Available online at www.sciencedirect.com

ScienceDirect

Journal of Computational Physics 225 (2007) 1066–1099

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

A new high-order immersed interface method for solving elliptic equations with imbedded interface of discontinuity

Xiaolin Zhong

Mechanical and Aerospace Engineering Department, University of California, Los Angeles, CA, United States

Received 28 September 2005; received in revised form 1 December 2006; accepted 17 January 2007
Available online 30 January 2007

Abstract

This paper presents a new high-order immersed interface method for elliptic equations with imbedded interface of discontinuity. Compared with the original second-order immersed interface method of [R.J. LeVeque, Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM J. Numer. Anal. 31 (1994) 1001–25], the new method achieves arbitrarily high-order accuracy for derivatives at an irregular grid point by imposing only two physical jump conditions together with a wider set of grid stencils. The new interface difference formulas are expressed in a general explicit form so that they can be applied to different multi-dimensional problems without any modification. The new interface algorithms of up to $O(h^4)$ accuracy have been derived and tested on several one and two-dimensional elliptic equations with imbedded interface. Compared to the standard second-order immersed interface method, the test results show that the new fourth-order immersed interface method leads to a significant improvement in accuracy of the numerical solutions. The proposed method has potential advantages in the application to two-phase flow because of its high-order accuracy and simplicity in applications.
© 2007 Elsevier Inc. All rights reserved.

Keywords: Immersed interface methods; High-order finite difference methods; Two-phase flow simulation methods

1. Introduction

Recently, there has been strong interest in developing numerical methods for computing multi-phase flow with unsteady interface. These methods have many practical applications, such as the simulation of the dynamics of gas bubbles in a liquid [11], drop deformation and breakup in viscous flow [44], free surface flow [51,41], and the breakup of a liquid jet emanating into another fluid [30].

Compared with single-phase numerical methods, algorithms for two-phase flow simulation face additional difficulties related to the interface treatment. Firstly, the shape of the interface can be complex, and can undergo change, merge and breakup during the course of the simulation. Consequently, it is difficult to use body-fitted unsteady grid to fit the evolving interface. A fixed Cartesian grid, where the

E-mail address: xiaolin@seas.ucla.edu

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.01.017

interface can cut through the grid lines, is often used. In a fixed grid, the interface can be treated by, among others, the volume-of-fluid method, the front tracking method [50,14], the level set method [42,43,35,36], and the boundary element method [39]. Each of these methods has its own advantages and disadvantages. The volume-of-fluid method is simple and robust. It can maintain a conservation of bubble or droplet volumes. But it is relatively inaccurate in tracking the interface. The front tracking method can track the interface with relatively high accuracy. But it is difficult to use the method to model the connectivity of the interface undergoing complex changes. The level set method, on the other hand, can easily handle the connectivity of complex interface by using a level set function to track the location and movement of the interface.

Secondly, flow variables and their derivatives can be discontinuous across the interface. Specific jump conditions at the interface depend on the physical property of the problems, the unsteadiness of the interface, and the geometric characteristics of the interface. Consequently, special treatments are necessary for computing flow equations at grid points adjacent to the interface (i.e. irregular points). One of the popular methods in treating interface discontinuity is the immersed boundary method (IBM) originally developed by Peskin (see review by Peskin [37]) for simulating blood flow in the heart. The basic idea of the immersed boundary method is to model the interface by adding a delta-function source term to the Navier–Stokes equations. The resulting equations are then discretized by a standard finite difference method in a fixed Cartesian (or non-Cartesian) grid. The singular delta function is regularized by an approximate smooth function spanning a few grid cells. The immersed boundary method has been incorporated in the front tracking method [50] and the level set method [45,46,6] in the interface treatment.

The immersed boundary method, however, is only first order accurate in computing two-phase flow with discontinuous solutions across the interface, even though higher-order approximation to the delta function can be achieved for problems with smooth solutions [4,22,49,15,9]. Beyer and LeVeque [4] studied the approximation to the delta function by a smooth function for the one-dimensional heat equation. The accuracy was measured by a discrete moment condition. They found that it is possible to achieve second-order accuracy by carefully choosing the discrete representation of the delta function. For some problems, however, it is necessary to add a correction term to the approximation to the delta function in order to maintain second-order accuracy. Griffith and Peskin [15] showed that higher-order convergence rates can only be achieved for sufficiently smooth problems. Tornberg and Engquist [49] studied the numerical approximations of singular source terms in differential equations. Specifically, regularization methods for the delta function were analyzed. They showed that fourth-order convergence can be achieved away from the singularity, when a fourth-order difference formula of the ordinary differential operator is coupled to a regularization function with moment order 4. In general, any delta function regularization produces $O(h)$ errors in the neighborhood of the singularity. Consequently, the interface is "smeared" in a numerical solution computed by the immersed boundary method [31].

An alternative to the immersed boundary method is the "sharp-interface" methods which achieve uniformly second (or higher) order accuracy by incorporating the jump conditions into the finite difference formulas. The immersed interface method (IIM) introduced by LeVeque and Li [24] is one of these methods. A similar idea was used earlier by Mayo [34] for the fast solution of the Poisson's and the biharmonic equations. In presenting their original IIM method, LeVeque and Li [24] considered finite difference methods for the following elliptic equation:

$$\nabla \cdot (\beta(x)\nabla u(x)) + \kappa(x)u(x) = f(x) \tag{1}$$

The equation is defined in a simple region with a uniform Cartesian grid. Fig. 1 shows a schematic of a two-dimensional grid. There is an irregular surface Γ, which may cut across the grid lines, in the computational domain. Across the interface, β, κ, and f may be discontinuous, and along it f may have a delta function singularity. In the derivation of finite difference formulas, the computational grid points are classified into two categories depending on their relative locations with respect to Γ: regular points away from Γ and irregular points adjacent to Γ. A globally $O(h^2)$ accuracy is achieved by using the conventional $O(h^2)$ central scheme for the regular points and a locally $O(h)$ scheme for the irregular points. In the one-dimensional case, a finite difference formula of $O(h)$ accuracy at an irregular point uses a three-point grid stencil together with an additional correction term. A Taylor series expansion at the interface is used to obtain a set of linear equations for

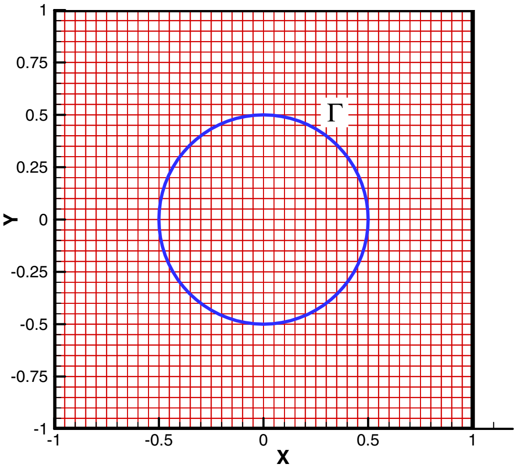


Fig. 1. A schematic of two-dimensional uniform grid with an immersed interface Γ with discontinuity in solutions.

the undetermined coefficients and the correction term. The linear equations are often problem dependent, and they need to be solved numerically every time they are used in the simulation. In order to reach a locally $O(h)$ approximation, the correction term requires jump conditions of up to the second derivatives, i.e.

$$[u], \quad [\beta u_x], \quad \text{and} \quad [\beta u_{xx}] \tag{2}$$

where [ ] denotes the jump in variables across the interface.

Since the publication of the original immersed interface method [24], there have been many further developments and analysis in various aspects of the immersed interface methods [26,27,55,56,18,12,28,7,1,8,2]. Among these developments, Wiegmann and Bube [55,56] developed an explicit-jump immersed interface method for the special cases where the explicit jump conditions of variables and derivatives ($[u]$, $[u_x]$, $[u_{xx}]$, etc.) are known. Though in a simpler explicit form, the explicit-jump immersed interface method is not applicable to the general jump conditions given by (2).

The immersed interface methods have been applied to the Stokes flow with elastic boundaries or surface tension [25], Hele–Shaw flow [17], incompressible flow based on the Navier–Stokes equations with singular source terms [29,5,23], and nonlinear problems in magneto-rheological fluids [19]. Despite these applications, the immersed interface methods are often difficult to apply to complex two or three-dimensional two-phase flow problems. In order to maintain a second-order accuracy, it is necessary to obtain jump conditions at the interface for flow variables and their first and second derivatives. For the Navier–Stokes equations with an interface of discontinuity, it is easy to derive the physical jump conditions for flow variables and their first derivatives across the interface. But it is difficult to obtain jump conditions for the second or higher-order derivatives. In order to develop third or higher-order immersed interface methods, it is necessary to obtain jump conditions for the third and higher derivatives. In addition, the finite difference formulas of the original immersed interface method need to be re-derived for different problems. The coefficients and the correction terms in the finite difference formulas at irregular points cannot be obtained explicitly. They are often computed numerically by solving a matrix equation. The repeated computations for the coefficients and correction terms can be computationally expensive.

Linnick and Fasel [31] presented a high-order immersed interface method for simulating unsteady incompressible flow in an irregular domain. Their method is an extension of the explicit-jump immersed interface method of Wiegmann and Bube [56]. Instead of using analytical jump conditions, they compute the jump conditions for higher derivatives numerically. A fourth-order compact scheme was successfully tested for computing incompressible flow over a cylinder. This method, however, is not applicable to two-phase flow with moving interface with general jump conditions, such as those of Eq. (2). Piraux and Lombard [38,32] presented another sharp-interface method for numerical computations of interface for wave equations. In order to discretize derivatives at irregular points, a set of modified variables across the interface are computed by using the original variables on both sides of the interface and a set of jump conditions for variables and derivatives. This method also requires the knowledge of jump conditions of high-order derivatives.

Another "sharp-interface" method is the ghost fluid method of Fedkiw et al. [10,21,13]. The basic idea is to extrapolate variables on one side of the interface into the "ghost cells" on the other side. Gibou and Fedkiw [13] introduced an $O(h^4)$ accurate finite difference discretization for the Laplace and heat equations on irregular domain. However, the ghost fluid method is only first order accurate for two-phase flow simulation. Helenbrook et al. [16] presented a second order interface method with ghost cells for incompressible flow with surface discontinuity. The limitation of the method is that it is most suitable for inviscid flow only. It should also be mentioned that sharp interface Cartesian grid methods have been developed for flow with moving solid bodies [54,53,52,3,20,33,40]. These methods can be up to second order accurate. The treatment of irregular points near the solid-fluid interface is mainly based on local polynomial extrapolations.

Based on the brief review above, it is desirable for a high-order immersed interface method to have the following properties:

1. Only two physical jump conditions of the variables and their first derivatives should be needed in the second or higher-order immersed interface methods.
2. Finite difference formulas at irregular points should be expressed in a general explicit form (without the need to compute matrix equations repeatedly) so that they can be applied to different problems without any modification.

To reach these goals, this paper presents a new set of high-order immersed interface methods. They can be arbitrarily high-order accurate, and require only jump conditions for variables and their first derivatives. The new methods also have the advantage that the finite difference formulas at irregular points are derived in a general explicit form. Compared with the original IIM method, one of the disadvantages of the new methods is that they lead to wider grid stencils. They also do not recover the original finite difference expressions in regular grid points when there is no jump at the interface. In the case of no discontinuity at the interface, however, the current methods are equivalent to a local high-order spline approximation at the interface. While the main purpose of the current approach is to develop higher-order interface treatments, at second order approximation, the current method provides an alternative approach to original second-order immersed interface method.

The derivation, analysis, and test results of the new methods are presented in following sections. Though the motivation of the present work is to apply the methods to multi-phase flow simulation [47,48], the new high-order immersed interface methods are presented in this paper for elliptic equations in the form of Eq. (1) with imbedded interface of discontinuity only. Nevertheless, the method has potential advantages in the application to two-phase flow because of its high-order accuracy and simplicity in applications by requiring only the physical jump conditions for variables and their first derivatives are needed in the finite difference formulas. The derivation of jump conditions involving the second or higher-order derivatives can be difficult for two-phase flow problem involving the Navier–Stokes equations.

During the review of this paper, one of the reviewers pointed out a paper by Zhou et al. [57] on a new high-order matched interface and boundary (MIB) method for solving elliptic equations with discontinuous coefficients and singular sources on Cartesian grids. The paper was in press by the Journal of Computational Physics at the time of making the revision of this paper. The MIB method is based on the use of fictitious points to achieve high-order accuracy. To construct higher-order interface schemes, the MIB method bypasses the major challenge of implementing high-order jump conditions by repeatedly enforcing the lowest order jump conditions. In treating straight, regular interfaces, MIB schemes up to 16th-order were constructed. For more general elliptic problems with curved, irregular interfaces and boundary, up to 6th-order MIB schemes were demonstrated. The approach presented in the current paper is similar to Zhou et al.'s approach in that both are high-order methods using lower order jump conditions only. The two methods are different in that Zhou's methods employ fictitious points to achieve high-order accuracy, which is similar to the idea of Ghost Fluid Methods. In addition, the MIB formulas are not explicitly derived. Instead, they are computed by a computer program. The current methods, on the other hand, do not use fictitious points and express the finite difference formulas at irregular points in a generally applicable and explicit form.

2. Explicit finite difference formulas at irregular grid points

The new high-order immersed interface method is presented for one-dimensional differential elliptic equations in the form of Eq. (1) in this section. The method is extended to two-dimensional elliptic equations afterward. For simplicity, only the finite difference approximation to $(\mathrm{d}u/\mathrm{d}x)_i$ and $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ is presented, though formulas for higher derivatives can be easily obtained by the same method. A uniform grid of mesh size h shown in Fig. 2 is used for the discretization. Without losing generality, it is assumed that the origin of the coordinate system is located at grid point i, i.e.

$$x_{i+k} = kh \quad (k = 0, \pm 1, \pm 2, \ldots) \tag{3}$$

The interface is located at:

$$x_\Gamma = x_i + \sigma h = \sigma h \tag{4}$$

where σ is the interface location parameter which satisfies

$$0 \leqslant \sigma \leqslant 1 \tag{5}$$

As discussed in the preceding section, only two jump conditions involving u and u_x are used in the finite difference approximation of the derivatives. A general jump conditions across the interface can be written as:

$$[\alpha u] = \alpha^+ u^+ - \alpha^- u^- = A \tag{6}$$

and

$$[\beta u_x] = \beta^+ u_x^+ - \beta^- u_x^- = B \tag{7}$$

where the superscripts, "+" and "−", represent the variables and constants at the right and left sides of the interface Γ, respectively. The constants, α^+, α^-, β^+, β^-, A, and B, are known constants determined by the nature of the equation being computed. In an actual two-phase flow problem, the jump conditions can be formulated such that α^+, α^-, β^+, β^- are dimensionless constants.

