interpolate the solution from one multigrid V-cycle to the next one (shown as right-bent arrows in Fig. 9).

5.1. Relaxation operator

The relaxation operator (the smoother) is very important in the multigrid method. The main task of the relaxation operator is to damp out the high frequency components of the errors on the current grid while leaving the low frequency components to be removed by the coarser grids. The discretization of the Poisson equation without immersed boundaries leads to a system of equations $\mathcal{A}\ \boldsymbol{u} = \boldsymbol{b}$ where the matrix $\mathcal{A}$ has a nine-diagonal structure (see Eq. (79)). This system is solved iteratively using the modified strongly implicit (MSI) procedure as proposed by Schneider and Zedan [29]. The MSI method has very good convergence properties, even for highly stretched grids and was shown to outperform the incomplete lower-upper decomposition (ILU) and Gauss–Seidel methods [29]. The basic idea in the MSI procedure is to replace the matrix $\mathcal{A}$ with approximated lower and upper diagonal matrices such that $\mathcal{A} = \mathcal{L}\mathcal{U} + \mathcal{M}$ where $||\mathcal{M}|| << ||\mathcal{A}||$. Both $\mathcal{L}$ and $\mathcal{U}$ matrices have five lower and upper diagonals, respectively. With this, a single-grid iterative method for solving $\mathcal{A}\ \boldsymbol{u} = \boldsymbol{b}$ with starting guess $\boldsymbol{u}^n$ is given by:

1. $\boldsymbol{r}^n = \boldsymbol{b} - \mathcal{A}\ \boldsymbol{u}^n$
2. $\mathcal{L}\ \boldsymbol{w}^{n+1} = \boldsymbol{r}^n$
3. $\mathcal{U}\ \boldsymbol{\delta}^{n+1} = \boldsymbol{w}^{n+1}$
4. $\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \boldsymbol{\delta}^{n+1}$

For brevity, a considerable amount of detail has to be left out of the present description regarding the MSI algorithm (for more detail see [29]). The extension to problems with immersed boundaries is considered next. The discretization of Eq. (1) for all regular and irregular grid points inside the computation domain with an immersed boundary leads to a set of $(nx-2)\times(ny-2)$ linear equations,

$$\mathcal{P}\ \boldsymbol{u} = \boldsymbol{q}\ , \tag{82}$$

where $\boldsymbol{q}$ includes the source term $\boldsymbol{b}$ (see Eq. (78)) as well as the known function values at the points of intersection with the immersed boundary. However, as discussed in the previous sections, the finite-difference schemes are modified when the nine-point stencil intersects with the immersed boundary. As a result, this introduces additional points to the nine-point compact discretization that are solution dependent, so that the coefficient matrix $\mathcal{P}$ is no longer nine-diagonal (see Fig. 7). It is important to note, however, the MSI procedure was designed and works only for nine-diagonal matrices. Therefore, to be able to use this efficient iterative method, the coefficient matrix $\mathcal{P}$ is decomposed into

$$\mathcal{P} = \overline{\mathcal{A}} + \mathcal{C}\ , \tag{83}$$

where the modified matrix $\overline{\mathcal{A}}$ has the same structure as $\mathcal{A}$ (nine-diagonal matrix). In fact, all the rows corresponding to the regular grid points are the same as the ones in matrix $\mathcal{A}$, however, some of the coefficients in the rows corresponding to the irregular grid points are modified. In Eq. (83), the matrix $\mathcal{C}$ represents irregular entries caused by the extended stencils at the irregular grid points. The key point is that the matrix $\mathcal{C}$ will be applied to the known solutions from the previous iteration. Therefore, the MSI algorithm was modified according to

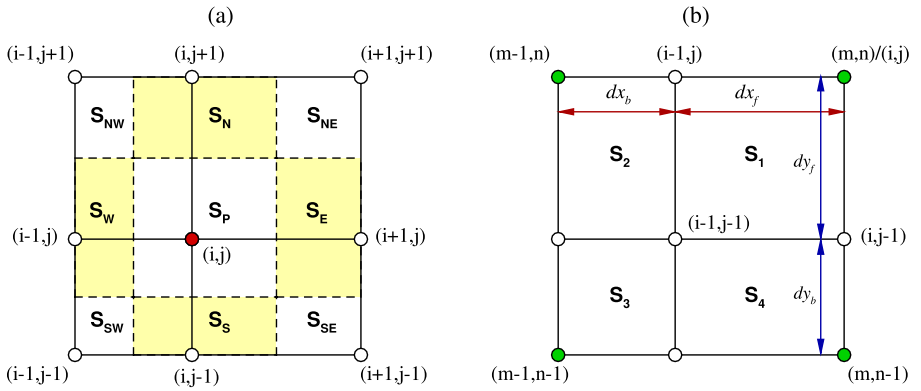

Fig. 10. Illustration of the restriction (a) and prolongation (b) operators and their corresponding areas on non-uniform grid.

1. $\boldsymbol{r}^n = \boldsymbol{q} - \overline{\mathcal{A}}\ \boldsymbol{u}^n - \langle \mathcal{C}\ \boldsymbol{u}^n \rangle$
2. $\overline{\mathcal{L}}\ \boldsymbol{w}^{n+1} = \boldsymbol{r}^n$
3. $\overline{\mathcal{U}}\ \boldsymbol{\delta}^{n+1} = \boldsymbol{w}^{n+1}$
4. $\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + \boldsymbol{\delta}^{n+1}$

In the modified line 1, the $\langle\ \rangle$ corresponds to the additional operations due to the presence of immersed boundaries. These operations are carried out only for irregular grid points. Furthermore, $\overline{\mathcal{L}}$ and $\overline{\mathcal{U}}$ are the lower and upper diagonal matrices obtained for the modified matrix $\overline{\mathcal{A}}$ according to the MSI algorithm. The procedure just outlined was empirically determined, and was found to converge in all cases in which it was employed to solve the Poisson equation with immersed boundaries. It is a convenient algorithm that allows the MSI procedure to be performed on the regular matrix $\overline{\mathcal{A}}$, yet it easily accommodates the irregularly located entries introduced by immersed boundaries.

5.2. Restriction operator

A restriction operator transfers the residuals from the fine grid to the coarser grid. A simple and fast but not robust restriction operator is the direct injection where the residual values at the fine grid points are "injected" at the corresponding coarse grid points. A more accurate and robust restriction operator is full weighting where the residuals on a coarser grid are computed by averaging the residual values on the fine grid points surrounding the coarse one. Full weighting restriction is generally beneficial to the properties of the multigrid method in which it provides extra smoothing to that of the basic iterative scheme. However, a full weighting restriction operator developed for uniform grids is not suitable for non-uniform grids as its inexactness (because it is derived for uniform grid) will deteriorate the efficiency and convergence rate of the standard multigrid and MSMG methods [30]. Grid points on coarse grids are also grid points on finer grids. Therefore, the reference grid point (i, j) on the finer grids has the most contributions to its equivalent point on the coarse grids. However, an effective and efficient full weighting restriction operator should take into account that the closer adjacent grid points (smaller areas) have more contributions than those far away from it (bigger areas). Therefore, to fully exploit the accuracy and efficiency of the MSMG method, we use the full weighting operator on non-uniform grids based on the area-ratio law given by,

$$\overline{r}_{i,j} = \frac{1}{S}\Big(S_{NW}\, r_{i+1,j-1} + S_N\, r_{i,j-1} + S_{NE}\, r_{i-1,j-1} + S_W\, r_{i+1,j} + S_P\, r_{i,j} + S_E\, r_{i-1,j} + S_{SW}\, r_{i+1,j+1} + S_S\, r_{i,j+1} + S_{SE}\, r_{i-1,j+1} \Big) . \tag{84}$$

The areas used in the restriction operator on non-uniform grids are marked in Fig. 10(a), in which S is the total area

$$S = S_{NW} + S_N + S_{NE} + S_W + S_P + S_E + S_{SW} + S_S + S_{SE} . \tag{85}$$

The "sub" areas are given by

$$\begin{aligned} S_{SW} &= \frac{dx_b\, dy_b}{4} , & S_S &= \frac{dx\, dy_b}{4} , & S_{SE} &= \frac{dx_f\, dy_b}{4} , \\ S_W &= \frac{dx_b\, dy}{4} , & S_P &= \frac{dx\, dy}{4} , & S_E &= \frac{dx_f\, dy}{4} , \\ S_{NW} &= \frac{dx_b\, dy_f}{4} , & S_N &= \frac{dx\, dy_f}{4} , & S_{NE} &= \frac{dx_f\, dy_f}{4} , \end{aligned} \tag{86}$$

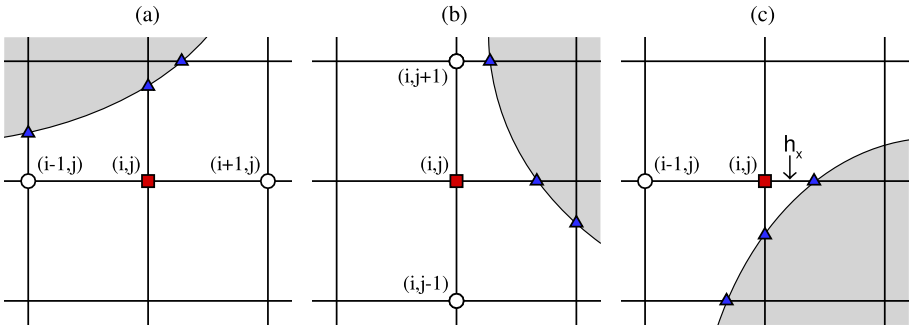

Fig. 11. Grid points used to compute the residual value at the irregular grid point (i, j). Eqs. (87)–(89) are used in (a), (b) and (c), respectively.

where $dx_f = x_{i+1} - x_i$, $dx_b = x_i - x_{i-1}$, $dy_f = y_{j+1} - y_j$, $dy_b = y_j - y_{j-1}$, $dx = dx_f + dx_b$, and $dy = dy_f + dy_b$. The restriction operator needs to be adjusted in the vicinity of the immersed boundary. For the irregular grid points, we use one-dimensional, second-order restriction operator. For illustration purposes, we consider three different cases as shown in Fig. 11. To obtain the residual value on the coarse grid point (i, j) in Fig. 11(a), Eq. (87) is applied, where the residual values along the $y = y_j$ are used. The residual values along the $x = x_i$ are used in Eq. (88) to compute the residual at the irregular grid point (i, j) in Fig. 11(b). Using the fact that the residual is zero at the IBI point, we can find the residual for the irregular grid point (i, j) in Fig. 11(c) using Eq. (89).

$$\overline{r}_{i,j} = \frac{dx_f}{dx_f + dx_b} r_{i-1,j} + \frac{1}{2} r_{i,j} + \frac{dx_b}{dx_f + dx_b} r_{i+1,j} \,, \tag{87}$$

$$\overline{r}_{i,j} = \frac{dy_f}{dy_f + dy_b} r_{i,j-1} + \frac{1}{2} r_{i,j} + \frac{dy_b}{dy_f + dy_b} r_{i,j+1} \,, \tag{88}$$

$$\overline{r}_{i,j} = \frac{h_x}{h_x + dx_b} r_{i-1,j} + \frac{1}{2} r_{i,j} \,. \tag{89}$$

In Eq. (89), $h_x = x_{IBI} - x_i$, where x_{IBI} is the x-location of the IBI point between the irregular grid point (i, j) and solid grid point $(i + 1, j)$ in Fig. 11(c).

5.3. Prolongation operator

The prolongation operator interpolates the corrections from coarse grids to fine grids using the nearest coarse grid neighbors. The quality of the prolongation operator plays a fundamental role for the convergence of the multigrid method. Similar to the restriction operator, a standard bilinear interpolation operator designed for uniform grids is not suitable for non-uniform grids as the convergence rate will deteriorate. Therefore, the area-ratio prolongation operator is used here. Grid points on coarse grids are also grid points on finer grids as shown in Fig. 10(b), where the filled circles denote the grid points on the coarse grid and the open circles are the grid points on the fine grid. For grid points that belong to the coarse and fine grids, the corrections are transferred directly from the coarse grid to the fine grid, as follows:

$$\delta_{i,j} = \overline{\delta}_{m,n} \,. \tag{90}$$

Linear interpolation is used to determine the corrections for the grid points on the fine grid located between two coarse neighbors,

$$\delta_{i,j-1} = \frac{dy_f}{dy_f + dy_b} \overline{\delta}_{m,n-1} + \frac{dy_b}{dy_f + dy_b} \overline{\delta}_{m,n} \,, \tag{91}$$

$$\delta_{i-1,j} = \frac{dx_f}{dx_f + dx_b} \overline{\delta}_{m-1,n} + \frac{dx_b}{dx_f + dx_b} \overline{\delta}_{m,n} \,, \tag{92}$$

where the notation used in the equations are defined in Fig. 10(b). Finally, for fine grid points surrounded by four coarse points, we use the four coarse grid points for interpolation,

$$\delta_{i-1,j-1} = \frac{1}{S} \left(S_1 \, \overline{\delta}_{m-1,n-1} + S_2 \, \overline{\delta}_{m,n-1} + S_3 \, \overline{\delta}_{m,n} + S_4 \, \overline{\delta}_{m-1,n} \right) \,, \tag{93}$$

in which the areas are defined as

$$S_1 = dx_f \, dy_f \,, \quad S_2 = dx_b \, dy_f \,, \quad S_3 = dx_b \, dy_b \,, \quad S_4 = dx_f \, dy_b \,. \tag{94}$$

In Eq. (93), S is the total area, $S = S_1 + S_2 + S_3 + S_4$. Similar to the restriction operator, a modification needs to be applied to the prolongation operator to accommodate for the immersed boundary. When the immersed boundary passes between a fine grid point and one of two neighbor coarse grid points, the 1-D linear interpolation between the IBI point and the coarse grid point can be used to find the correction on the fine grid point. The interpolation can be even more simplified using the fact that the correction is zero on the IBI point.

6. Numerical experiments

In this section, numerical experiments are conducted to demonstrate the accuracy and efficiency of the new method for solving Eq. (1) on uniform and non-uniform grids with immersed boundaries. Sample problems presented in Sections 6.1 to 6.4 are solved on the domain $\Omega = [-1, 1] \times [-1, 1]$ with the same number of grid points in the x and y directions. Node coordinates are given by

$$x_i = -1 + 2\frac{i-1}{N-1} + \frac{\alpha_x}{2\pi} sin\left(2\pi \frac{i-1}{N-1}\right), \quad 1 \le i \le N, \tag{95}$$

$$y_j = -1 + 2\frac{j-1}{N-1} + \frac{\alpha_y}{2\pi} sin\left(2\pi \frac{j-1}{N-1}\right), \quad 1 \le j \le N, \tag{96}$$

where N is the number of grid points in each direction and α_x and α_y control the non-uniformity of the grid in x and y directions, respectively. For those problems with exact solutions, the right-hand function $\rho(x, y)$ and the Dirichlet boundary conditions on $\partial\Omega$ and Γ are prescribed to satisfy the given exact solution. For the multigrid V-cycle in the context of the MSMG method, one iteration (relaxation) is performed at each grid level before projecting the residual to the coarser grid (pre-smoothing) and one relaxation after interpolating the solution back to a finer grid level (post-smoothing). For the final stage (multigrid V-cycle on Ω_Δ), two iterations are used for pre- and post-smoothing. The order of accuracy is computed using the definition

$$p = \frac{log(||E_{N_1}||/||E_{N_2}||}{log(N_2/N_1)}, \tag{97}$$

where $||E_{N_1}||$ and $||E_{N_2}||$ are norms of the errors between the analytical, $u^{(ex)}$, and the numerical solutions for two different grids with N_1 and N_2 points in the x directions, computed as follows

$$||E_N||_\infty = \max_{1\le j\le N} \max_{1\le i\le N} |u_{i,j} - u^{(ex)}_{i,j}|. \tag{98}$$

Eq. (98) corresponds to the infinity-norm. The last point to mention is that the solution inside the immersed body, Ω^-, is trivial and set to zero in the present work.

6.1. Computational performance

To demonstrate the accuracy of the new method and efficiency of the MSMG solver on a non-uniform grid, solving the following Poisson equation with Dirichlet boundary conditions is considered:

$$\begin{aligned} u_{xx} + u_{yy} &= 12(10r^2 - 1)e^{-10r^2} + \sum_{k=1}^{4} 40(10r_k^2 - 1)e^{-10r_k^2}, \qquad \text{in } \Omega^+, \\ u(x, y) &= 0.3e^{-10r^2} + \sum_{k=1}^{4} e^{-10r_k^2}, \qquad\qquad\qquad \text{on } \Gamma \,\&\, \partial\Omega, \end{aligned} \tag{99}$$

where $r = \sqrt{x^2 + y^2}$ and each r_k is given by

$$r_1 = \sqrt{x^2 + (y - 0.45)^2}, \quad r_2 = \sqrt{x^2 + (y + 0.45)^2}, \quad r_3 = \sqrt{(x - 0.45)^2 + y^2}, \quad r_4 = \sqrt{(x + 0.45)^2 + y^2}. \tag{100}$$

Eq. (99) is solved for two cases: (i) A simple domain without immersed boundary, and (ii) a domain with an immersed boundary. For the latter case, the domain Ω^+ is the region outside the immersed boundary Γ defined by the parametric equations,

$$\begin{aligned} x_\Gamma(\theta) &= \left[0.305 + 0.117cos(4\theta)\right] cos(\theta), \\ y_\Gamma(\theta) &= \left[0.305 + 0.117cos(4\theta)\right] sin(\theta), \end{aligned} \tag{101}$$

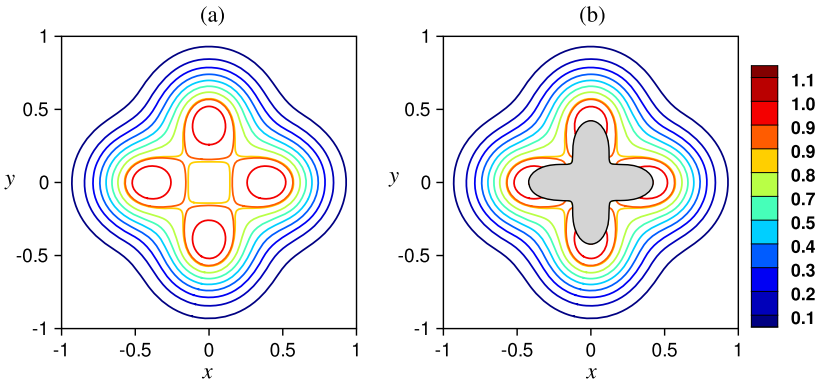

Fig. 12. Numerical solutions of Eq. (99) in the computational domain with 129 × 129 non-uniform grid points for two different cases: (*a*) Simple domain without immersed boundary, (*b*) domain with immersed boundary.