A grid point is called a regular point if the finite difference formulas at this point only involve grid points on the same side of the interface. Otherwise, it is an irregular point. If grid point i is a regular point (without the interface), finite difference approximation of an arbitrary order can be easily derived by a Taylor series expansion or by a polynomial interpolation. For example, the second and fourth order central difference approximations to $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ are:

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} + O(h^2) \tag{8}$$

and

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{-u_{i-2} + 16u_{i-1} - 30u_i + 16u_{i+1} - u_{i+2}}{12h^2} + O(h^4) \tag{9}$$

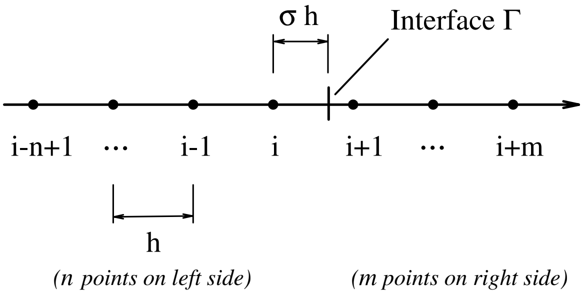


Fig. 2. Uniform grid with an interface located on the right side of irregular point i with a general grid stencil of n and m points on the left and right sides, respectively.

Therefore, in the case of $O(h^2)$ approximation in the regular points, i and $i+1$ (Fig. 2) are irregular points. For the $O(h^4)$ approximation, there are four irregular points, from $i-1$ to $i+2$. Alternatively, we can treat only points i and $i+1$ as irregular points, while fourth-order one-sided approximation is used for points $i-1$ and $i+2$. The latter approach is used in this paper.

For irregular point i shown in Fig. 2, the use of Eq. (8) will lead to large computational errors because of the discontinuity of u at Γ. LeVeque and Li [24] derived a locally $O(h)$ approximation at i by adding a correction term to the three-point stencil. The difference formula and the correction term are then determined by a Taylor expansion at the interface. In order to reach $O(h)$ approximation, it is necessary to know the jump conditions involving u, u_x and u_{xx}. The first two jump conditions can be obtained easily. The jump conditions for the second derivatives can be obtained by taking derivatives of the lower order jump conditions together with the differential equation.

Therefore, the high-order immersed interface method presented below achieves a high-order approximation at the irregular point i by imposing the two jump conditions given by Eqs. (6) and (7) only. Instead of using more and more jump conditions of higher-order derivatives to achieve higher-order accuracy at the irregular point as done in the original IIM method, we use more and more grid points on both sides of the interface so that arbitrary order approximation can be achieved while only the two jump conditions (6) and (7) are imposed. General difference formulas for $(\mathrm{d}u/\mathrm{d}x)_i$ and $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ in explicit form can be derived by a matched polynomial expansion, so that they are problem independent.

Finite difference approximation for $(\mathrm{d}u/\mathrm{d}x)_i$ and $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ at the irregular point i is considered by using a stencil of n points on the left of Γ and m points on the right (Fig. 2). The order of the approximation increases with the increasing values of n and m. In order to have a uniform accuracy, it is desirable to have the same number of points on both sides of the interface, i.e.

$$n = m \tag{10}$$

Since n and m can be different in some special circumstances, we derive the general formulas by assuming arbitrary values of n and m. The finite difference formulas for i can be derived by a Taylor series expansion with σ as a parameter. The case of locally $O(h)$ approximation for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ at interface is considered first as an example below. A general formula for arbitrary values of n and m is derived afterward.

2.1. Difference formulas for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ at irregular points $(O(h), n = m = 2)$

For the case of local $O(h)$ approximation, it can be shown that it is necessary to use two points on both sides of Γ $(n = m = 2)$ when only two jump conditions (6) and (7) are used in the formula. A general finite difference formula, using the four-point stencil and the two jump conditions, can be written in the following form:

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{d_{-1}u_{i-1} + d_0u_i + d_1u_{i+1} + d_2u_{i+2} + d_AA + hd_BB}{h^2} + O(h) \tag{11}$$

where d_ks are determined by Taylor expansions around x_Γ with the requirement that the approximation is $O(h)$. The constants A and B are given by Eqs. (6) and (7). By paying special attention to the fact that u and its derivatives can be discontinuous at Γ, we obtain the Taylor expansions at x_Γ as:

$$u_{i-1} = u_\Gamma^- - (1+\sigma)h\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^- + \frac{1}{2!}(1+\sigma)^2h^2\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^- + \cdots \tag{12}$$

$$u_i = u_\Gamma^- - \sigma h\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^- + \frac{1}{2!}\sigma^2h^2\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^- + \cdots \tag{13}$$

$$u_{i+1} = u_\Gamma^+ + (1-\sigma)h\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^+ + \frac{1}{2!}(1-\sigma)^2h^2\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^+ + \cdots \tag{14}$$

$$u_{i+2} = u_\Gamma^+ + (2-\sigma)h\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^+ + \frac{1}{2!}(2-\sigma)^2h^2\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^+ + \cdots \tag{15}$$

where superscripts "−" and "+" represent the variables on the left and right sides of the interface, respectively. Substituting Eqs. (6) and (7) into Eqs. (14) and (15), we have

$$u_{i+1} = \frac{A}{\alpha^+} + \frac{(1-\sigma)hB}{\beta^+} + c_\alpha u_\Gamma^- + c_\beta(1-\sigma)h\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^- + \frac{1}{2!}(1-\sigma)^2h^2\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^+ + \cdots \tag{16}$$

$$u_{i+2} = \frac{A}{\alpha^+} + \frac{(2-\sigma)hB}{\beta^+} + c_\alpha u_\Gamma^- + c_\beta(2-\sigma)h\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^- + \frac{1}{2!}(2-\sigma)^2h^2\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^+ + \cdots \tag{17}$$

where c_α and c_β are parameters determined by the jump conditions of the physical problems, i.e.,

$$c_\alpha = \alpha^-/\alpha^+ \tag{18}$$

$$c_\beta = \beta^-/\beta^+ \tag{19}$$

Substituting Eqs. (12), (13), (16), and (17) into Eq. (11) leads to

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{1}{h^2}\left(a_1u_\Gamma^- + a_2\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_\Gamma^- h + a_3\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^- h^2 + a_4\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_\Gamma^+ h^2 + a_5A + +a_6Bh + O(h^3)\right) \tag{20}$$

where the coefficients a_i for $(i = 1,\ldots,6)$ can be obtained easily. The conditions for an $O(h)$ approximation of $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ are then obtained by Eq. (20), i.e.,

$$\begin{cases} a_i = 0 \quad (i = 1,\ldots,6 \quad \text{and} \quad i \neq 3) \\ a_3 = 1 \end{cases} \tag{21}$$

Hence the conditions for $O(h)$ approximation of $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ can be written as:

$$\begin{bmatrix} 1 & 1 & c_\alpha & c_\alpha & 0 & 0 \\ -(1+\sigma) & -\sigma & (1-\sigma)c_\beta & (2-\sigma)c_\beta & 0 & 0 \\ (1+\sigma)^2 & \sigma^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & (1-\sigma)^2 & (2-\sigma)^2 & 0 & 0 \\ 0 & 0 & 1 & 1 & \alpha^+ & 0 \\ 0 & 0 & (1-\sigma) & (2-\sigma) & 0 & \beta^+ \end{bmatrix} \begin{bmatrix} d_{-1} \\ d_0 \\ d_1 \\ d_2 \\ d_A \\ d_B \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{22}$$

It is necessary to solve the matrix equation above in order to obtain the values of d_k $(k = -1,\ldots,2)$, d_A, and d_B. For the current case of using two grid points on both sides of the interface, the grid stencil is relatively small and the analytical solution of Eq. (22) can be obtained as:

$$\begin{aligned} d_{-1} &= \frac{1}{D}\{c_\alpha(3\sigma - 2\sigma^2) - c_\beta(-2 + 3\sigma - \sigma^2)\} \\ d_0 &= \frac{1}{D}\{c_\alpha(-3 - \sigma + 2\sigma^2) - c_\beta(2 - 3\sigma + \sigma^2)\} \\ d_1 &= \frac{1}{D}\{4 - 4\sigma + \sigma^2\} \\ d_2 &= \frac{1}{D}\{-1 + 2\sigma - \sigma^2\} \\ d_A &= \frac{1}{\alpha^+D}\{-3 + 2\sigma\} \\ d_B &= -\frac{1}{\beta^+D}\{2 - 3\sigma + \sigma^2\} \end{aligned} \tag{23}$$

where

$$D = \frac{1}{2}\{c_\beta(2 + \sigma - 5\sigma^2 + 2\sigma^3) - c_\alpha(-3\sigma - \sigma^2 + 2\sigma^3)\}$$

and d_ks are functions of σ (and jump parameters: c_α, c_β, α^+, and β^+), i.e.

$$d_k = d_k(\sigma) \quad k = (-1, \ldots, 2, A, B) \tag{24}$$

Eq. (11), together with Eq. (23), is an explicit difference formula for $O(h)$ approximation to $(\mathrm{d}^2u/\mathrm{d}x^2)_i$. The same general formula can be used for different problems as long as the jump conditions are specified in the form of Eqs. (6) and (7). Eq. (23) shows that the current formula at the irregular point does not have singularity even for the special cases of Γ coinciding with grid i ($\sigma = 0$) or $i+1$ ($\sigma = 1$).

For high-order approximations with larger values of n and m, the same procedure can be used to derive the finite difference approximation to the second derivative of $(\mathrm{d}^2u/\mathrm{d}x^2)_i$. For the general case of arbitrary n and m, the Taylor expansion leads to a linear system of $n+m$ equations for the coefficients d_k. The analytical solution for the matrix equation similar to Eq. (22) is difficult to obtain in the general case. The matrix equation can be solved numerically during the simulation after the value of σ is determined at a particular grid point. Such calculations were used in LeVeque and Li's original second-order IIM method. Since σ can change during the course of the calculations, repeatedly solving the matrix equation for d_k can be computationally expensive (if n and m are large). Therefore, it is desirable to have explicit difference formulas for d_k, similar to those of Eq. (23), for any order of approximation, instead of a matrix equation. These general formulas are derived by a matched polynomial interpolation in next section.

2.2. *Difference formulas at irregular point with a general $n+m$ grid stencil*

The general case of finite difference formulas at irregular point i for arbitrary stencil of n and m points are considered in this section. The finite difference approximation at irregular grid point i (Fig. 2) can be derived either by a Taylor series expansion or by a matched polynomial interpolation. Both methods lead to the same results. Since an explicit formula for a general finite difference approximation for arbitrary values of σ, n, and m can be obtained by a matched polynomial interpolation, it is used for the derivation here.

Because of the discontinuity at Γ, it is necessary to use two separate polynomials to interpolate through grid points on both sides of the interface. The two polynomials satisfy the two jump conditions of Eqs. (6) and (7). The two polynomials are dependent on each other because of the constraint imposed by the jump conditions. The appropriate polynomials on the either side of Γ will be one degree higher than the Lagrange polynomial supported by the one-sided grid stencil, with an arbitrary undetermined coefficient. The two unknown coefficients are subsequently determined by the jump conditions. It can be shown that the polynomial on the left side of Γ, interpolating through n grid points (Fig. 2), can be written as

$$P^-(x) = \sum_{k=0}^{-n+1} l_k(x) u_{i+k} + a_n R(x) \tag{25}$$

where a_n is an undetermined coefficient to be decided by the jump conditions, and

$$R(x) = \prod_{k=0}^{-n+1} (x - x_{i+k}) \tag{26}$$

$l_k(x)$ is the Lagrange polynomial interpolating through the n grid points on the left hand side of Γ, i.e.

$$l_k(x) = \prod_{l=0,l\neq k}^{-n+1} (x - x_{i+l}) \Bigg/ \prod_{l=0,l\neq k}^{-n+1} (x_{i+k} - x_{i+l}) \tag{27}$$

Similarly, the polynomial on the right hand side of Γ is:

$$P^+(x) = \sum_{k=1}^{m} h_k(x) u_{i+k} + b_m Q(x) \tag{28}$$

where b_m is an undetermined coefficient, and

$$Q(x) = \prod_{k=1}^{m} (x - x_{i+k}) \tag{29}$$

$h_k(x)$ is the Lagrange polynomial through the m grid points on the right hand side, i.e.

$$h_k(x) = \prod_{l=1,l\neq k}^{m} (x - x_{i+l}) \bigg/ \prod_{l=1,l\neq k}^{m} (x_{i+k} - x_{i+l}) \tag{30}$$

The two undetermined coefficients, a_n and b_m, are determined by Eqs. (6) and (7). Substituting Eqs. (25) and (28) into Eq. (6):

$$\alpha^+\left\{\sum_{k=1}^{m} h_k(x_\Gamma)u_{i+k} + b_m Q(x_\Gamma)\right\} - \alpha^-\left\{\sum_{k=0}^{-n+1} l_k(x_\Gamma)u_{i+k} + a_n R(x_\Gamma)\right\} = A \tag{31}$$

Rearrange the equation above:

$$c_{11}a_n + c_{12}b_m = \beta_1 \tag{32}$$

where

$$\begin{aligned} c_{11} &= -\alpha^- R(x_\Gamma) \\ c_{12} &= \alpha^+ Q(x_\Gamma) \\ \beta_1 &= A - \alpha^+ \sum_{k=1}^{m} h_k(x_\Gamma)u_{i+k} + \alpha^- \sum_{k=0}^{-n+1} l_k(x_\Gamma)u_{i+k} \end{aligned} \tag{33}$$

Similarly, the second jump condition is imposed by substituting Eqs. (25) and (28) into Eq. (7), i.e.

$$\beta^+\left\{\sum_{k=1}^{m} h'_k(x_\Gamma)u_{i+k} + b_m Q'(x_\Gamma)\right\} - \beta^-\left\{\sum_{k=0}^{-n+1} l'_k(x_\Gamma)u_{i+k} + a_n R'(x_\Gamma)\right\} = B \tag{34}$$

or

$$c_{21}a_n + c_{22}b_m = \beta_2 \tag{35}$$

where

$$\begin{aligned} c_{21} &= -\beta^- R'(x_\Gamma) \\ c_{22} &= \beta^+ Q'(x_\Gamma) \\ \beta_2 &= B - \beta^+ \sum_{k=1}^{m} h'_k(x_\Gamma)u_{i+k} + \beta^- \sum_{k=0}^{-n+1} l'_k(x_\Gamma)u_{i+k} \end{aligned} \tag{36}$$