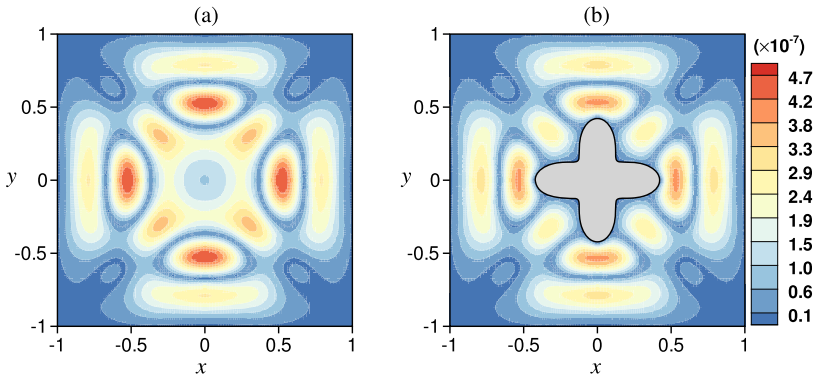

Fig. 13. Errors corresponding to numerical solutions of Eq. (99) in the computational domain with 129 × 129 non-uniform grid points for two different cases: (*a*) Simple domain without immersed boundary, (*b*) domain with immersed boundary.

Table 2
Maximum absolute error and the accuracy order of the current method for solutions of Eq. (99) for domains without and with immersed boundary (IB).

Grid	Case (i): Without IB		Case (ii): With IB	
	$\|\|E_N\|\|_\infty$	Order	$\|\|E_N\|\|_\infty$	Order
33 × 33	1.223×10^{-04}		8.518×10^{-05}	
65 × 65	7.541×10^{-06}	4.02	5.520×10^{-06}	3.95
129 × 129	4.698×10^{-07}	4.00	3.875×10^{-07}	3.83
257 × 257	2.934×10^{-08}	4.00	2.527×10^{-08}	3.94
513 × 513	1.835×10^{-09}	4.00	1.595×10^{-09}	3.99
1025 × 1025	1.148×10^{-10}	4.00	9.983×10^{-11}	4.00

for $0 \leq \theta \leq 2\pi$. The grid stretching parameter is the same for both directions and chosen to be $\alpha_x = \alpha_y = 0.7$. Computed solutions and the corresponding errors, $e(x, y) = |u(x, y) - u^{(ex)}(x, y)|$ are shown in Figs. 12 and 13, respectively, for both cases on a 129 × 129 non-uniform grid. The reader's attention is drawn to the sharp interface in the solution obtained, and the relative smoothness of the error distribution in the vicinity of the immersed boundary (see Fig. 13(b)).

The solution and the corresponding error is computed on several grids, and the errors in the infinity-norm, $||E_N||_\infty$, is presented as a function of grid size in Table 2. The results confirm the fourth-order accuracy of the new method for both the simple domain and, most importantly, also for the domain with the immersed boundary. No loss of accuracy for the domain with immersed boundary is observed.

In the following, the convergence characteristics and the computational performance of the MSMG strategy is compared with the standard multigrid V-cycle method for solving Eq. (99) with immersed boundary. To allow for a fair comparison, two iterations for pre- and post-smoothing are used for the standard multigrid method, similar to the last level of the MSMG method. Fig. 14 displays the convergence behavior of both methods for various grid sizes. Shown is the infinity-norm of the error after each V-cycle. The multigrid method convergence rate is almost identical for different grid sizes as shown in Fig. 14(a). However, more V-cycle iterations are required to achieve higher accuracy on finer grids. At the same time, the MSMG converges to the discretization error in a fixed number of iterations independent of the grid size as illustrated in

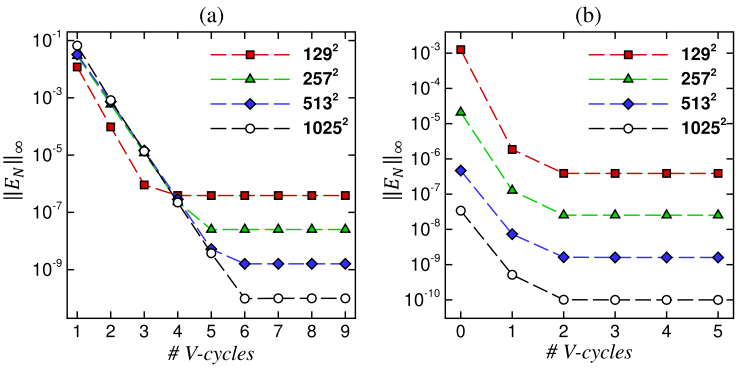

Fig. 14. Convergence behavior of the numerical error versus V-cycle iteration number for various grid sizes for the domain with the immersed boundary. (*a*) Standard multigrid method, (*b*) MSMG method.

Table 3
Comparison of computation times between the standard multigrid method and the MSMG method for domain with immersed boundary.

Grid size →	129 × 129	257 × 257	513 × 513	1025 × 1025
Standard multigrid method	0.0245	0.1621	0.8372	3.5956
The MSMG method	0.0146	0.0741	0.3454	1.4137
Run-time ratio	0.596	0.457	0.413	0.393

Table 4
Comparison of computation times for problem with immersed boundary and without immersed boundary with the MSMG method.

Grid size →	129 × 129	257 × 257	513 × 513	1025 × 1025
Simple domain	0.0143	0.0720	0.3385	1.4043
With immersed boundary	0.0146	0.0741	0.3454	1.4137
(%) increase	2.945	2.916	2.056	0.674

Fig. 14(b). Note that in Fig. 14(b), the number of V-cycle iterations on level Ω_Δ is considered for the MSMG method and zero V-cycle corresponds to the error in the initial approximation which is interpolated from $\Omega_{2\Delta}$.

While the standard multigrid method includes only V-cycles on Ω_Δ, the MSMG method is composed of inexpensive V-cycles on several coarser grid levels, the V-cycle on the finest grid Ω_Δ and solution interpolation between grids as well (see Fig. 9). Regarding the total computational cost of the method to reduce the error to the level of discretization error, we compare the CPU time of the two methods presented in Table 3. The number of V-cycles can be obtained from Fig. 14 for various grid sizes. For the standard multigrid method, therefore, four and five V-cycles are needed for grid sizes of 129^2 and 257^2, respectively, and six V-cycles are required for grids with 513^2 and 1025^2 points. The MSMG method requires only two V-cycles on Ω_Δ for all grid sizes. It should be remembered that one V-cycle is used for the coarser grids in the MSMG method (see Fig. 9). For the test cases considered, the MSMG method uses less CPU time and the ratio of CPU time between the two methods decreases as the grids get finer. For the finest grid, 1025^2, the MSMG run time is about 39% of the standard multigrid run time. This demonstrates that it is paying off to use the highest quality initial guess possible.

Another important aspect of the solution of the Poisson equation with the MSMG method is the time penalty associated with adding immersed boundaries. Table 4 presents run times to compute two V-cycles in the last stage of the MSMG method for the problem with and without immersed boundary. For the problem sizes of interest, Table 4 shows that the extra computational time due to the presence of immersed boundary is negligible for the MSMG solver. For the largest grid size investigated here, the extra CPU time is less than 1%.

The relaxation operator constitutes the most expensive part of the MSMG method. With a proper design of this operator which accommodates the immersed boundary without loss of performance of the original algorithm, the new solution strategy is proven to be equally efficient for domains with immersed boundaries and for simple domains (without immersed boundaries).

6.2. Slow and rapid oscillatory source term

In this example, solving the Helmholtz equation with Dirichlet boundary conditions is considered,

$$\begin{aligned} u_{xx} + u_{yy} - u &= -(1 + 8\pi^2)\, sin(2\pi x)\, cos(2\pi y)\,, \qquad \text{in } \Omega^+\,, \\ u(x, y) &= sin(2\pi x)\, cos(2\pi y)\,, \qquad \text{on } \Gamma \,\&\, \partial\Omega\,. \end{aligned} \tag{102}$$

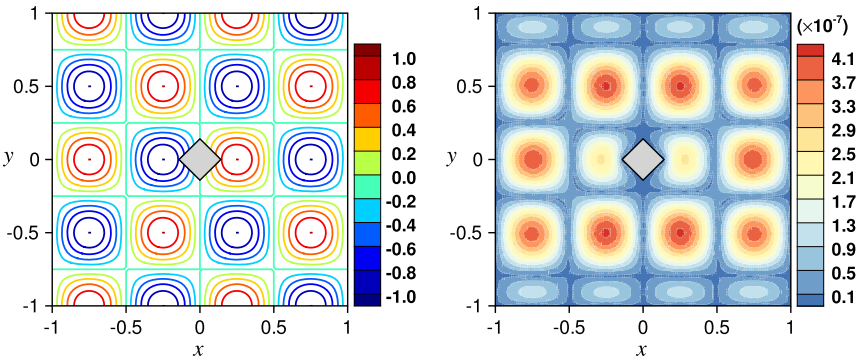

Fig. 15. Numerical solution and the corresponding error of Eq. (102) in the computational domain with 129 × 129 uniform grid points for case (i).

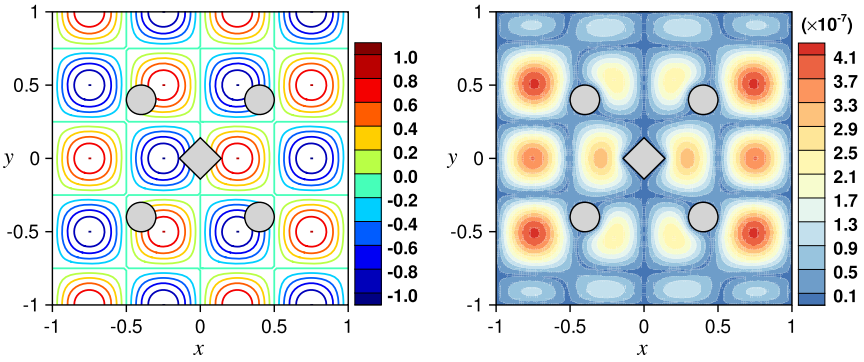

Fig. 16. Numerical solution and the corresponding error of Eq. (102) in the computational domain with 129 × 129 uniform grid points for case (ii).

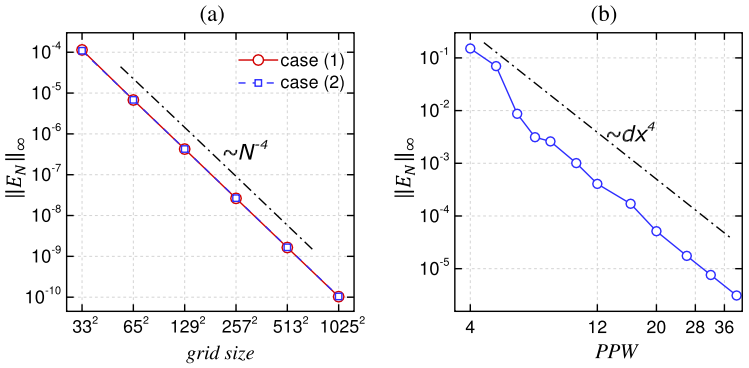

Fig. 17. (*a*) Numerical error in the infinity-norm as a function of grid points for solutions of Eq. (102). (*b*) Numerical error in the infinity-norm as a function of points per wavelength (PPW) for solutions of Eq. (103). The dashed-dot line represents the 4th-order slope.

Eq. (102) is solved on a uniform grid in x and y directions for two different cases: (i) Domain with one immersed square with a dimension of 0.2 located at the center (see Fig. 15), and (ii) same as case (i) with additionally four immersed circles with radius 0.1 (see Fig. 16). Figs. 15 and 16 show solutions of Eq. (102) and the corresponding errors for a uniform grid with 129 × 129 points, respectively. The solutions are sharp across the interfaces and the local error has a smooth distribution in the vicinity of the immersed boundary. The results of the error study for the MSMG solver are presented in Fig. 17(a), which demonstrates that the method is fourth-order-accurate based on the error measured in the infinity-norm.

The wavelength associated with the source term of Eq. (102) is $\lambda_x = 1$, therefore the number of points per wavelength (PPW) for the coarsest grid (33 × 33) is $\lambda_x/dx = 16$. As a result, the biased stencil applied at the irregular grid points does not degrade the accuracy since the PPW is large enough. It is necessary, therefore, to examine the accuracy of the current method for a highly oscillatory source term where the PPW could become very small. Hence, case (ii) with five immersed bodies is used for the following example:

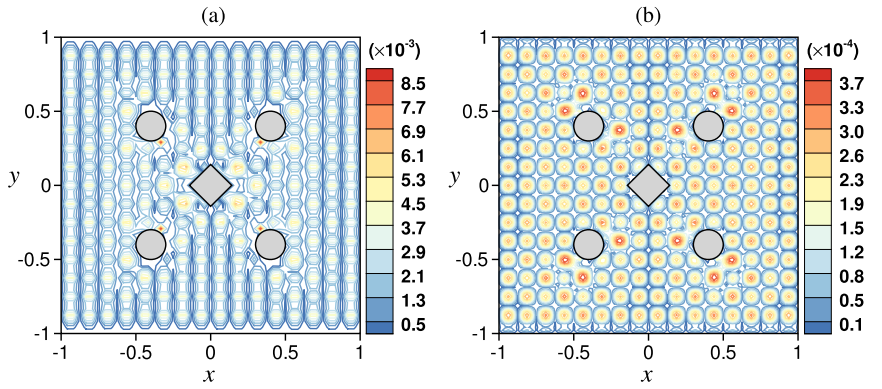

Fig. 18. Numerical errors of Eq. (103). (*a*) 49×49 uniform grid points, $PPW = 6$. (*b*) 97×97 uniform grid points, $PPW = 12$.

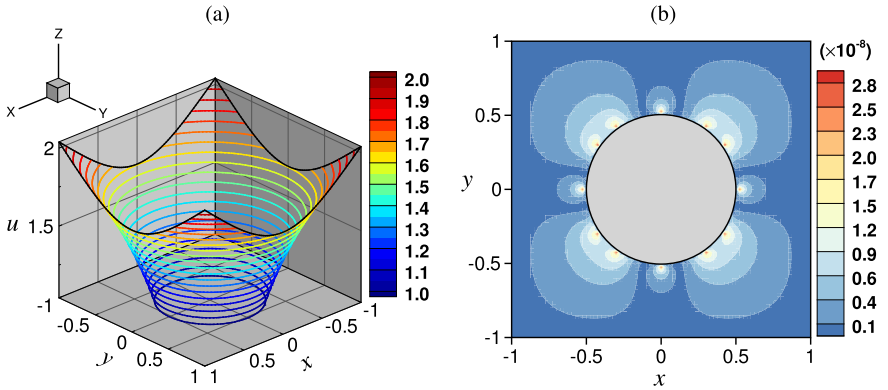

Fig. 19. Numerical solution (*a*) and contours of local error (*b*) computed for Eq. (104) on 80×80 uniform grid points.

$$\begin{aligned} u_{xx} + u_{yy} - u &= -(1 + 128\pi^2)\, sin(8\pi x)\, cos(8\pi y)\,, \qquad \text{in } \Omega^+\,, \\ u(x, y) &= sin(8\pi x)\, cos(8\pi y)\,, \qquad\qquad \text{on } \Gamma \,\&\, \partial\Omega\,. \end{aligned} \tag{103}$$

The solution and the corresponding error are computed on a set of uniform grids form 33×33 points up to 321×321 points. Thus, the PPW, $(N-1)/8$, varies between 4 and 40. The errors in the infinity-norm, $||E_N||_\infty$, is presented as a function of PPW in Fig. 17(b). It reveals that the current method recovers its fourth-order accuracy even for the small PPW. Fig. 18 shows the contour lines of the numerical errors for a uniform grid with 49×49 and 97×97 which results in $PPW = 6$ and $PPW = 12$, respectively.

6.3. Case from LeVeque and Li [11]

Next, the following example is considered which was used by LeVeque and Li [11] and Zhong [19] to test their IIM methods:

$$\begin{aligned} u_{xx} + u_{yy} &= 0\,, \qquad\qquad \text{in } \Omega^+\,, \\ u(x, y) &= 1 + log(2r)\,, \qquad \text{on } \Gamma \,\&\, \partial\Omega\,, \end{aligned} \tag{104}$$

where $r = \sqrt{x^2 + y^2}$ and the interface Γ is a circle defined by $x^2 + y^2 = 1/4$. The numerical solution computed by the MSMG method using 80×80 grid point as well as the contours of the local errors are shown in Fig. 19. Fig. 19(b) indicates that the largest errors in the computations occur at the interface. Similar behavior was observed by Zhong [19]. Table 5 summarizes the numerical errors of the current method for two sets of uniform grids: 40×40 and 80×80 based on the infinity-norm, as well as the results of LeVeque and Li's original second-order immersed interface method, the delta function approximation results and two different fourth-order approximations by Zhong [19]. For the grid with 80×80 points, the results demonstrate the much higher-accuracy of our present method compared to the other methods: Four orders of magnitudes lower than the LeVeque and Li IIM method, two and one order of magnitudes lower than Zhong's fourth-order IIM methods *E* and *F*, respectively. Note that the error ratio in Table 5 is the ratio of the errors of two successive sets of grids.