Solving Eqs. (32) and (35) results in the following general formulas for a_n (the value of b_m is not needed for the approximation of $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ at i, which is located on the left side of Γ):

$$a_n = \sum_{k=-n+1}^{m} \gamma_k u_{i+k} + \xi_A^- A + \xi_B^- B \tag{37}$$

where

$$\begin{aligned} \gamma_k &= \begin{cases} \frac{1}{J}\{\alpha^-\beta^+ Q'(x_\Gamma)l_k(x_\Gamma) - \alpha^+\beta^- Q(x_\Gamma)l'_k(x_\Gamma)\} & (k = -n+1\ldots 0) \\ \frac{\alpha^+\beta^+}{J}\{-Q'(x_\Gamma)h_k(x_\Gamma) + Q(x_\Gamma)h'_k(x_\Gamma)\} & (k = 1,\ldots,m) \end{cases} \\ \xi_A^- &= \frac{1}{J}\{\beta^+ Q'(x_\Gamma)\} \\ \xi_B^- &= -\frac{1}{J}\{\alpha^+ Q(x_\Gamma)\} \\ J &= \alpha^+\beta^- R'(x_\Gamma)Q(x_\Gamma) - \alpha^-\beta^+ Q'(x_\Gamma)R(x_\Gamma) \end{aligned} \tag{38}$$

Based on Eq. (25) and a_n given by Eq. (37), finite difference approximation of derivatives of any order at an irregular grid point located on the left side of Γ can be derived by evaluating the derivatives of $P^-(x)$ at the corresponding location. Hence, the lth derivative at point i can be approximated by

$$\left(\frac{\mathrm{d}^l u}{\mathrm{d}x^l}\right)_i \approx \left(\frac{\mathrm{d}^l P^-(x)}{\mathrm{d}x^l}\right)_{x=x_i} \tag{39}$$

where l is an arbitrary positive integer. By substituting Eqs. (25) and (37) in to Eq. (39), the general finite difference formula is:

$$\left(\frac{\mathrm{d}^l u}{\mathrm{d}x^l}\right)_i \approx \sum_{k=-n+1}^{m} d_k u_{i+k} + d_A A + d_B B \tag{40}$$

where

$$\begin{aligned} d_k &= \begin{cases} l_k^{(l)}(x_i) + \gamma_k R^{(l)}(x_i) & (k = -n+1 \ldots 0) \\ \gamma_k R^{(l)}(x_i) & (k = 1, \ldots, m) \end{cases} \\ d_A &= \xi_A^- R^{(l)}(x_i) \\ d_B &= \xi_B^- R^{(l)}(x_i) \end{aligned} \tag{41}$$

Eq. (40) can be used to obtain finite difference approximation for $(\mathrm{d}u/\mathrm{d}x)_i$ and $(\mathrm{d}^2u/\mathrm{d}x^2)_i$, as well as any other derivatives, for the case of Γ located on the right side of irregular point i. Formulas for the case of interface located on the left side of the irregular grid point can be derived from Eq. (40) by a coordinate transformation described later in the section.

The specific difference formulas of various orders are derived in this paper by a computer program based on the general formulas listed above. The order of accuracy increases when the grid stencil surrounding the irregular point becomes larger. In order to maintain a uniform accuracy, the same numbers ($n = m$) of grid points are used on both sides of Γ. The most useful formulas are presented below and in the Appendix of this paper.

2.3. Difference formulas at irregular points with four-point stencil ($n = m = 2$)

2.3.1. Interface located on the right side of irregular grid point

This case is shown in Fig. 2, where the interface is located on the right side of i with a given value of σ. A four-point grid stencil, two on each side of the interface, is used for the finite difference approximation of derivatives at i. The general formula for the second derivative was derived by Taylor expansions in Eqs. (11) and (23). In addition to the same results for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$, the matched polynomial interpolation formula (40) can provide formulas for first and higher derivatives. The general formulas are:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = \frac{d_{-1}u_{i-1} + d_0 u_i + d_1 u_{i+1} + d_2 u_{i+2} + d_A A + h d_B B}{2h} + \frac{u_i - u_{i-1}}{h} + O(h^2) \tag{42}$$

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{d_{-1}u_{i-1} + d_0 u_i + d_1 u_{i+1} + d_2 u_{i+2} + d_A A + h d_B B}{h^2} + O(h) \tag{43}$$

where d_ks are the same as those derived by the Taylor expansion in Eq. (23).

2.3.2. Interface located on the left side of irregular grid point

This case is demonstrated in Fig. 3, where Γ is specified by σ. Finite difference formulas for this case can be obtained from Eqs. (42) and (43) by a coordinate transformation, i.e.

$$x' = -x \tag{44}$$

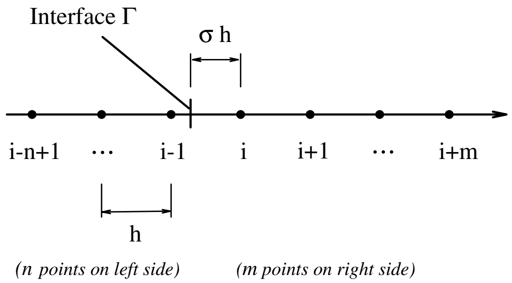


Fig. 3. Uniform grid with an interface located on the left side of irregular point *i*.

It can be shown that the finite difference formulas at *i* for the current case are:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = -\frac{d_{-1}u_{i+1} + d_0u_i + d_1u_{i-1} + d_2u_{i-2} - d_AA + hd_BB}{2h} + \frac{u_{i+1} - u_i}{h} + O(h^2) \tag{45}$$

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{d_{-1}u_{i+1} + d_0u_i + d_1u_{i-1} + d_2u_{i-2} - d_AA + hd_BB}{h^2} + O(h) \tag{46}$$

where d_ks are given by Eq. (23).

The difference formulas above are general in the sense that they can be used without modification for different problems as long as the jump conditions are given in the form of (6) and (7). For the special case that there is no discontinuity at the interface, corresponding to:

$$[\alpha] = 0, \quad [\beta] = 0, \quad A = 0, \quad B = 0$$

Eqs. (42) and (43) reduce to a local piecewise polynomial approximation with continuous zero and first derivatives at the interface. As stated before, in the limiting case of Γ coinciding with the two grid points i ($\sigma = 0$) and $i + 1$ ($\sigma = 1$), Eqs. (42) and (43) are not singular. In other words, the coefficients given by Eq. (23) are well defined in $\sigma \in [0, 1]$, including the two end points.

The four-grid stencil ($n = m = 2$) leads to a piecewise second-degree polynomial approximation at the irregular point. This results in a $O(h)$ approximation for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ and a $O(h^2)$ approximation for $(\mathrm{d}u/\mathrm{d}x)_i$. This approximation has the same order of accuracy as LeVeque and Li's [24] locally first-order (but globally second-order) immersed interface method.

The difference formulas at irregular points with 6–10-point stencils ($n = m = 3$, $n = m = 4$ and $n = m = 5$) have also been computed, and they are given in the Appendix.

2.4. Difference formulas at irregular points with two-point stencil ($n = m = 1$)

Finite difference formula can be derived for the special case of using one grid point on both sides of the interface at the irregular grid point *i* shown in Fig. 2. Second derivative cannot be obtained in this case because the stencil is not wide enough. For the case shown in Fig. 2:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = \frac{d_0u_i + d_1u_{i+1} + d_AA + hd_BB}{h} + O(h) \tag{47}$$

where

$$\begin{aligned} d_0 &= \frac{1}{D}\{c_\alpha\} \\ d_1 &= \frac{-1}{D} \\ d_A &= \frac{1}{\alpha^+D} \\ d_B &= -\frac{1}{\beta^+D}\{-1 + \sigma\} \end{aligned} \tag{48}$$

where

$$D = \{c_\beta(-1+\sigma) - c_\alpha(\sigma)\}$$

In the special case of $\alpha^\pm = \beta^\pm = 1$, the jump conditions (6) and (7) reduce to: $[u] = A$ and $[u_x] = B$. In this case, Eq. (47) can be written as:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = \frac{u_{i+1} - u_i - J}{h} + O(h) \tag{49}$$

where the correction term is:

$$J = A + (1-\sigma)hB = [u]_\Gamma + (1-\sigma)h[u_x]_\Gamma \tag{50}$$

This correction term is the same as the correction term when a linear extrapolation used in the ghost fluid method to approximate the first derivative at the interface.

2.5. Special case of irregular point surrounded by two interfaces

The difference formulas presented in the preceding sections are for an irregular grid point which is affected by one interface either on its left side or on its right side. In actual computations, however, special cases can arise where these general formulas do not apply. Fig. 4 shows an example of the special cases when an irregular point is surrounded by two interfaces from both sides. Similar special situations also appear when the interface is very close to the boundary of the computation domain, or when two or more interfaces are very close to a grid point. The difference formulas for irregular grid points need to be modified for the special cases.

In this section, only the special case shown in Fig. 4 is considered. Other special cases can be treated similarly. They are not discussed here. Fig. 5 shows a one-dimensional counter part of Fig. 4 where an irregular point i is surrounded by two interfaces. The locations of the two interfaces are specified by σ_1 ($\sigma_1 \in [0,1]$) and σ_2 ($\sigma_2 \in [0,1]$) as shown in Fig. 5. The grid points are separated into three sections by the two interfaces. We assume a general case of approximating derivatives at point i by using a grid stencil of three groups of grids (K, N, and M) in the three sections separated by the interfaces, i.e.

- K points in the left section ($j = i-K-N+1,\ldots,i-N$);
- N points in the middle section ($j = i-N+1,\ldots,i$);
- M points in the right section ($j = i+1,\ldots,i+M$).

Finite difference formulas at the irregular point i for this case are derived by a matched polynomial interpolation satisfying the following four jump conditions at the two interfaces:

At Γ_1:

$$[\alpha_1 u] = \alpha_1^+ u^+ - \alpha_1^- u^- = A_1 \tag{51}$$

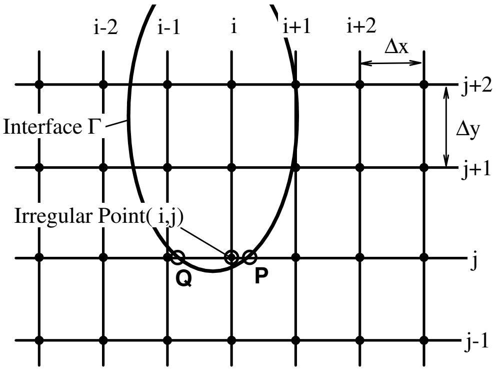


Fig. 4. A special case where the irregular point (i,j) is surrounded by two interfaces from both sides along the horizontal grid line.

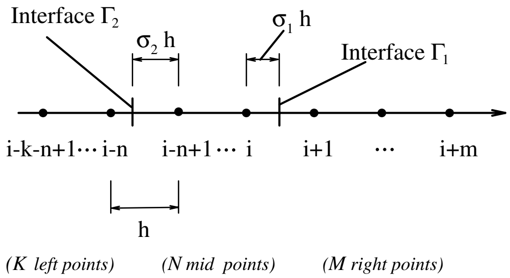


Fig. 5. Uniform grid with two interfaces surrounding an irregular point *i*.

$$[\beta_1 u_x] = \beta_1^+ u_x^+ - \beta_1^- u_x^+ = B_1 \tag{52}$$

At Γ_2:

$$[\alpha_2 u] = \alpha_2^+ u^+ - \alpha_2^- u^- = A_2 \tag{53}$$

$$[\beta_2 u_x] = \beta_2^+ u_x^+ - \beta_2^- u_x^+ = B_2 \tag{54}$$

Because of the discontinuity at the interfaces, it is necessary to use three polynomials for the three groups of grid points. Similar to the derivation of Section 2.2, it can be shown that the polynomials for the three sections can be written as follows.

2.5.1. Middle region with N grid points

The polynomial, which interpolates through N points in the middle section between Γ_1 and Γ_2 ($j = i - N + 1, \ldots, i$) and satisfies the jump conditions from both interfaces, is:

$$P_0(x) = \sum_{k=0}^{-N+1} l_k(x) u_{i+k} + (a_1 + a_2 x) R(x) \tag{55}$$

where $R(x)$ and $l_k(x)$ are given by Eqs. (26) and (27), respectively. Because of the additional jump conditions at the interfaces, there are two undetermined coefficients, a_1 and a_2, in this case.

2.5.2. Right region with M grid points

The polynomial, which interpolates through M points located on the right hand side of interface Γ_1 ($j = i + 1, \ldots, i + M$), is:

$$P_1(x) = \sum_{k=1}^{M} h_k(x) u_{i+k} + a_3 Q(x) \tag{56}$$

where $Q(x)$ and $h_k(x)$ are given by Eqs. (29) and (30). There is one undetermined coefficient a_3 to be decided by the jump conditions.

2.5.3. Left region with K grid points

The polynomial, which interpolates through K points located on the left hand side of interface Γ_2 ($j = i - K - N + 1, \ldots, i - N$), is:

$$P_2(x) = \sum_{k=-N}^{-K-N+1} t_k(x) u_{i+k} + a_4 T(x) \tag{57}$$

where

$$T(x) = \prod_{k=-N}^{-N-K+1} (x - x_{i+k}) \tag{58}$$

and $t_k(x)$ is the Lagrange interpolation polynomial through the K grid points on the left hand side of the interface Γ_2, i.e.,

$$t_k(x) = \prod_{l=-N,l\neq k}^{-N-K+1} (x - x_{i+l}) \Bigg/ \prod_{l=-N,l\neq k}^{-N-K+1} (x_{i+k} - x_{i+l}) \tag{59}$$

There is one undetermined coefficient a_4 to be decided by the jump conditions.

There are four undetermined coefficients, a_l ($l = 1,\ldots,4$), in polynomials (55)–(57). They are decided by the four jump conditions (51)–(54). Substituting Eqs. (55)–(57) into the four jump conditions, we have:

$$\alpha_1^+\left\{\sum_{k=1}^{M} h_k(x_{\Gamma_1})u_{i+k} + a_3Q(x_{\Gamma_1})\right\} - \alpha_1^-\left\{\sum_{k=0}^{-N+1} l_k(x_{\Gamma_1})u_{i+k} + (a_1 + a_2x_{\Gamma_1})R(x_{\Gamma_1})\right\} = A_1 \tag{60}$$

$$\beta_1^+\left\{\sum_{k=1}^{M} h'_k(x_{\Gamma_1})u_{i+k} + a_3Q'(x_{\Gamma_1})\right\} - \beta_1^-\left\{\sum_{k=0}^{-N+1} l'_k(x_{\Gamma_1})u_{i+k} + (a_1 + a_2x_{\Gamma_1})R'(x_{\Gamma_1}) + a_2R(x_{\Gamma_1})\right\} = B_1 \tag{61}$$

$$\alpha_2^+\left\{\sum_{k=0}^{-N+1} l_k(x_{\Gamma_2})u_{i+k} + (a_1 + a_2x_{\Gamma_2})R(x_{\Gamma_2})\right\} - \alpha_2^-\left\{\sum_{k=-N}^{-N-K+1} t_k(x_{\Gamma_2})u_{i+k} + a_4T(x_{\Gamma_2})\right\} = A_2 \tag{62}$$

$$\beta_2^+\left\{\sum_{k=0}^{-N+1} l'_k(x_{\Gamma_2})u_{i+k} + (a_1 + a_2x_{\Gamma_2})R'(x_{\Gamma_2}) + a_2R(x_{\Gamma_2})\right\} - \beta_2^-\left\{\sum_{k=-N}^{-N-K+1} t'_k(x_{\Gamma_2})u_{i+k} + a_4T'(x_{\Gamma_2})\right\} = B_2 \tag{63}$$

Similar to the derivation in Section 2.2, the undetermined coefficients can be obtained explicitly by solving the four linear equations above. Specifically, and arbitrary coefficient, a_l, can be expressed in the following form:

$$a_l = \sum_{k=-K-N+1}^{M} \gamma_k^{(l)}u_{i+k} + \gamma_{A_1}^{(l)}A_1 + \gamma_{B_1}^{(l)}B_1 + \gamma_{A_2}^{(l)}A_2 + \gamma_{B_2}^{(l)}B_2 \quad (l = 1,\ldots,4) \tag{64}$$

where $\gamma_k^{(l)}$s are obtained by solving Eqs. (60)–(63) as explicit functions of σ_1 and σ_2, i.e.

$$\gamma_k^{(l)} = \gamma_k^{(l)}(\sigma_1,\sigma_2) \tag{65}$$

Once these coefficients are determined, finite difference approximation of the first and second derivatives, as well as derivatives of any other order, at any irregular grid point in these three sections can be derived by evaluating the derivatives of the corresponding polynomial in the corresponding section. For irregular point i in the middle section, for example, the second derivative can be approximated as:

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i \approx \left(\frac{\mathrm{d}^2P_0(x)}{\mathrm{d}x^2}\right)_{x=x_i} \tag{66}$$

The equation above leads to a general finite difference in the following form:

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i \approx \frac{1}{h^2}\left\{\sum_{k=-K-N+1}^{M} d_ku_{i+k} + d_{11}A_1 + d_{12}hB_1 + d_{13}A_2 + d_{14}hB_2\right\} \tag{67}$$

where the coefficients can be expressed as explicit rational functions of σ_1 and σ_2. The finite difference formulas for irregular grid points in the left and right sections in Fig. 5 can also be derived similarly.

In order to maintain uniform accuracy, it is necessary to use a grid stencil where the numbers of grid points in the three sections satisfy:

$$M = K = N + 1 \tag{68}$$

In other words, we can use one less grid point in the mid section without lowering the order of approximation. For example, if there is only one grid point in the mid section ($N = 1$), we can still obtain an $O(h)$ approximation for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ by using the following grid stencil of one point in the middle section and two points in each of the two other sections, i.e.

$$\begin{aligned}&M = K = 2\\&N = 1\end{aligned}\tag{69}$$

The corresponding finite difference formula is

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{d_{-2}u_{i-2} + d_{-1}u_{i-1} + d_0u_i + d_1u_{i+1} + d_2u_{i+2} + d_{11}A_1 + d_{12}hB_1 + d_{13}A_2 + d_{14}hB_2}{h^2} + O(h)\tag{70}$$

The coefficients in the above equation can be derived by using the general formulas presented in this section.

3. Application to one-dimensional equations

The new high-order finite difference approximation to derivatives at irregular points has been tested in several one-dimensional equations with discontinuous coefficients and delta function source terms. An example is shown here for the following one-dimensional equation:

$$\frac{\mathrm{d}^2u}{\mathrm{d}x^2} + \alpha^2u = \beta\delta(x - x_\Gamma) \quad (-0.5 \leqslant x \leqslant 0.5)\tag{71}$$

where β is a constant, and α is discontinuous across the interface located at $x = x_\Gamma$:

$$\alpha = \begin{cases}\alpha_1 & -0.5 \leqslant x \leqslant x_\Gamma\\ \alpha_2 & x_\Gamma \leqslant x \leqslant 0.5\end{cases}\tag{72}$$

where α_1 and α_2 are known integers. The boundary conditions are: $u(-0.5) = u(0.5) = 0$. In this equation, there is a discontinuity in $u_x(x)$ across the interface x_Γ, i.e.

$$\begin{aligned}&[u]_\Gamma = 0\\&[u_x]_\Gamma = \beta\end{aligned}\tag{73}$$

The exact solution is:

$$u_{\mathrm{ex}}(x) = \begin{cases}\dfrac{\beta\cos(\alpha_2x_\Gamma)\cos(\alpha_1x)}{\alpha_1\cos(\alpha_2x_\Gamma)\sin(\alpha_1x_\Gamma) - \alpha_2\sin(\alpha_2x_\Gamma)\cos(\alpha_1x_\Gamma)} & -0.5 \leqslant x \leqslant x_\Gamma\\[2ex] \dfrac{\beta\cos(\alpha_1x_\Gamma)\cos(\alpha_2x)}{\alpha_1\cos(\alpha_2x_\Gamma)\sin(\alpha_1x_\Gamma) - \alpha_2\sin(\alpha_2x_\Gamma)\cos(\alpha_1x_\Gamma)} & x_\Gamma \leqslant x \leqslant 0.5\end{cases}\tag{74}$$

A set of uniform grid of $N + 1$ points is used to discretize the equation. The coordinate of an arbitrary grid point is:

$$x_i = -0.5 + ih \quad (i = 0, \ldots, N)\tag{75}$$

where $h = 1/N$. The interface is located at the interval between $i = i_\Gamma$ and $i = i_\Gamma + 1$, where $i_\Gamma = \lfloor (x_\Gamma + 0.5)/h \rfloor$. The interface location parameter σ is:

$$\tilde{\sigma} = \frac{x_\Gamma + 0.5}{h} - i_\Gamma\tag{76}$$

In this case, all grid points are regular points except the two grid points located immediately next to the interface, i.e., the indices of the two irregular points are: $i = i_\Gamma$ and $i = i_\Gamma + 1$.

In this paper, six versions of the new immersed interface method (Methods A–F in Table 1) with different combinations of orders for the regular points and for the irregular points are considered. The finite difference formulas to discretize Eq. (71) by Method A are shown here as an example. In Eq. (71), $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ is computed by Eq. (8) for the regular grid points. For the irregular points ($i = i_\Gamma$ and $i = i_\Gamma + 1$), locally $O(h)$ ($n = m = 2$) finite difference formulas for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ are used. The difference formulas for the irregular points have different forms, depending on whether x_Γ is located on the left or right side of the grid point: Eq. (43) for $i = i_\Gamma$ and Eq. (46) for $i = i_\Gamma + 1$. Hence,

Table 1
Six versions of current immersed interface method tested in this paper

Methods	Order at regular grid points	Order at irregular grid points	Expected global order of accuracy
Method A	$O(h^2)$	$O(h)$ $(n = m = 2)$	$O(h^2)$
Method B	$O(h^2)$	$O(h^2)$ $(n = m = 3)$	$O(h^2)$
Method C	$O(h^4)$	$O(h)$ $(n = m = 2)$	$O(h^2)$
Method D	$O(h^4)$	$O(h^2)$ $(n = m = 3)$	$O(h^3)$
Method E	$O(h^4)$	$O(h^3)$ $(n = m = 4)$	$O(h^4)$
Method F	$O(h^4)$	$O(h^4)$ $(n = m = 5)$	$O(h^4)$

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \begin{cases} \dfrac{d_{-1}u_{i-1} + d_0u_i + d_1u_{i+1} + d_2u_{i+2} + d_AA + hd_BB}{h^2} & i = i_\Gamma \\ \dfrac{\tilde{d}_2u_{i-1} + \tilde{d}_1u_i + \tilde{d}_0u_{i+1} + \tilde{d}_{-1}u_{i+2} - \tilde{d}_AA + h\tilde{d}_BB}{h^2} & i = i_{\Gamma+1} \\ \dfrac{u_{i-1} - 2u_i + u_{i+1}}{h^2} & \text{others} \end{cases} \tag{77}$$

where d_ks for $i = i_\Gamma$ are computed by Eq. (23) directly by using the value of $\sigma = \tilde{\sigma}$ given by Eq. (76) and the following jump parameters for Eq. (71):

$$\alpha^+ = \beta^+ = c_\alpha = c_\beta = 1, \quad A = 0, \ B = \beta \tag{78}$$

The coefficients $\tilde{d}_k$s for $i = i_{\Gamma+1}$ are computed by Eq. (23) using the value of $\sigma = 1 - \tilde{\sigma}$ and jump conditions (78) (see Section 2.3).

3.1. Stiffness of difference operator with new high-order approximation at interface

Eq. (77) can be written in the vector form:

$$\frac{\mathrm{d}^2}{\mathrm{d}x^2}\begin{bmatrix} u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} = \mathbf{M}\begin{bmatrix} u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} + \mathbf{b} \tag{79}$$

where $\mathbf{M}$ is the coefficient matrix. The stability and the stiffness of the numerical computations involving the difference formulas given by Eq. (79) can be measured by the condition number of $\mathbf{M}$. Therefore, it is interesting to evaluate the effect of the current interface treatment on the stiffness of $\mathbf{M}$ by comparing its condition numbers for two cases with and without the interface treatment. The condition number of $\mathbf{M}$ is approximately measured by the ratio of the maximum and minimum magnitudes of its eigenvalue spectrum, i.e.

$$\mathrm{Cond} = \frac{(\max|\lambda|)_{\lambda\in\sigma(\mathbf{M})}}{(\min|\lambda|)_{\lambda\in\sigma(\mathbf{M})}} \tag{80}$$

where $\sigma(\mathbf{M})$ denotes the set of all eigenvalues of $\mathbf{M}$. The relative stiffness of $\mathbf{M}$ is measured by the ratio of the condition number when there are irregular point treatments and that when there are no interface treatments, i.e. the following "stiffness ratio" is computed:

$$\text{Stiffness ratio} = \frac{\mathrm{Cond}_{\mathrm{IIM}}}{\mathrm{Cond}_{\mathrm{Reg}}} \tag{81}$$

where $\mathrm{Cond}_{\mathrm{IIM}}$ and $\mathrm{Cond}_{\mathrm{Reg}}$ are the condition numbers of $\mathbf{M}$ with and without using the IIM formulas at the irregular grid points. Therefore, the ratio measures the relative increase of stiffness of $\mathbf{M}$ when the IIM formulas are used at the irregular points. At the same time, the effect of the interface location on the stiffness of the coefficient matrix can be examined. It should be pointed out that the value of β does not show up in the coefficient matrix. Instead, it is part of the non-homogeneous term in Eq. (79). Hence it does not affect the stiffness of the matrix.

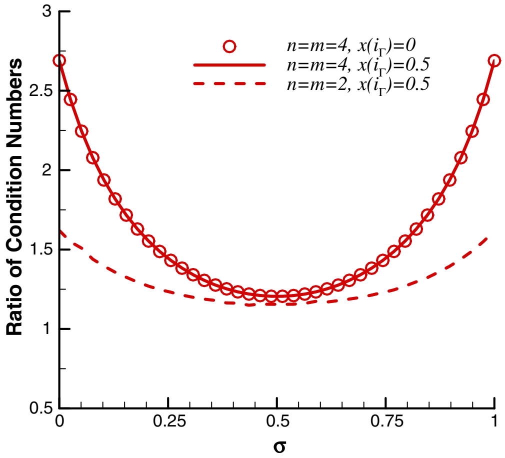


Fig. 6. Condition-number ratios of coefficient matrix **M** for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ by Method A ($O(h^2)$ and $n = m = 2$) and Method E ($O(h^4)$ and $n = m = 4$) as functions of the interface location parameter σ.

Fig. 6 shows the stiffness ratios of **M** for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ for Method A (globally $O(h^2)$ and $n = m = 2$) and Method E (globally $O(h^4)$ and $n = m = 4$) as functions of the interface location parameter σ. The figure shows that the stiffness of **M** is the lowest when the interface is located midway between two grid points ($\sigma = 0.5$). The stiffness increases when the interface moves closer to one of the two end points. It is the highest when x_Γ coincides with one of the two grid points ($\sigma = 0$ or $\sigma = 1$). In the case of Method A with a locally $O(h)$ at the interface, the stiffness ratio is:

- 1.15 when $\sigma = 0.5$
- 1.62 when $\sigma = 0$ or $\sigma = 1$.

This means that the interface treatment in this case is at most 1.62 times as stiff as the regular case without the interface. When the order of the approximation is increased to $O(h^4)$ for Method E, the stiffness ratio increases slightly to:

- 1.21 when $\sigma = 0.5$
- 2.69 when $\sigma = 0$ or $\sigma = 1$.

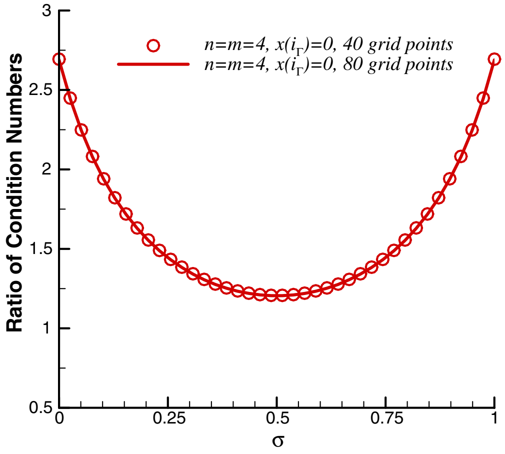


Fig. 7. Condition-number ratios of **M** for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$ by Method E ($O(h^4)$ and $n = m = 4$) as a function σ for two cases of different number of grid points.