Table 5
Comparison of the computational errors of the current method and those of the original IIM of LeVeque and Li [11] and 4th-order-accurate IIM of Zhong [19]. The errors are measured in the infinity-norm.

Method	$\|\|E_N\|\|_\infty$ (40×40)	$\|\|E_N\|\|_\infty$ (80×80)	Error ratio	Order
δ function method	2.6467×10^{-2}	1.3204×10^{-2}	2.0045	1.003
LeVeque & Li's IIM $\left[O(dx^2)\right]$	8.3461×10^{-4}	2.4451×10^{-4}	3.4134	1.771
Zhong's IIM Method E $\left[O(dx^4)\right]$	1.2215×10^{-4}	6.1514×10^{-6}	19.8574	4.311
Zhong's IIM Method F $\left[O(dx^4)\right]$	1.5521×10^{-5}	3.4286×10^{-7}	45.2678	5.500
Current Method $\left[O(dx^4)\right]$	1.3621×10^{-6}	2.8143×10^{-8}	48.3992	5.597

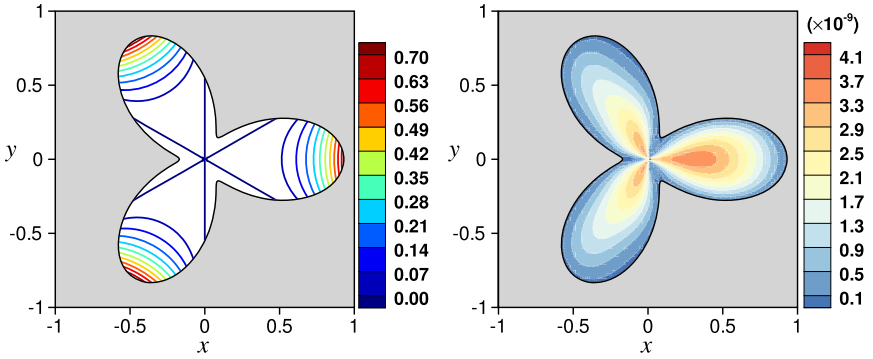

Fig. 20. Numerical solution and error of Eq. (105) on 129×129 non-uniform grid with irregular domain boundary. (*a*) Computed solution, (*b*) the corresponding local error.

Table 6
Maximum absolute error and the accuracy order of the current method for example presented in Section 6.4.

Grid size	$\|\|E_N\|\|_\infty$	Error ratio	Order
65×65	6.5810×10^{-08}		
129×129	4.1004×10^{-09}	16.05	4.004
257×257	2.5116×10^{-10}	16.33	4.029
513×513	1.5433×10^{-11}	16.27	4.024
1025×1025	9.6101×10^{-13}	16.06	4.005

6.4. Irregular domain boundary

In this example, the following equation with Dirichlet boundary condition is solved on a non-uniform grid with an irregular domain boundary:

$$\begin{aligned} u_{xx} + u_{yy} &= 7r^2\, cos(3\theta)\ ,\text{in}\ \ \Omega^+\ , \\ u(x,y) &= r^4\, cos(3\theta)\ ,\quad \text{on}\ \ \Gamma\ , \end{aligned} \tag{105}$$

where $r = \sqrt{x^2 + y^2}$ and the domain Ω^+ is the region inside the boundary Γ defined by the parametric equations,

$$\begin{aligned} x_\Gamma(\theta) &= \left[0.55 + 0.38cos(3\theta)\right] cos(\theta)\ , \\ y_\Gamma(\theta) &= \left[0.55 + 0.38cos(3\theta)\right] sin(\theta)\ , \end{aligned} \tag{106}$$

for $0 \le \theta \le 2\pi$. Node coordinates for non-uniform grids are given by Eqs. (95)–(96) with $\alpha_x = \alpha_y = 0.5$. In contrast to the other problems, Eq. (105) is solved inside the immersed boundary. Note that fourth-order derivatives of u are discontinuous at the origin, and higher-order derivatives are singular.

The numerical solutions and error obtained for Eq. (105) are illustrated in Fig. 20 for a grid size of 129×129. Different grid sizes ranging from 65^2 to 1025^2 were used to evaluate the order of accuracy presented in Table 6. It can be observed that the new method preserves its fourth-order accuracy for this problem as well.

6.5. Comparison with fourth-order IIM by Li and Ito [20]

In this example, the proposed compact difference scheme is examined for the two cases presented by Li and Ito [20] in order to test their fourth-order-accurate IIM on irregular domains. The two cases are,

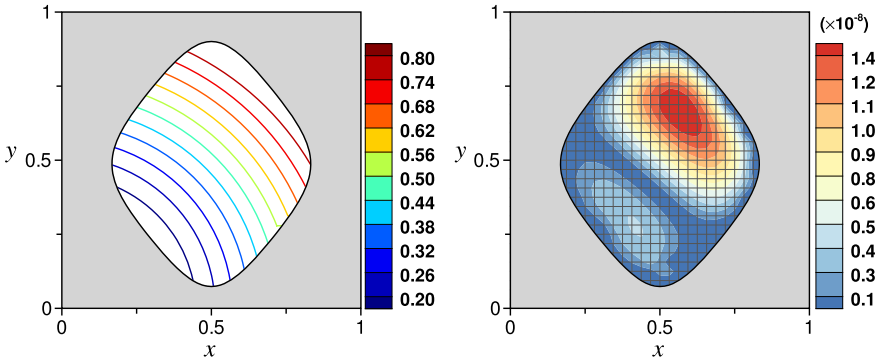

Fig. 21. Numerical solution (left plot) and the corresponding error (right plot) of Eq. (107) in the computational domain with 33×33 uniform grid for case (i) of Li and Ito [20].

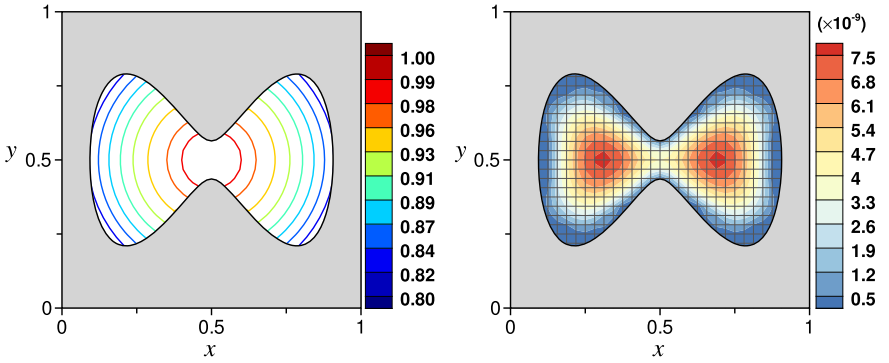

Fig. 22. Numerical solution (left plot) and the corresponding error (right plot) of Eq. (108) in the computational domain with 33×33 uniform grid for case (ii) of Li and Ito [20].

Case (i):

$$\begin{aligned} u_{xx} + u_{yy} &= 4cos(x^2+y^2) - 4(x^2+y^2)sin(x^2+y^2)\,, \quad \text{in } \Omega^+\,, \\ u(x,y) &= sin(x^2+y^2)\,, \qquad\qquad\qquad\qquad \text{on } \Gamma\,, \\ \Gamma &= \left\{(x,y)|(1.95y-0.95)^2 + (4.1x-2.05)^2/[1+(4.1x-2.05)^2] - 0.65 = 0\right\}. \end{aligned} \tag{107}$$

Case (ii):

$$\begin{aligned} u_{xx} + u_{yy} &= -4(1+2r)\,exp(-r)\,, \quad \text{in } \Omega^+\,, \\ u(x,y) &= exp(-r)\,, \qquad\qquad\qquad \text{on } \Gamma\,, \\ \Gamma &= \left\{(x,y)|0.5(2.5y-1.25)^2 - 0.5(3.5x-1.75)^2 + 0.25(3.5x-1.75)^4 - 0.012866 = 0\right\}, \end{aligned} \tag{108}$$

where $r = (x-0.5)^2 + (y-0.5)^2$. In both cases, the domain Ω^+ is the region inside the boundary Γ defined by the implicit equations. Figs. 21 and 22 show the numerical solutions and the corresponding errors computed for a uniform grid with $dx = dy = 1/32$ for cases (i) and (ii), respectively. A grid refinement study based on the infinity-norm for numerical errors, $||E_N||_\infty$, has been conducted by using three sets of uniform grids: 17×17, 33×33 and 65×65. Table 7 summarizes the numerical errors of the current method, as well as the results of Li and Ito's fourth-order immersed interface method. Of particular interest is the observation that while the fourth-order accuracy is recovered in both schemes and for both cases, the numerical solutions of the present method have much higher accuracy. For three sets of grids investigated, the errors for the present method are two orders of magnitudes lower than those of the IIM by Li and Ito [20].

6.6. Compact scheme versus fourth-order standard (non-compact) finite-difference method

Most of the high-order immersed interface methods for elliptic equations with discontinuous and non-smooth solutions were developed based on the standard (non-compact) finite-difference approximations [17–19]. One of the important applications of these IIMs is that they can be applied for numerical solutions of Eq. (1) defined in an irregular domain or a domain with an immersed boundary where the solution inside the region Ω^- is trivial. Therefore, it is useful to compare the

Table 7
Comparison of the numerical error of the current method and the fourth-order-accurate IIM of Li and Ito [20] for Eqs. (107) and (108).

dx	*Case (i)*				*Case (ii)*			
	Li and Ito's IIM		Current method		Li and Ito's IIM		Current method	
	$\|\|E_N\|\|_\infty$	Order	$\|\|E_N\|\|_\infty$	Order	$\|\|E_N\|\|_\infty$	Order	$\|\|E_N\|\|_\infty$	Order
1/16	1.013×10^{-4}		2.100×10^{-7}		1.068×10^{-4}		1.636×10^{-7}	
1/32	6.924×10^{-6}	4.13	1.482×10^{-8}	3.83	1.250×10^{-6}	6.42	7.708×10^{-9}	4.41
1/64	1.688×10^{-7}	5.36	8.950×10^{-10}	4.05	3.576×10^{-8}	5.13	5.327×10^{-10}	3.86

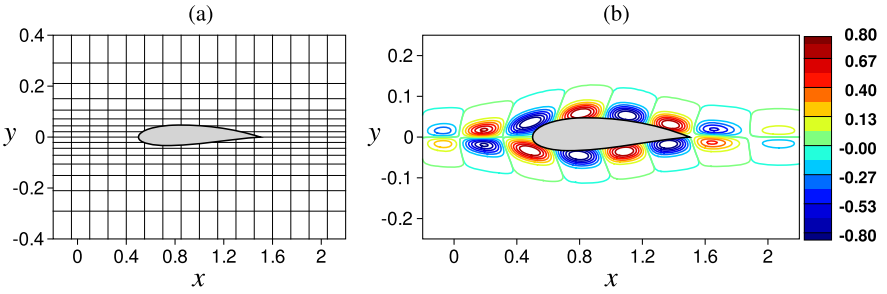

Fig. 23. (*a*) Computational domain used in Section 6.6. (*b*) Numerical solution of the Poisson equation for test case in Section 6.6 obtained from the compact difference scheme on 129×129 non-uniform grid points.

present fourth-order compact scheme to its equivalent non-compact method in terms of accuracy and computational efficiency. Towards this end, a fourth-order accurate standard finite difference (SFD) scheme was developed (see Appendix A). It is worth noting that the resulting coefficient matrix is also nine-diagonal for a simple domain. However, unlike the compact scheme, where the discretization of the grid point centered at (i, j) includes adjacent points $(i \pm 1, j \pm 1)$, the SFD scheme requires grid points $(i \pm 1, j)$, $(i \pm 2, j)$, $(i, j \pm 1)$ and $(i, j \pm 2)$. Consequently, the MSI procedure cannot be employed as a relaxation operator in the MSMG method (or multigrid method). Thus, Successive Over Relaxation (SOR) method was used as a smoother operator in the MSMG method for the SFD scheme. The Poisson equation with the following exact solution is solved using both the new compact and the SFD schemes:

$$\begin{aligned} u(x,y) &= sin\big(\omega(x-y)\big)e^{-\sigma\left(y-s_1(x)\right)^2} - sin\big(\omega(x+y)\big)e^{-\sigma\left(y-s_2(x)\right)^2}, \quad \text{on } \Gamma \,\&\, \partial\Omega\,, \\ s_1(x) &= 0.060e^{-3(x-0.9)^2}\,, \\ s_2(x) &= 0.045e^{-3(x-0.8)^2}\,, \end{aligned} \tag{109}$$

with $\omega = 10\pi/3$ and $\sigma = 2000$. In this example, the computational domain is $[-0.2, 2.2] \times [-0.4, 0.4]$ and the immersed body is the NACA 1608 airfoil [31] as shown in Fig. 23(a). This problem represents one of the canonicals test case of numerical investigations of the external flow over a wing section in computational fluid dynamics (CFD). The Poisson equation is solved on a uniform grid in the x direction while a non-uniform grid is used in the y direction, for which the stretching function is given by

$$y_j = -0.4 + 0.4\left[1 + \frac{sinh(\mu(\eta - S))}{sinh(\mu S)}\right], \quad \eta = \frac{j-1}{ny-1}\,, \quad 1 \le j \le ny\,. \tag{110}$$

The parameter S is defined as

$$S = \frac{1}{2\mu} log\left[\frac{1 + 0.5(e^{\mu} - 1)}{1 + 0.5(e^{-\mu} - 1}\right]. \tag{111}$$

The parameter μ controls the clustering of the grid near $y = 0$. It is set to $\mu = 5$ which puts more grid points around $y = 0$. Fig. 23(b) displays the numerical solution computed by the compact difference method for a grid with 129×129 points. Numerical solution with no visible differences is also obtained from the non-compact difference scheme, so it is not shown here. Contours of the local numerical errors for two schemes are provided in Fig. 24 for a grid with 129×129 points.

Different grid sizes ranging from 33^2 to 1025^2 were used to evaluate the order of accuracy for the compact and SFD schemes as summarized in Table 8. It can be observed that both methods preserve their fourth-order accuracy for this problem. However, attention is drawn to the higher-accuracy of the present method (compact difference) compared to the SFD scheme. It is important to mention that the SFD solver was applied to the previous problems as well, and that for all cases the new compact difference scheme resulted in solutions with higher accuracy compared to the standard (non-compact) scheme.

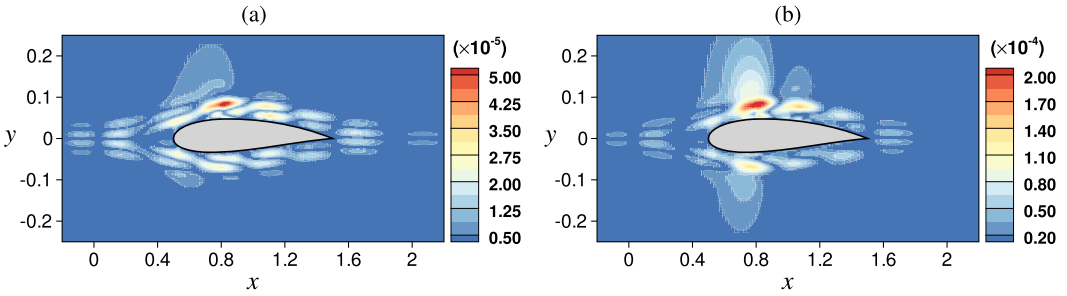

Fig. 24. Contours of numerical errors for problem in Section 6.6 on 129×129 non-uniform grid points with immersed body. (*a*) Compact difference scheme, (*b*) standard difference scheme. Note the contour levels of the errors for two schemes.

Table 8
Maximum absolute error and the accuracy order of the current method and the standard finite-difference scheme for test case in Section 6.6.

Grid	Compact difference scheme		Standard difference scheme	
	$\|\|E_N\|\|_\infty$	Order	$\|\|E_N\|\|_\infty$	Order
33×33	5.084×10^{-2}		1.450×10^{-1}	
65×65	1.715×10^{-3}	4.89	7.096×10^{-3}	4.35
129×129	5.865×10^{-5}	4.87	2.399×10^{-4}	4.89
257×257	2.651×10^{-6}	4.47	7.160×10^{-6}	5.07
513×513	1.634×10^{-7}	4.02	4.430×10^{-7}	4.01
1025×1025	1.015×10^{-8}	4.01	2.724×10^{-8}	4.02

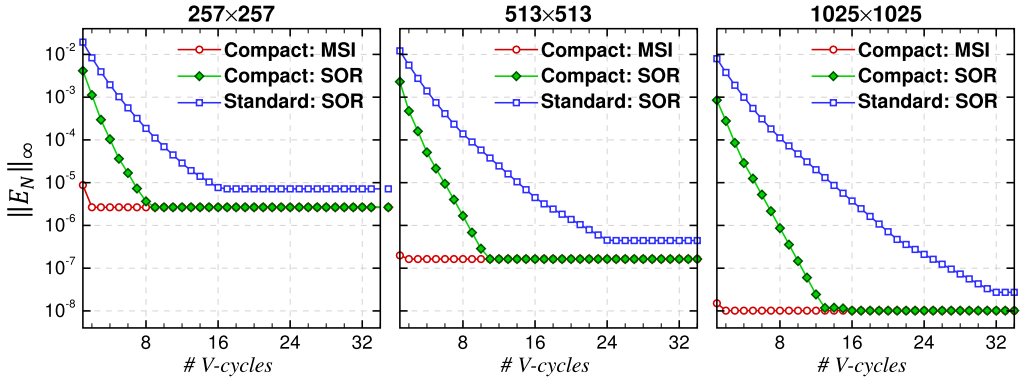

Fig. 25. Convergence behavior of the numerical error as a function of the number of V-cycle iterations on level Ω_Δ for various grid sizes for problem in Section 6.6. Compact: "MSI" and Compact: "SOR" corresponds to compact difference scheme solved with the MSMG method with the MSI and SOR methods as relaxation operators, respectively. Standard: "SOR" represents the standard (non-compact) difference scheme with the MSMG method and the SOR method as a relaxation operator.