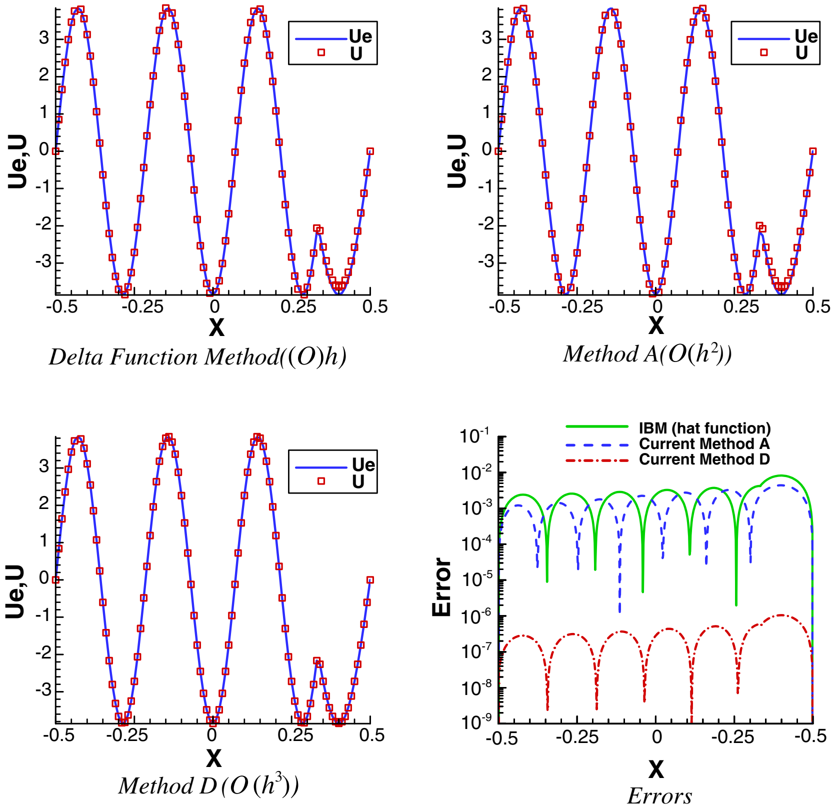
This figure also shows that the stiffness ratio does not change much for different values of $x(i = i_\Gamma)$ (the figure shows the cases of $x(i = i_\Gamma) = 0$ and $x(i = i_\Gamma) = 0.5$). Similarly, the computational results also show that the stiffness ratio stays about the same between two cases of different grid points ($N = 40$ and $N = 80$) as shown in Fig. 7.

Therefore, the use of current interface treatment slightly increases the stiffness of the coefficient matrix **M**. Along the same line, in order to reach higher-order accuracy, the current interface treatment results in a non-symmetric coefficient matrix **M**, which is different from that of the regular case without an interface. The non-symmetric coefficient matrix may increase the cost of solving the linear equations involving **M**. Nevertheless, this is a reasonable price to pay in order to obtain higher-order accuracy at the interface as long as the benefits of the higher-order accuracy outweight the slight increases of stiffness and the loss of symmetry in **M**. The actual benefits of the current high-order interface treatment for practical two and three-dimensional two-phase flow simulation will be studied in a separate paper as mentioned in the abstract.

3.2. Results of one-dimensional test case

Fig. 8 shows the results of Eq. (71) computed by the current interface methods with the following parameters:

$$\beta = -40\pi$$
$$x_\Gamma = 0.25$$
$$\alpha_1 = 7\pi$$
$$\alpha_2 = 5\pi$$



Delta Function Method($(O)h$)

Method A($O(h^2)$)

Method D ($O(h^3)$)

Errors

Fig. 8. Comparison of the exact and numerical solutions for computing one-dimensional equation ($N = 80$) and numerical errors for solutions computed by three numerical methods ($N = 640$).

Table 2
Comparison of numerical errors of three methods for computing one-dimensional equation with a discontinuous interface

N	σ	Delta function method $O(h)$			Current method A $O(h^2)$			Current method D $O(h^3)$		
		$\Vert E_N\Vert_\infty$	Ratio	p	$\Vert E_N\Vert_\infty$	Ratio	p	$\Vert E_N\Vert_\infty$	Ratio	p
20	2/3	4.268(+1)			1.798(+1)			2.059		
40	1/3	1.377			1.279			2.820(−2)		
80	2/3	2.788(−1)	153.1	3.63	3.039(−1)	59.2	2.94	5.484(−3)	375.5	4.28
160	1/3	8.754(−2)	15.7	1.99	7.049(−2)	18.1	2.09	1.173(−4)	240.4	3.95
320	2/3	1.755(−2)	15.9	1.99	1.866(−2)	16.3	2.01	5.996(−5)	91.5	3.26
640	1/3	8.199(−3)	10.7	1.71	4.390(−3)	16.1	2.00	1.052(−6)	111.5	3.40
1280	2/3	2.276(−3)	7.71	1.47	1.167(−3)	16.0	2.00	8.244(−7)	72.7	3.09
2560	1/3	1.195(−3)	6.86	1.39	2.745(−4)	16.0	2.00	1.368(−8)	76.9	3.13

Three interface methods are tested:

1. Immersed boundary method where the delta function in Eq. (71) is approximated by the standard linear hat function.
2. Method A in Table 1 with global $O(h^2)$ accuracy, where (77) is used for $(\mathrm{d}^2u/\mathrm{d}x^2)_i$.
3. Method D of Table 1 with a global $O(h^3)$ accuracy, where $O(h^4)$ approximation is used at regular points and a locally $O(h^2)$ formula ($n = m = 3$, Eq. (117)) is used at the irregular points.

Fig. 8 compares the solutions of the three methods with the exact solution. The solution is continuous at x_Γ, but there is a jump in the first derivatives at the interface. The three methods capture the solutions and the discontinuity at the interface very well. There are no numerical oscillations in the three sets of numerical solutions.

Fig. 8 also shows the errors of the solution obtained by using the three methods. For the current grid of $N = 640$, the second order Method A is slightly more accurate than the standard IBM method in computing the interface. On the other hand, the numerical errors of the 3rd order Method D is about two orders of magnitudes lower than those of the standard IBM method.

Table 2 shows the maximum-norm errors of the three methods, the corresponding error ratios, and computed orders of accuracy p using eight sets of grids: $N = 20, 40, 80, \ldots, 2560$. Since the location of the interface, x_Γ, is fixed when the grids are refined, the values of σ, which represents the relative location of the interface with respect to the grid points, vary alternatively between 1/3 and 2/3 (Table 2). In order to compare the grid refinement results with the same conditions, the results are compared between grids N and $N/4$ (both have the same σ) in the grid refinement study. When the number of grids is increased by a factor of 4, the errors are expected to decrease by a factor of 4^p for a pth order method. The error ratio in Table 2 is defined as

$$\text{Ratio} = \frac{\Vert E_{N/4}\Vert_\infty}{\Vert E_N\Vert_\infty} \tag{82}$$

The order p in the same table computed by the grid refinement study is:

$$p = \frac{\ln(\Vert E_{N/4}\Vert_\infty / \Vert E_N\Vert_\infty)}{\ln(4)} \tag{83}$$

Table 2 shows that both method A second order) and method D (third order) produce numerical results which are consistent with their orders of accuracy. The current second and third order methods lead to much more accurate results than the first-order δ function method.

4. Application to two-dimensional equations

All six versions of the new high-order immersed interface method listed in Table 1, with orders ranging from $O(h^2)$ to $O(h^4)$, have been tested for a two-dimensional equation (in Section 4.1) used by LeVeque and Li [24]. In addition, the second order Method A and the fourth-order Method E have been tested for another example (in Section 4.2) of LeVeque and Li [24]. The accuracy of the new methods is evaluated by grid refinements and by comparing the current results with those of the original immersed interface method by LeVeque and Li.

4.1. Two-dimensional example 1

The first two-dimensional example, which was used by LeVeque and Li [24] to test their second order IIM method, is:

$$u_{xx} + u_{yy} = \int_{\Gamma} 2\delta(x - X(s))\delta(y - Y(s))\mathrm{d}s \tag{84}$$

where the interface Γ is a circle defined by: $x^2 + y^2 = 1/4$. The computational domain is $-1 \leqslant x, y \leqslant 1$ shown in Fig. 1. The Dirichlet boundary condition is specified along the boundary by using the exact solution:

$$u(x,y) = \begin{cases} 1 & \text{if } r \leqslant 1/2 \\ 1 + \log(2r) & \text{if } r > 1/2 \end{cases} \tag{85}$$

where $r = \sqrt{x^2 + y^2}$. The jump conditions at all points on Γ are

$$[u]_{\Gamma} = 0 \tag{86}$$

$$\left[\frac{\partial u}{\partial n}\right]_{\Gamma} = 2 \tag{87}$$

The computational domain is discretized by uniform grid as shown in Fig. 1.

The current one-dimensional difference formulas in Section 2 for high-order immersed interface method are extended to two (and three) dimensional equations by a dimension by dimension approach. For a regular grid point away from the interface, the derivatives with respective to x and y in Eq. (84) can be approximated by standard central difference formulas below:

$$\left(\frac{\partial^2 u}{\partial x^2}\right)_{i,j} = \begin{cases} \dfrac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{\Delta x^2} & O(\Delta x^2) \text{ approximation} \\ \dfrac{-u_{i-2,j} + 16u_{i-1,j} - 30u_{i,j} + 16u_{i+1,j} - u_{i+2,j}}{12\Delta x^2} & O(\Delta x^4) \text{ approximation} \end{cases} \tag{88}$$

$$\left(\frac{\partial^2 u}{\partial y^2}\right)_{i,j} = \begin{cases} \dfrac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{\Delta y^2} & O(\Delta y^2) \text{ approximation} \\ \dfrac{-u_{i,j-2} + 16u_{i,j-1} - 30u_{i,j} + 16u_{i,j+1} - u_{i,j+2}}{12\Delta y^2} & O(\Delta y^4) \text{ approximation} \end{cases} \tag{89}$$

The irregular points are defined according to the coordinate directions. For example, a grid point is termed i-irregular point if it is next to the interface along the i grid line. Fig. 9 shows a schematic of a two-dimensional uniform grid with an immersed interface. In the figure, (i,j) is an i-directional irregular point, where the interface cuts across the horizontal grid line at the nearby point P. The relative location of the interface is measured by σ shown in the figure. Similarly, for y derivatives, the interface point for the j-irregular point (i,j) is located at point Q. In general, a grid point can be regular in one direction, but irregular in another. For interface point P, the normal vector of the interface is $\mathbf{n}$, which has angle α with respect to the x-axis.

In this case, the general finite difference formulas for the x-derivatives, such as Eqs. (43) and (46) for Method A, are used directly. The jump conditions on u and u_x for derivatives in x-direction are derived by applying a coordinate transformation to the general jump condition (87). The jump conditions of the first derivatives in the current example can be written in the following form:

$$\begin{aligned} \left[\frac{\partial u}{\partial s}\right]_{\Gamma} &= A \\ \left[\frac{\partial u}{\partial n}\right]_{\Gamma} &= B \end{aligned} \tag{90}$$

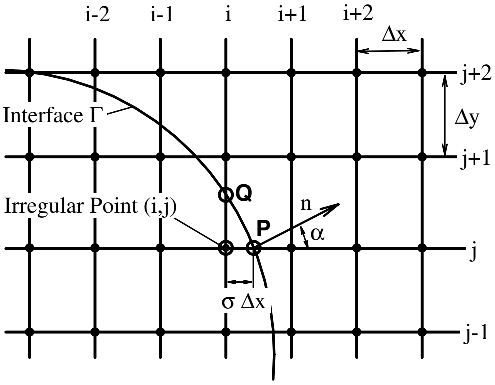


Fig. 9. A schematic of a two-dimensional uniform grid with in immersed interface Γ, where (*i*,*j*) is an *I*-directional irregular point (*J*-directional irregular points can be defined similarly).

where (s, n) is the local coordinate along the tangential and normal directions of the interface, respectively. In the case of Eq. (87), the corresponding values of A and B are:

$$A = 0 \quad \text{and} \quad B = 2 \tag{91}$$

The normal and tangential unit vectors in Fig. 9 are:

$$\begin{aligned}\mathbf{n} &= \cos\alpha\mathbf{i} + \sin\alpha\mathbf{j}\\ \mathbf{s} &= -\sin\alpha\mathbf{i} + \cos\alpha\mathbf{j}\end{aligned} \tag{92}$$

Hence Eq. (90) leads to:

$$\begin{aligned}\left[\cos\alpha\frac{\partial u}{\partial x} + \sin\alpha\frac{\partial u}{\partial y}\right]_\Gamma &= B\\ \left[-\sin\alpha\frac{\partial u}{\partial x} + \cos\alpha\frac{\partial u}{\partial y}\right]_\Gamma &= A\end{aligned} \tag{93}$$

Solving the two equations above:

$$\left[\frac{\partial u}{\partial x}\right]_\Gamma = B\cos\alpha - A\sin\alpha \tag{94}$$

$$\left[\frac{\partial u}{\partial y}\right]_\Gamma = B\sin\alpha + A\cos\alpha \tag{95}$$

Therefore, in this case, we can directly use the general one-dimensional formulas for derivatives in the x-direction at i-irregular points together with the jump conditions (86) and (94). For example, if Method A ($n = m = 2$) is used, $(\partial^2 u/\partial x^2)_{i,j}$ at an irregular grid point can be calculated by Eq. (43) together with Eqs. (86) and (94). Using the same approach, formulas for $(\partial^2 u/\partial y^2)_{i,j}$ in the y-direction at a j-irregular point can be derived at interface point Q shown in Fig. 9.

In the computer programming for a two-dimensional problem, the grid points are classified into four categories according to their relative locations with respect to the interface in a dimension-by-dimension manner. For example, a grid point can be classified as one of the four kinds for the purpose of discretization in the i-direction:

1. Regular point in i-direction;
2. Irregular point with an interface located between (*i*,*j*) and $(i + 1, j)$;
3. Irregular point with an interface located between $(i - 1, j)$ and (*i*,*j*);
4. Irregular point of special kind, such as the case when point (*i*,*j*) is near two interfaces on both sides as shown in Fig. 4.

Similar classification of grid points is also defined independently for the derivatives in the y-direction. Subsequently, all interface information can be calculated, and the finite difference formulas can be derived for each grid point. The resulting linear finite difference equation can be solved by a number of elliptic equation solvers. Since the focus of this paper is on the introduction of the basic algorithms, the linear equations are computed by a direct solver in this paper, for the sake of simplicity.

4.1.1. Eigenvalue spectra of the coefficient matrices

The discretization of Eq. (84) by an immersed interface method in a uniform grid of $N \times M$ points leads to a set of $K = (N-2) \times (M-2)$ linear equations for all $u_{i,j}$ in the interior of the computational domain in the range of $i = 2, \ldots, N-1$ and $j = 2, \ldots, M-1$. The linear equations can be written in the vector form as:

$$\mathbf{PU} = \mathbf{b} \tag{96}$$

where

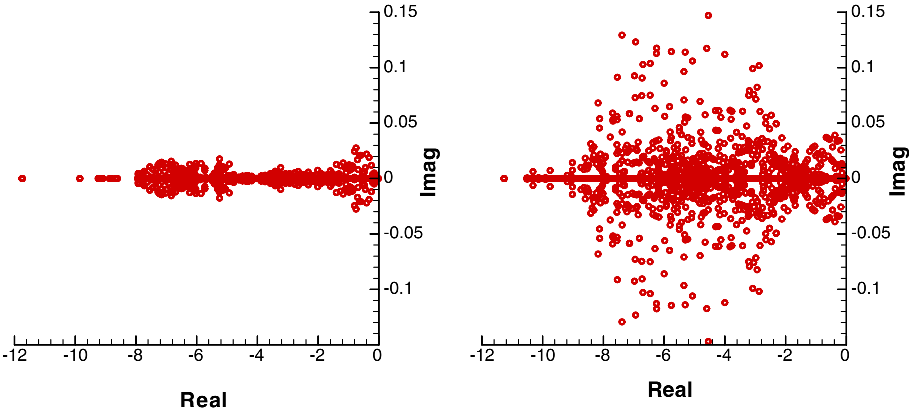
$$\mathbf{U} = [u_{2,2}, \ldots, u_{2,N-1}, u_{3,2}, \ldots, u_{M-1,N-1}]^{\mathrm{T}} \tag{97}$$

where $\mathbf{P}$ is the coefficient matrix of dimension $K \times K$, $\mathbf{U}$ is the vector for all $u_{i,j}$ in the interior of the computational domain, and $\mathbf{b}$ is the non-homogeneous term. The stability of numerical computations involving a finite difference approximation for Eq. (84) requires that all eigenvalues λ of $\mathbf{P}$ satisfies:

$$Re(\lambda) \leqslant 0 \tag{98}$$

Therefore, it is necessary to evaluate the effect of the current interface treatment method on the eigenvalue spectra of matrix $\mathbf{P}$.

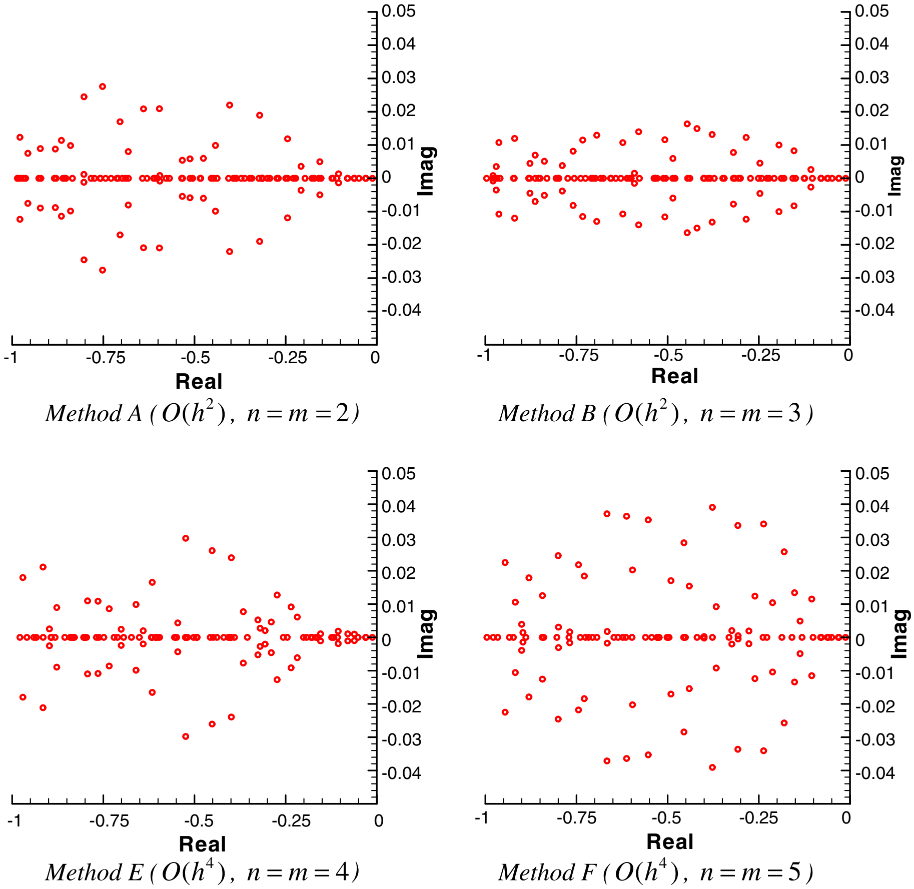
Fig. 10 shows the eigenvalue spectra of the coefficient matrix for discretizing Eq. (84) by using methods A and F defined in Table 1. The two methods treat the regular and irregular points with different orders of approximation. Method A is globally second order, while method F is fourth order. A uniform 40 × 40 grid is used in the computations. The eigenvalues in the figure are normalized by a factor of Δx^2 so that the normalized eigenvalues are "dimensionless". The figure shows that the real part of all eigenvalues of both method A and method F are always negative, which satisfies the stability condition of Eq. (98). The computations of the eigenvalue spectra for all six methods listed in Table 1 show that the current interface treatment is stable for Eq. (84). Fig. 11 shows the spectra of four of those methods in the local area near the origin. Again, all eigenvalues satisfy the stability condition given by Eq. (98). Therefore, the proposed high-order immersed interface methods do satisfy the stability condition in terms of eigenvalue spectra for the two-dimensional test problem of Eq. (84).



Method A ($O(h^2)$, $n = m = 2$)

Method F ($O(h^4)$, $n = m = 5$)

Fig. 10. Eigenvalue spectra of the coefficient matrix for 2D example 1 approximated by methods A and F, which treat the regular and irregular points by different orders of approximation (grid: 40 × 40).



Method A ($O(h^2)$, $n = m = 2$)

Method B ($O(h^2)$, $n = m = 3$)

Method E ($O(h^4)$, $n = m = 4$)

Method F ($O(h^4)$, $n = m = 5$)

Fig. 11. Eigenvalue spectra of the coefficient matrix for 2D example 1 approximated by methods A, B, E and F defined in Table 1 (grid: 40 × 40).

4.1.2. Numerical results and comparison

LeVeque and Li [24] computed the same test case by using both the Delta function approach of the immersed boundary method and their second-order immersed interface method. Here, their results are compared with those computed by the six methods listed in Table 1.

Fig. 12 shows the numerical solution computed by Method A using 80 × 80 grid. There is a discontinuity in derivatives at the interface. This figure also compares the contours of the exact solution and those computed by current fourth-order immersed interface method E. The two solutions agree very well with each other. There are no visible differences between the two solutions as shown in the figures. In addition, there are no oscillations in the numerical solutions at the interfaces with discontinuous gradients.

A grid refinement study based on the infinity-norm numerical errors, $\|E_N\|_\infty$, has been conducted by using the following two sets of grids: 40 × 40 and 80 × 80 grids. Table 3 shows the numerical errors of the six current methods of various orders (Table 1), as well as the results of LeVeque and Li's original second-order immersed interface method and the delta function approximation results. For a pth order method, the ratio of the errors of two successive sets of grids should approach the following limit:

$$\text{Ratio} = \frac{\|E_N\|_\infty}{\|E_{2N}\|_\infty} \to 2^p \qquad (99)$$

Table 3 shows that the current second-order methods (Methods A–C) produce the same level of accuracy as the original second-order IIM of LeVeque and Li [24]. The error ratios and computed order of accuracy p are also listed in the table. Their error ratios are slightly better than the asymptotic value of 4. In the case of Meth-

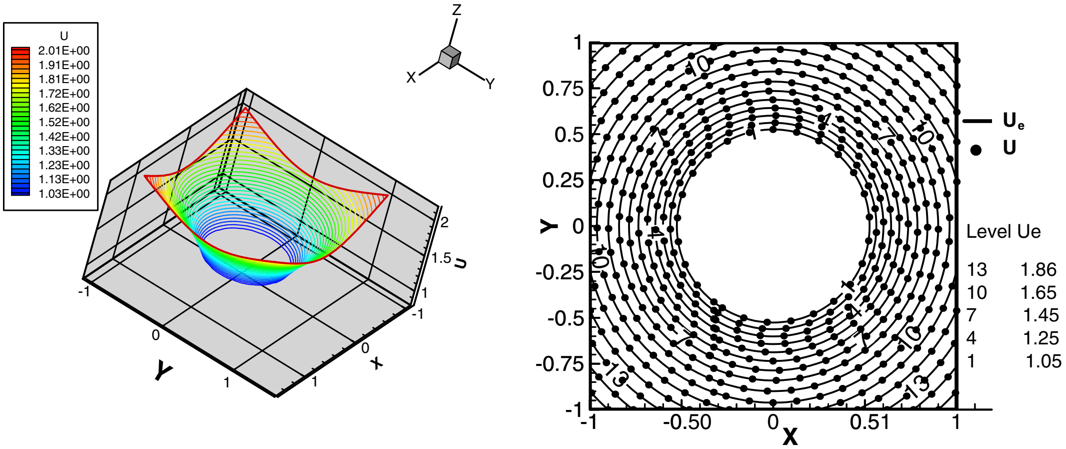


Fig. 12. Solution, $z = u(x, y)$, computed by Method A ($O(h^2)$) for the first two-dimensional example and comparison of the contours of the exact (u_e) and numerical (u) solutions (Method E $O(h^4)$, irregular points: $n = m = 4$, grid: 80 × 80).

Table 3
Comparison of computational errors of six current immersed interface methods (from $O(h^2)$ to $O(h^3)$) and those of the original second-order IIM of LeVeque and Li [24]

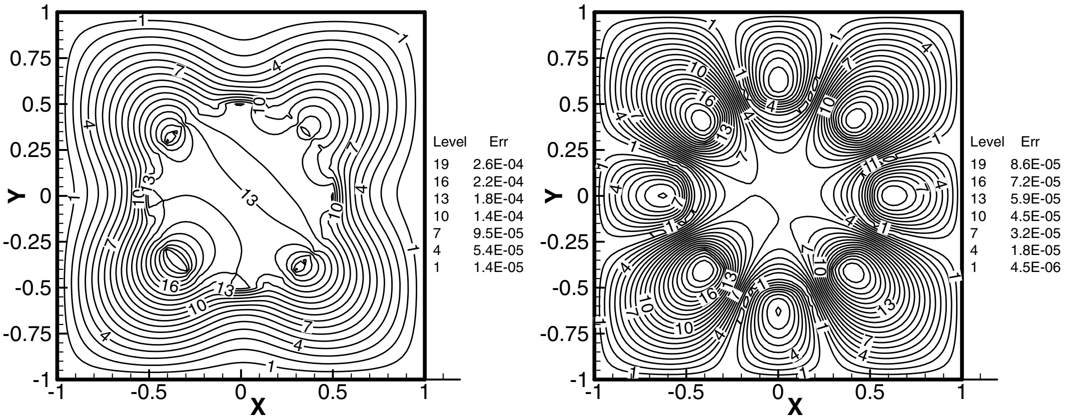
Methods	$\|E_N\|_\infty$ 40 × 40 Grid	$\|E_N\|_\infty$ 80 × 80 Grid	Error ratio	p
δ Function method	2.6467×10^{-2}	1.3204×10^{-2}	2.0045	1.003
LeVeque & Li's ($O(h^2)$) IIM	8.3461×10^{-4}	2.4451×10^{-4}	3.4134	1.771
Current Method A ($O(h^2)$)	1.6339×10^{-3}	2.8581×10^{-4}	5.7165	2.515
Current Method B ($O(h^2)$)	4.4405×10^{-4}	9.5040×10^{-5}	5.7244	2.517
Current Method C ($O(h^2)$)	1.5715×10^{-3}	2.5039×10^{-4}	6.2763	2.650
Current Method D ($O(h^3)$)	4.9529×10^{-4}	4.7499×10^{-5}	10.4275	3.382
Current Method E ($O(h^4)$)	1.2215×10^{-4}	6.1514×10^{-6}	19.8574	4.311
Current Method F ($O(h^4)$)	1.5521×10^{-5}	3.4286×10^{-7}	45.2678	5.500

od C, though the fourth-order central scheme is used in the regular points, only the locally first-order method (Eq. (43), $n = m = 2$) is used at the interface. This table shows this method is only globally $O(h^2)$ because the interface treatment is first order accurate.

The table also shows that when a fourth-order approximation is used at the regular points with second or higher-order approximation at the interface (Methods D–F), the numerical solutions have much higher accuracy. In the case of overall fourth order Method F, the errors with 80 × 80 grid are two orders of magnitudes lower than those of LeVeque and Li's original second-order immersed interface method. The errors ratios also show very high convergence rates. The contours of the local errors of the current six IIM methods tested are shown in Figs. 13 and 14. The figures show that the main errors in the computations are originated from the approximation at the interface, which demonstrates the importance of using higher-order approximation methods for interface treatment. The fourth-order IIM methods (Methods E and F) maintain high level of computational accuracy as expected. Table 3 and Figs. 13 and 14 also show that although both Method E and Method F are fourth order, the use of fourth-order approximation at irregular points in Method F results in much higher accuracy.