Judging from the numerical errors obtained from both schemes in Table 8, whereas the compact difference scheme gives more accurate solutions, it implies that the compact difference method does not have significant advantage over the SFD method for the solution of Eq. (1) for problems involving domains with immersed bodies or irregular domains. Therefore, in addition to accuracy, the second objective is a comparison of the computational efficiency of two schemes. Fig. 25 displays the convergence behavior of both methods for various grid sizes. Shown is the infinity-norm of the error as a function of V-cycle iterations on level Ω_Δ in the MSMG method. In agreement with example presented in Section 6.1 where the MSMG converged only in two V-cycles for all grid levels, the MSMG method with the MSI iterative solver applied to the compact scheme converges to the discretization error in a fixed number of iterations independent of the grid size, i.e. in only two V-cycles. On the other hand, the non-compact method with SOR as the relaxation operator in the MSMG method, requires many more V-cycle iterations to converge. It should be noted that the number of iterations for pre- and post-smoothing used for the SOR method is double of that used in the MSI procedure to allow for a fair comparison in terms of required computer time. Therefore, the computer time spent for every V-cycle is almost identical for the MSI and SOR methods. It is an important finding that the standard finite difference will lead to a significant increase in the computational time needed to solve Eq. (1). Therefore, for an incompressible Navier–Stokes solver (in the stream function–vorticity or velocity–vorticity formulations) where the total computational time needed is dominated by solving the Poisson equation(s), the fourth-order SFD is not suitable due to its high computational time demand.

Furthermore, included in Fig. 25 is the convergence characteristic of the MSMG method for the compact scheme where the SOR method is used as a relaxation operator. It demonstrates that the compact scheme with SOR method is still much more computationally efficient than the SFD scheme with SOR. This implies that the difference in the convergence behavior between the compact and non-compact schemes is not merely due to the different relaxation operators, but that the properties of the coefficient matrices play an important role. Another important observation is the superior convergence characteristic of the MSI procedure for the compact scheme compared to the SOR method.

Finally, it is important to mention one of the main difficulties dealing with the fourth-order SFD method, namely that the coefficient matrix is not diagonally dominant even when employed for a simple domain with a uniform grid. As a result, this may lead to a robustness issues when iterative methods such as Gauss–Seidel or SOR methods are used. When we applied the fourth-order SFD scheme to the other problems involving immersed boundaries, it was noticed that for some cases it does not converge. To remedy that, different relaxation parameters in the SOR method were tested to find appropriate ones, so that convergence could be accomplished.

7. Conclusion

In this paper, the development of a sharp-interface and high-order method to solve the two-dimensional Poisson equation on non-uniform grids with immersed boundaries is presented. Fourth-order compact difference schemes are used to discretize the Poisson equation on the non-uniform mesh resulting in regular nine-point compact FD stencils. The sharp representation of the interface of the immersed body is accomplished by modifying the finite-difference stencils at irregular grid points near the immersed boundary. In particular, additional grid points are added to the regular compact nine-point FD stencils to obtain a sharp solution across the interface while retaining the fourth-order formal accuracy.

A new solution strategy is designed to solve the large sparse algebraic system that arose from the discretization of Poisson equation. This new idea is based on a multiscale multigrid method that reduces the error to the level of discretization error in a fixed number of iterations which is independent of the grid size. A standard multigrid V-cycle technique is used to build the MSMG method and the corresponding operators are properly derived and constructed on stretched grids with irregular boundaries. Specifically, the relaxation operator is modified to easily accommodate the irregularly located entries introduced by immersed boundaries without losing the performance of the original algorithm. The restriction and prolongation operators are designed for non-uniform meshes using area-ratio law and are adjusted near the immersed boundary.

The accuracy and the efficiency of the proposed method was demonstrated through verification using problems with analytical solutions for domains with and without immersed boundaries on uniform and non-uniform meshes. The numerical results demonstrate that the solutions are sharp across the interface and that the new method is fourth-order-accurate in the maximum-norm, including the irregular grid points near the interface of the immersed boundary. In addition, the solution technique was found to be very efficient in the sense that the same computational efficiency is obtained for problems with or without irregular boundaries.

In addition to the compact difference scheme, Eq. (1) was discretized by a fourth-order standard (non-compact) finite difference method. Accuracy and computational efficiency of the new method, compact difference scheme, was compared with the SFD scheme. The numerical results show that compact scheme produces more accurate numerical solutions. However, a striking difference observed in the numerical solutions of the Poisson equation for a domain with an immersed body with two different schemes was the fact that the number of multigrid V-cycle iterations needed for convergence was significantly lower for the new compact scheme compared to its equivalent fourth-order standard difference method.

The extension of the current method for solving the 3D Poisson equation with Dirichlet boundary conditions should present no particular difficulties beyond the increased complexity required to determine the intersections of the immersed boundary with the underlying Cartesian grid. Similar to the 2D problem, for an irregular grid point located at (i, j, k), modified FD operators need to be employed along each grid line in the range $x_{i-1} \leq x \leq x_{i+1}$, $y_{j-1} \leq y \leq y_{j+1}$ and $z_{k-1} \leq z \leq z_{k+1}$, if there is an intersection with the immersed boundary. Otherwise, the standard FD operators can be used. With respect to parallelization of the present method, an efficient and effective parallel implementation of the relaxation operator is required for which a large body of published research is available, e.g. [32–34].

Acknowledgements

This work was supported by the Air Force Office of Scientific Research (AFOSR) under grant number FA9550-14-1-0184, with Dr. Douglas Smith serving as the program manager.

Appendix A. Fourth-order standard finite-difference scheme

The fourth-order accurate standard (non-compact) finite difference approximations for second derivatives in x and y are

$$u_{xx_{i,j}} = a_i^x\, u_{i-2,j} + b_i^x\, u_{i-1,j} + c_i^x\, u_{i,j} + d_i^x\, u_{i+1,j} + e_i^x\, u_{i+2,j}\,, \tag{112}$$

$$u_{yy_{i,j}} = a_j^y\, u_{i,j-2} + b_j^y\, u_{i,j-1} + c_j^y\, u_{i,j} + d_j^y\, u_{i,j+1} + e_j^y\, u_{i,j+2}\,. \tag{113}$$

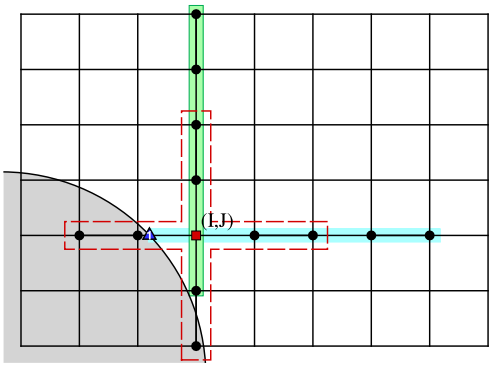

Fig. 26. Intersection of fourth-order accurate non-compact FD nine-point stencil at irregular grid point located at (i, j) with an immersed boundary. Grid points inside the colored rectangles are used to approximate second derivatives in the x and y directions. Immersed boundary intersection (IBI) point is numbered 1.

Coefficients in Eq. (112), obtained by matching the coefficients in the Taylor expansion about $u_{i,j}$ in the x-direction, are given by

$$\begin{aligned}
a_i^x &= \frac{2(h_2h_3+h_2h_4+h_3h_4)}{h_1(h_1-h_2)(h_1-h_3)(h_1-h_4)}\,,\\
b_i^x &= \frac{2(h_1h_3+h_1h_4+h_3h_4)}{h_2(h_2-h_1)(h_2-h_3)(h_2-h_4)}\,,\\
c_i^x &= \frac{2(h_1h_2+h_1h_3+h_1h_4+h_2h_3+h_2h_4+h_3h_4)}{h_1h_2h_3h_4}\,,\\
d_i^x &= \frac{2(h_1h_2+h_1h_4+h_2h_4)}{h_3(h_3-h_4)(h_3-h_2)(h_3-h_1)}\,,\\
e_i^x &= \frac{2(h_1h_2+h_1h_3+h_2h_3)}{h_4(h_4-h_3)(h_4-h_2)(h_4-h_1)}\,.
\end{aligned} \tag{114}$$

In Eq. (114) $h_1 = x_{i-2} - x_i$, $h_2 = x_{i-1} - x_i$, $h_3 = x_{i+1} - x_i$ and $h_4 = x_{i+2} - x_i$. The coefficients for Eq. (113) are the same except that x is replaced with y. Substituting Eqs. (112) and (113) into Eq. (1) leads to the fourth-order accurate non-compact FD stencil for a regular grid point located at (i, j),

$$a_i^x\,u_{i-2,j} + b_i^x\,u_{i-1,j} + (c_i^x + c_j^y - \beta)\,u_{i,j} + d_i^x\,u_{i+1,j} + e_i^x\,u_{i+2,j} + a_j^y\,u_{i,j-2} + b_j^y\,u_{i,j-1} + d_j^y\,u_{i,j+1} + e_j^y\,u_{i,j+2} = \rho_{i,j}\,. \tag{115}$$