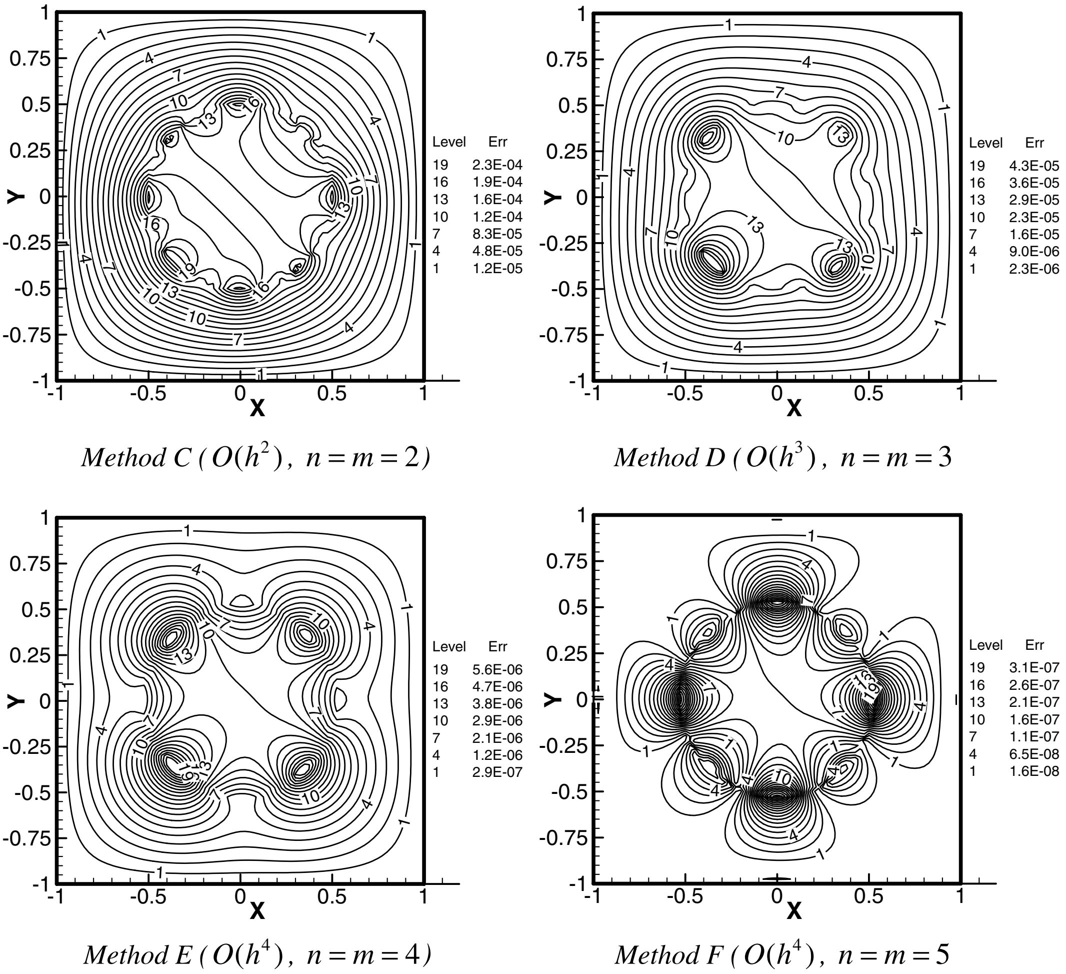
4.2. Two-dimensional example 2

In this section, current Method A ($O(h^2)$) and Method E ($O(h^4)$) are tested on the second example computed by LeVeque and Li [24]. The equation is a two-dimensional Poisson equation with a delta function source term and with a discontinuous coefficient β as follows:



Method A ($O(h^2)$, $n = m = 2$)   *Method B* ($O(h^2)$, $n = m = 3$)

Fig. 13. Contours of local errors computed by two current interface methods. Both methods use the same $O(h^2)$ approximation at regular points, but their irregular point treatments are of different orders, ranging from $O(h)$ to $O(h^2)$ (grid: 80 × 80).



Method C ($O(h^2)$, $n = m = 2$)   *Method D* ($O(h^3)$, $n = m = 3$

Method E ($O(h^4)$, $n = m = 4$)   *Method F* ($O(h^4)$, $n = m = 5$

Fig. 14. Contours of local errors computed by four current interface methods. All methods use the same $O(h^4)$ approximation at regular points, but their irregular point treatments are of different orders, ranging from $O(h)$ to $O(h^4)$ (grid: 80 × 80).

$$(\beta u_x)_x + (\beta u_y)_y = f(x,y) + C\int_\Gamma \delta(\vec{x} - \vec{X}(s))\mathrm{d}s \tag{100}$$

where

$$f(x,y) = 8r^2 + 4 \tag{101}$$

The interface Γ is a circle specified by $x^2 + y^2 = 1/4$, and the computational domain is $-1 \leqslant x, y \leqslant 1$ shown in Fig. 1. The coefficient β is discontinuous at the interface:

$$\beta(x,y) = \begin{cases} r^2 + 1 & \text{if } r^2 \leqslant 1/4 \\ b & \text{if } r^2 > 1/4 \end{cases} \tag{102}$$

where the value of b determine the magnitude of jump in β across the interface. The Dirichlet boundary condition is specified along the boundary by using the following exact solution:

$$u_e(x,y) = \begin{cases} r^2 & \text{if } r \leqslant \frac{1}{2} \\ \frac{1}{4} + \frac{-9+r^4+2r^2+c\log(2r)}{2b} & \text{if } r > \frac{1}{2} \end{cases} \tag{103}$$

For the current case, the jump conditions at Γ are

$$[u]_\Gamma = 0 \tag{104}$$

$$\left[\frac{\partial u}{\partial s}\right]_\Gamma = 0 \tag{105}$$

$$\left[\beta\frac{\partial u}{\partial n}\right]_\Gamma = C \tag{106}$$

A uniform grid is used to compute the two-dimensional equation. Again, the extension of the current interface Methods A and E to a general two-dimensional problem is done by a dimension-by-dimension approach. Before the difference approximation at regular and irregular points is applied, Eq. (100) is rewritten in the following form:

$$\beta(u_{xx} + u_{yy}) + \beta_x u_x + \beta_y u_y = f(x,y) + C\int_\Gamma \delta(\vec{x} - \vec{X}(s))\mathrm{d}s \tag{107}$$

The approximation for the x derivatives, u_x and u_{xx} at a grid point (i,j), can then be approximated by using finite difference formulas for either regular or i-irregular points. In order to be consistent with the orders at the interface, the second and fourth-order central schemes are used for the regular points for Methods A and E respectively. Similar approximation can be done for the y derivatives, u_y and u_{yy}.

In order to apply the new immersed interface method in a grid direction, it is necessary to derive jump conditions for partial derives of u in either x or y-directions only. The general jump conditions for the normal and tangential derivatives can be expressed in the general form below:

$$\left[\alpha\frac{\partial u}{\partial s}\right]_\Gamma = B_s \tag{108}$$

$$\left[\beta\frac{\partial u}{\partial n}\right]_\Gamma = B_n \tag{109}$$

Using the normal and tangential unit vectors at the interface given by Eqs. (92), (108) and (109) are transformed into Cartesian coordinates as follows:

$$\left[\tilde{\beta}\frac{\partial u}{\partial x}\right] = \widetilde{B}_x \tag{110}$$

where

$$\tilde{\beta} = \beta\cos^2\theta + \alpha\sin^2\theta \tag{111}$$

$$\widetilde{B}_x = B_n\cos\theta - B_s\sin\theta - \left[(\beta - \alpha)\cos\theta\sin\theta\frac{\partial u}{\partial y}\right] \tag{112}$$

For finite difference approximation of x derivatives at an irregular point, the jump condition (110) is used. The y derivative term, $\mathrm{d}u/\mathrm{d}y$, on the right hand side of (112) is evaluated by one-sided difference formulas at an order of accuracy which is consistent with the order of accuracy of the overall calculations. Similarly, for finite difference approximation of y derivatives, the following jump condition is used:

$$\left[\tilde{\alpha}\frac{\partial u}{\partial y}\right] = \widetilde{B}_y \tag{113}$$

where

$$\tilde{\alpha} = \beta\sin^2\theta + \alpha\cos^2\theta \tag{114}$$

$$\widetilde{B}_y = B_n\sin\theta + B_s\cos\theta - \left[(\beta-\alpha)\cos\theta\sin\theta\frac{\partial u}{\partial x}\right] \tag{115}$$

Again, the x derivative term, $\mathrm{d}u/\mathrm{d}x$, on the right hand side of (115) is evaluated by one-sided difference formulas at an order of accuracy which is consistent with the order of accuracy of the overall calculations. This dimension-by-dimension treatment of the two-dimensional jump conditions is tested in the current example.

This two-dimensional example has been computed by the current second-order interface method A ($n = m = 2$) and the fourth-order method E ($n = m = 4$), with three sets of grids: 20 × 20, 40 × 40 and 80 × 80 points. Figs. 15 and 16 show the results for the case with the following parameters used by LeVeque and Li [24]: $b = 10$ and $C = 0.1$. The figures show that the current numerical results agree very well with the exact solution. The maximum-norm errors for the solutions computed by using Methods A and E are compared in Table 4 with those of LeVeque and Li [24] using their original second order IIM method. The table shows that the current Method A has a similar second-order error ratio as the original IIM method, though the magnitudes of the errors of the current method A are about an order of magnitude smaller. For the case of 80 × 80 points, the error of method A is: $\|E_n\|_\infty = 1.6862\times10^{-5}$. This is equivalent to the second order original IIM method of LeVeque and Li [24], where the corresponding error is 1.6512×10^{-4}. Table 4 also shows that the use of the current and fourth-order IIM method E leads to much better accuracy for the computations for this two-dimensional example. The error for method E with the 80 × 80 grid is 2.7026×10^{-6}. Therefore, the use of the current high-order immersed interface methods lead to significant improvement of the accuracy of the solutions.

For the 2D example 2, the magnitude of the jump in the first derivatives across the interface depends on the parameter b as shown in Eq. (102). According to the exact solution (103), the jump in derivatives increases as the value of b decreases. In addition, as b decreases, the maximum magnitude of $|u(x,y)|$ increases. Therefore, it is expected that, with the same grid resolution, the computational errors will increase when the value of b

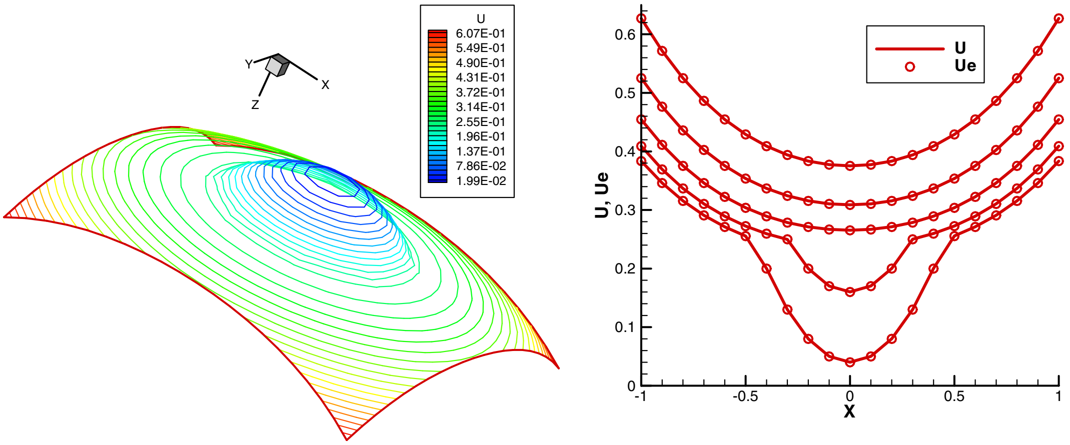
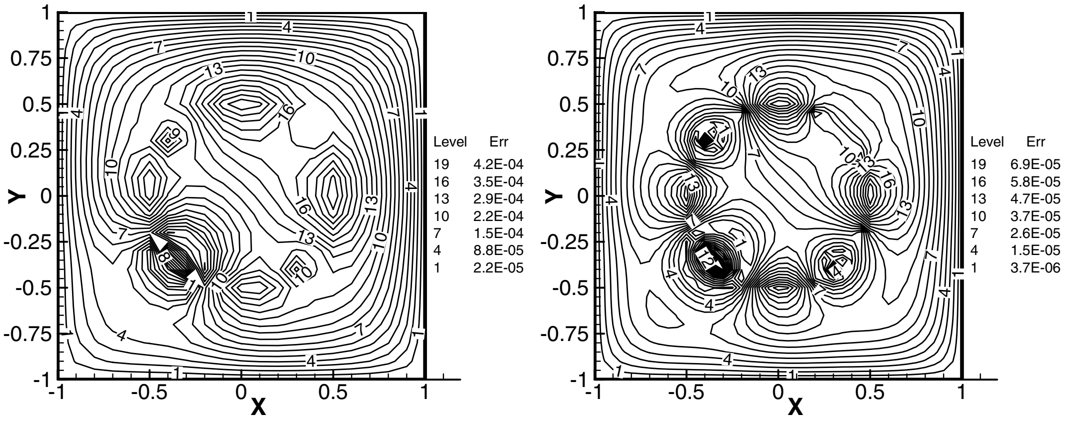


Fig. 15. Solution, $z = u(x,y)$ and comparison of the profiles of the exact (u_e) and numerical (u) solutions along a number of horizontal grid lines computed by Method A ($O(h^2)$) for the second two-dimensional example with a set of 20 × 20 grid.



(20×20 *grid)*

(40×40 *grid)*

Fig. 16. Contours of local errors computed by current second-order interface method A for two sets of grids.

Table 4
Comparison of numerical errors of current immersed interface method A ($O(h^2)$) and those of the original second-order IIM of LeVeque and Li [24]

Methods	20 × 20 Grid	40 × 40 Grid		80 × 80 Grid	
	$\|E_N\|_\infty$	$\|E_N\|_\infty$	Error ratio	$\|E_N\|_\infty$	Error ratio
LeVeque & Li's $O(h^2)$ IIM	3.5195×10^{-3}	7.5613×10^{-4}	4.65	1.6512×10^{-4}	4.58
Current $O(h^2)$ Method A	4.6344×10^{-4}	7.4775×10^{-5}	6.20	1.6862×10^{-5}	4.43
Current $O(h^4)$ Method E	7.0678×10^{-6}	3.0719×10^{-5}	0.23	2.7026×10^{-6}	11.4

decreases. In this paper, the new second and fourth-order immersed interface methods A and E have been tested for a wide range of different values of b between 10 and 10^{-4}. The results are shown in Tables 5–7.