For an irregular grid point as illustrated in Fig. 26 for example, the second derivative approximations need to be modified and adjusted to obtain a solution that is sharp across the immersed boundary while keeping the fourth-order accuracy. Along the line $y = y_j$, $u_{xx_{i,j}}$ is computed according to

$$u_{xx_{i,j}} = \overline{c_i^x}\,u_{i,j} + \overline{d_i^x}\,u_{i+1,j} + \overline{e_i^x}\,u_{i+2,j} + \phi_1\,u_{i+3,j} + \phi_2\,u_{i+4,j} + \phi_{b1}\,u_{IBI_1}\,, \tag{116}$$

where u_{IBI_1} is the known function value at the immersed boundary intersection location 1, IBI_1. In Eq. (116), two additional points are used to keep the fourth-order accuracy (see Fig. 26). Taylor series expansions about $u_{i,j}$ are used to find the coefficients by solving the following system of equations:

$$\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1\\
h_b & 0 & h_1 & h_2 & h_3 & h_4\\
h_b^2 & 0 & h_1^2 & h_2^2 & h_3^2 & h_4^2\\
h_b^3 & 0 & h_1^3 & h_2^3 & h_3^3 & h_4^3\\
h_b^4 & 0 & h_1^4 & h_2^4 & h_3^4 & h_4^4\\
h_b^5 & 0 & h_1^5 & h_2^5 & h_3^5 & h_4^5
\end{bmatrix}
\begin{bmatrix}
\phi_{b1}\\ \overline{c_i^x}\\ \overline{d_i^x}\\ \overline{e_i^x}\\ \phi_1\\ \phi_2
\end{bmatrix}
=
\begin{bmatrix}
1\\0\\2\\0\\0\\0
\end{bmatrix}, \tag{117}$$

where $h_1 = x_{i+1} - x_i$, $h_2 = x_{i+2} - x_i$, $h_3 = x_{i+3} - x_i$, $h_4 = x_{i+4} - x_i$ and $h_b = x_{IBI_1} - x_i$. In the y-direction, a biased finite difference scheme is employed to approximate $u_{yy_{i,j}}$,

$$u_{yy_{i,j}} = \overline{b_j^y}\, u_{i,j-1} + \overline{c_j^y}\, u_{i,j} + \overline{d_j^y}\, u_{i,j+1} + \overline{e_j^y}\, u_{i,j+2} + \psi_1\, u_{i,j+3} + \psi_2\, u_{i,j+4}\,. \tag{118}$$

The coefficients in Eq. (118) can be found from solving Eq. (117) where $h_1 = y_{j+1} - y_j$, $h_2 = y_{j+2} - y_j$, $h_3 = y_{j+3} - y_j$, $h_4 = y_{j+4} - y_j$ and $h_b = y_{j-1} - y_j$. Plugging Eqs. (116) and (118) into Eq. (1) leads to the fourth-order standard FD stencil for the irregular grid point at (i, j),

$$(\overline{c_i^x} + \overline{c_j^y} - \beta)\, u_{i,j} + \overline{d_i^x}\, u_{i+1,j} + \overline{e_i^x}\, u_{i+2,j} + \overline{b_j^y}\, u_{i,j-1} + \overline{d_j^y}\, u_{i,j+1} + \overline{e_j^y}\, u_{i,j+2} + \mathcal{C}_{i,j} = \rho_{i,j} - \phi_{b1}\, u_{IBI_1}\,, \tag{119}$$

where the correction term is given by

$$\mathcal{C}_{i,j} = \phi_1\, u_{i+3,j} + \phi_2\, u_{i+4,j} + \psi_1\, u_{i,j+3} + \psi_2\, u_{i,j+4}\,. \tag{120}$$

References

[1] S. Marella, S. Krishnan, H. Liu, H. Udaykumar, Sharp interface Cartesian grid method I: an easily implemented technique for 3D moving boundary computations, J. Comput. Phys. 210 (1) (2005) 1–31.
[2] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (1) (2005) 239–261.
[3] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (2) (1972) 252–271.
[4] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (3) (1977) 220–252.
[5] M.-C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2) (2000) 705–719.
[6] S.-W. Su, M.-C. Lai, C.-A. Lin, An immersed boundary technique for simulating complex flows with rigid boundary, Comput. Fluids 36 (2) (2007) 313–324.
[7] Y.-H. Tseng, J.H. Ferziger, A ghost-cell immersed boundary method for flow in complex geometry, J. Comput. Phys. 192 (2) (2003) 593–623.
[8] R. Mittal, H. Dong, M. Bozkurttas, F. Najjar, A. Vargas, A. von Loebbecke, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, J. Comput. Phys. 227 (10) (2008) 4825–4852.
[9] H. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface Cartesian grid method for simulating flows with complex moving boundaries, J. Comput. Phys. 174 (1) (2001) 345–380.
[10] H.S. Udaykumar, R. Mittal, P. Rampunggoon, Interface tracking finite volume method for complex solid–fluid interactions on fixed meshes, Commun. Numer. Methods Eng. 18 (2) (2002) 89–97.
[11] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (4) (1994) 1019–1044.
[12] A. Wiegmann, K.P. Bube, The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions, SIAM J. Numer. Anal. 37 (3) (2000) 827–862.
[13] M.N. Linnick, H.F. Fasel, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, J. Comput. Phys. 204 (1) (2005) 157–192.
[14] S. Hosseinverdi, H.F. Fasel, Very high-order accurate sharp immersed interface method: application to direct numerical simulations of incompressible flows, in: 23rd AIAA Computational Fluid Dynamics Conference, 2017, AIAA 2017-3624.
[15] M.N. Linnick, A High-Order Immersed Boundary Method for Unsteady Incompressible Flow Calculations, Ph.D. Thesis, The University of Arizona, 2003.
[16] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem, J. Comput. Phys. 202 (2) (2005) 577–601.
[17] Y. Zhou, S. Zhao, M. Feig, G. Wei, High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources, J. Comput. Phys. 213 (1) (2006) 1–30.
[18] S. Yu, Y. Zhou, G. Wei, Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces, J. Comput. Phys. 224 (2) (2007) 729–756.
[19] X. Zhong, A new high-order immersed interface method for solving elliptic equations with imbedded interface of discontinuity, J. Comput. Phys. 225 (1) (2007) 1066–1099.
[20] Z. Li, K. Ito, The Immersed Interface Method, Society for Industrial and Applied Mathematics, 2006.
[21] Z. Jomaa, C. Macaskill, The Shortley–Weller embedded finite-difference method for the 3D Poisson equation with mixed boundary conditions, J. Comput. Phys. 229 (10) (2010) 3675–3690.
[22] T. Guillet, R. Teyssier, A simple multigrid scheme for solving the Poisson equation with arbitrary domain boundaries, J. Comput. Phys. 230 (12) (2011) 4756–4771.
[23] H.L. Meitz, H.F. Fasel, A compact-difference scheme for the Navier–Stokes equations in vorticity–velocity formulation, J. Comput. Phys. (ISSN 0021-9991) 157 (1) (2000) 371–403.
[24] C. Davies, P.W. Carpenter, A novel velocity–vorticity formulation of the Navier–Stokes equations with applications to boundary layer disturbance evolution, J. Comput. Phys. (ISSN 0021-9991) 172 (1) (2001) 119–165.
[25] G. Sutmann, B. Steffen, High-order compact solvers for the three-dimensional Poisson equation, J. Comput. Appl. Math. (ISSN 0377-0427) 187 (2) (2006) 142–170.
[26] G. Sutmann, Compact finite difference schemes of sixth order for the Helmholtz equation, J. Comput. Appl. Math. (ISSN 0377-0427) 203 (1) (2007) 15–31.
[27] W.L. Briggs, V.E. Henson, S.F. McCormick, A Multigrid Tutorial, second edition, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, ISBN 0-89871-462-1, 2000.
[28] Y. Wang, J. Zhang, Sixth order compact scheme combined with multigrid method and extrapolation technique for 2D Poisson equation, J. Comput. Phys. 228 (1) (2009) 137–146.
[29] G.E. Schneider, M. Zedan, A modified strongly implicit procedure for the numerical solution of field problems, Numer. Heat Transf. 4 (1) (1981) 1–19.
[30] Y. Ge, F. Cao, Multigrid method based on the transformation-free HOC scheme on nonuniform grids for 2D convection diffusion problems, J. Comput. Phys. (ISSN 0021-9991) 230 (10) (2011) 4051–4070.
[31] I.H. Abbott, A.E. Von Doenhoff, Theory of Wing Sections, Including a Summary of Airfoil Data, Courier Corporation, 1959.
[32] N.H. Decker, V.K. Naik, M. Nicoules, Parallelization of Implicit Finite Difference Schemes in Computational Fluid Dynamics, Tech. Rep., ICASE Report No. 90-53, 1990.
[33] J. Reeve, A.D. Scurr, J.H. Merlin, Parallel versions of Stone's strongly implicit algorithm, Concurr. Comput. 13 (12) (2001) 1049–1062.
[34] F. Dierich, K. Wittig, A. Richter, P. Nikrityuk, Parallelization of the 3D SIP Algorithm, AIP Conference Proceedings, vol. 1648, AIP Publishing, 2015, p. 030035.