For the second order Method A, Table 5 shows that the errors increase as the values of b decrease as expected. For the three cases of b larger or equal to 1.0, the error ratios approach the expected value of 4.0 for the 80 × 80 grid. When b is smaller than 1.0, the error ratios gradually deviate from the expected values of 4.0. This is a result of the fact that the relative grid resolution drops when b decreases. More grid points are needed in order to achieve the expected error ratios for the cases of very small b. In other words, more grid points are necessary if higher accuracy is needed for those cases.

Table 5
Comparison of numerical errors of current immersed interface method A ($O(h^2)$) for example 2 with large and small jump conditions (different b values)

b	20 × 20 Grid	40 × 40 Grid		80 × 80 Grid	
	$\|E_N\|_\infty/u(1,1)$	$\|E_N\|_\infty/u(1,1)$	Error ratio	$\|E_N\|_\infty/u(1,1)$	Error ratio
10.0	7.3905×10^{-4}	1.1924×10^{-4}	6.20	2.6890×10^{-5}	4.43
5.0	9.1942×10^{-4}	1.5175×10^{-4}	6.06	3.4342×10^{-5}	4.42
1.0	1.3457×10^{-3}	2.6576×10^{-4}	5.06	6.4985×10^{-5}	4.09
0.5	1.8102×10^{-3}	4.4112×10^{-4}	4.10	1.1784×10^{-4}	3.74
0.1	4.6614×10^{-3}	1.9357×10^{-3}	2.41	5.2898×10^{-4}	3.66
0.05	7.9192×10^{-3}	3.7478×10^{-3}	2.11	1.0276×10^{-3}	3.65
0.01	2.8857×10^{-2}	1.9098×10^{-2}	1.51	5.1582×10^{-3}	3.70
0.005	4.6733×10^{-2}	4.1145×10^{-2}	1.14	1.0777×10^{-2}	3.82
0.001	9.5708×10^{-2}	6.1626×10^{-1}	0.155	8.8844×10^{-2}	6.94
0.0005	1.1041×10^{-1}	8.1457×10^{-1}	0.136	9.7951×10^{-1}	0.832
0.0001	1.2593×10^{-1}	2.8489×10^{-1}	0.442	1.3940×10^{-1}	2.04

Table 6
Comparison of numerical errors of current immersed interface method E ($O(h^4)$) for example 2 with large and small jump conditions (different b values)

b	20 × 20 Grid	40 × 40 Grid		80 × 80 Grid	
	$\|E_N\|_\infty/u(1,1)$	$\|E_N\|_\infty/u(1,1)$	Error ratio	$\|E_N\|_\infty/u(1,1)$	Error ratio
10.0	1.1271×10^{-5}	4.8988×10^{-5}	0.23	4.3099×10^{-6}	11.4
1.0	2.0064×10^{-5}	7.9568×10^{-6}	2.52	8.8516×10^{-7}	8.99
0.1	6.7636×10^{-5}	1.3981×10^{-4}	0.48	2.5626×10^{-5}	3.66
0.01	4.9414×10^{-4}	1.0503×10^{-3}	0.47	2.4357×10^{-4}	4.31
0.001	3.6403×10^{-3}	9.1621×10^{-3}	0.40	2.4271×10^{-3}	3.77
0.0001	1.0775×10^{-2}	4.7165×10^{-2}	0.23	2.5376×10^{-2}	1.86

Table 7
Comparison of numerical errors of current immersed interface method E ($O(h^4)$) for example 2 with large and small jump conditions (b values) with a different set of grids

b	21 × 21 Grid	42 × 42 Grid		84 × 84 Grid	
	$\|E_N\|_\infty/u(1,1)$	$\|E_N\|_\infty/u(1,1)$	Error ratio	$\|E_N\|_\infty/u(1,1)$	Error ratio
10.0	1.1948×10^{-5}	3.5563×10^{-5}	0.34	2.8838×10^{-6}	12.3
1.0	1.8540×10^{-4}	9.8585×10^{-6}	18.8	2.7161×10^{-6}	3.63
0.1	1.3619×10^{-3}	8.8457×10^{-5}	15.4	2.8870×10^{-5}	3.06
0.01	1.3569×10^{-2}	8.3963×10^{-4}	16.2	2.6654×10^{-4}	3.15
0.001	8.4079×10^{-1}	8.8555×10^{-3}	95.0	2.6261×10^{-3}	3.37
0.0001	1.6260×10^{-1}	2.4029×10^{-1}	0.677	2.5311×10^{-2}	9.49

On the other hand, the use of fourth-order immersed interface method E leads to significant improvement in terms of numerical accuracy as shown by the results in Table 6. The general trend of the errors for Method E when b decreases is the same as that for Method A. In order words, the grids for very small b are not fine enough to produce the expected error ratios for the fourth-order method. In addition, as discussed in Section 3.2, the computational errors for the current immersed interface method depend on the location of the interface represented by the value of σ. When the grids are refined from 40 × 40 grid to 80 × 80 grid, for example, the σ values at all interface locations will be different. As a result, the computational errors will be affected by the differences in σ. Therefore, the correct error ratios can be achieved only when σ is the same for the two sets of grids. This is shown in the 1D results of Table 2 for different grids with $\sigma = 2/3$ or $\sigma = 1/3$. To demonstrate this point, Table 7 shows the results of fourth-order method E for different values of b computed by slightly different grids: 21 × 21, 42 × 42 and 84 × 84 points. Though the trend of the results are the same as those of Table 6, the slight change in grid points lead to relatively large changes in accuracy mainly due to the change in σ. Overall, for all cases tested, fourth order Method E leads to much better numerical accuracy than the corresponding second order Method A.

5. Conclusions

A new arbitrarily high-order immersed interface method has been presented in this paper. The new method can be of arbitrarily high-order accuracy and it is simple to be applied to practical two-phase flow problems by requiring only the physical jump conditions for variables and first derivatives. It also has the advantage that the finite difference formulas at irregular points are expressed in an explicit form so that they can be applied to difference problems without modifications. Six versions of the new method of up to fourth-order accuracy have been tested for both one and two-dimensional model equations. The numerical results show that they can produce very accurate results for elliptic equations with embedded interfaces.

Acknowledgments

This work was sponsored by the Air Force Office of Scientific Research, USAF, under AFOSR Grant FA9550-04-1-0029. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of the Air Force Office of Scientific Research or the US Government.

Appendix A

A.1. Difference formulas at irregular points with six-point stencil $(n = m = 3)$

For the case shown in Fig. 2, where the interface is located on the right side of i with a given value of σ, we have:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = \frac{\sum_{k=-2}^{3} d_k u_{i+k} + d_A A + h d_B B}{3h} + \frac{3u_i - 4u_{i-1} + u_{i-2}}{2h} + O(h^3) \tag{116}$$

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{\sum_{k=-2}^{3} d_k u_{i+k} + d_A A + h d_B B}{h^2} + \frac{u_{i-2} - 2u_{i-1} + u_i}{h^2} + O(h^2) \tag{117}$$

The coefficients in the equations above are:

$$\begin{aligned}
d_{-2} &= \frac{1}{D}\left\{c_\alpha\left(\frac{11}{2}\sigma - \frac{1}{2}\sigma^2 - \frac{9}{2}\sigma^3 + \frac{3}{2}\sigma^4\right) - c_\beta\left(-3 - \frac{1}{2}\sigma + 8\sigma^2 - \frac{11}{2}\sigma^3 + \sigma^4\right)\right\}\\
d_{-1} &= \frac{1}{D}\{c_\alpha(-22\sigma + 13\sigma^2 + 6\sigma^3 - 3\sigma^4) - c_\beta(12 - 10\sigma - 10\sigma^2 + 10\sigma^3 - 2\sigma^4)\}\\
d_0 &= \frac{1}{D}\left\{c_\alpha\left(11 + \frac{9}{2}\sigma - \frac{19}{2}\sigma^2 - \frac{3}{2}\sigma^3 + \frac{3}{2}\sigma^4\right) - c_\beta\left(-9 + \frac{21}{2}\sigma + 2\sigma^2 - \frac{9}{2}\sigma^3 + \sigma^4\right)\right\}\\
d_1 &= \frac{1}{D}\left\{-18 + 30\sigma - \frac{37}{2}\sigma^2 + 5\sigma^3 - \frac{1}{2}\sigma^4\right\}\\
d_2 &= \frac{1}{D}\{9 - 24\sigma + 22\sigma^2 - 8\sigma^3 + \sigma^4\}\\
d_3 &= \frac{1}{D}\left\{-2 + 6\sigma - \frac{13}{2}\sigma^2 + 3\sigma^3 - \frac{1}{2}\sigma^4\right\}\\
d_A &= \frac{1}{\alpha^+ D}\{11 - 12\sigma + 3\sigma^2\}\\
d_B &= -\frac{1}{\beta^+ D}\{-6 + 11\sigma - 6\sigma^2 + \sigma^3\}
\end{aligned} \tag{118}$$

where

$$D = \frac{1}{6}\{c_\beta(-12 - 14\sigma + 36\sigma^2 - \sigma^3 - 12\sigma^4 + 3\sigma^5) - c_\alpha(22\sigma + 9\sigma^2 - 19\sigma^3 - 3\sigma^4 + 3\sigma^5)\}$$

For the case of Γ located on the left side of i (Fig. 3), the corresponding difference formulas can be derived by a coordinate transformation. The details are not presented here.

A.2. Difference formulas at irregular points with eight-point stencil $(n = m = 4)$

For the case shown in Fig. 2:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = \frac{\sum_{k=-3}^{4} d_k u_{i+k} + d_A A + h d_B B}{(11/3)h} + \frac{11u_i - 18u_{i-1} + 9u_{i-2} - 2u_{i-3}}{6h} + O(h^4) \tag{119}$$

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{\sum_{k=-3}^{4} d_k u_{i+k} + d_A A + h d_B B}{h^2} + \frac{-u_{i-3} + 4u_{i-2} - 5u_{i-1} + 2u_i}{h^2} + O(h^3) \tag{120}$$

The coefficients in the equations above are:

$$\begin{aligned}
d_{-3} &= \frac{1}{D}\left\{c_\alpha\left(\frac{50}{3}\sigma + \frac{5}{3}\sigma^2 - \frac{50}{3}\sigma^3 + 2\sigma^4 + 3\sigma^5 - \frac{2}{3}\sigma^6\right)\right.\\
&\qquad \left. -c_\beta\left(-8 - \frac{22}{3}\sigma + \frac{79}{3}\sigma^2 - \frac{20}{3}\sigma^3 - \frac{47}{6}\sigma^4 + 4\sigma^5 - \frac{1}{2}\sigma^6\right)\right\}\\
d_{-2} &= \frac{1}{D}\left\{c_\alpha(-75\sigma + 5\sigma^2 + 70\sigma^3 - 19\sigma^4 - 7\sigma^5 + 2\sigma^6)\right.\\
&\qquad \left. -c_\beta\left(36 + 21\sigma - \frac{223}{2}\sigma^2 + 50\sigma^3 + 14\sigma^4 - 11\sigma^5 + \frac{3}{2}\sigma^6\right)\right\}\\
d_{-1} &= \frac{1}{D}\left\{c_\alpha(150\sigma - 85\sigma^2 - 60\sigma^3 + 28\sigma^4 + 5\sigma^5 - 2\sigma^6)\right.\\
&\qquad \left. -c_\beta\left(-72 + 30\sigma + 109\sigma^2 - 70\sigma^3 - \frac{11}{2}\sigma^4 + 10\sigma^5 - \frac{3}{2}\sigma^6\right)\right\}\\
d_0 &= \frac{1}{D}\left\{c_\alpha\left(-50 - \frac{65}{3}\sigma + \frac{145}{3}\sigma^2 + \frac{32}{3}\sigma^3 - 11\sigma^4 - \sigma^5 + \frac{2}{3}\sigma^6\right)\right.\\
&\qquad \left. -c_\beta\left(44 - \frac{131}{3}\sigma - \frac{143}{6}\sigma^2 + \frac{80}{3}\sigma^3 - \frac{2}{3}\sigma^4 - 3\sigma^5 + \frac{1}{2}\sigma^6\right)\right\}\\
d_1 &= \frac{1}{D}\left\{96 - 208\sigma + \frac{554}{3}\sigma^2 - 86\sigma^3 + \frac{133}{6}\sigma^4 - 3\sigma^5 + \frac{1}{6}\sigma^6\right\}\\
d_2 &= \frac{1}{D}\left\{-72 + 228\sigma - \frac{553}{2}\sigma^2 + 164\sigma^3 - 51\sigma^4 + 8\sigma^5 - \frac{1}{2}\sigma^6\right\}\\
d_3 &= \frac{1}{D}\left\{32 - 112\sigma + 154\sigma^2 - 106\sigma^3 + \frac{77}{2}\sigma^4 - 7\sigma^5 + \frac{1}{2}\sigma^6\right\}\\
d_4 &= \frac{1}{D}\left\{-6 + 22\sigma - \frac{193}{6}\sigma^2 + 24\sigma^3 - \frac{29}{3}\sigma^4 + 2\sigma^5 - \frac{1}{6}\sigma^6\right\}\\
d_A &= \frac{1}{\alpha^+ D}\{-50 + 70\sigma - 30\sigma^2 + 4\sigma^3\}\\
d_B &= -\frac{1}{\beta^+ D}\{24 - 50\sigma + 35\sigma^2 - 10\sigma^3 + \sigma^4\}
\end{aligned} \tag{121}$$

where

$$\begin{aligned}
D = \frac{1}{22}\{&c_\beta(144 + 228\sigma - 458\sigma^2 - 94\sigma^3 + 216\sigma^4 - 18\sigma^5 - 22\sigma^6 + 4\sigma^7) - c_\alpha(-300\sigma - 130\sigma^2 + 290\sigma^3\\
&+ 64\sigma^4 - 66\sigma^5 - 6\sigma^6 + 4\sigma^7)\}
\end{aligned}$$

A.3. Difference formulas at irregular points with 10-point stencil $(n = m = 5)$

For the case shown in Fig. 2:

$$\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)_i = \frac{\sum_{k=-4}^{5} d_k u_{i+k} + d_A A + h d_B B}{\frac{25}{6}h} + \frac{25u_i - 48u_{i-1} + 36u_{i-2} - 16u_{i-3} + 3u_{i-4}}{12h} + O(h^5) \tag{122}$$

$$\left(\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right)_i = \frac{\sum_{k=-4}^{5} d_k u_{i+k} + d_A A + h d_B B}{h^2} + \frac{11u_{i-4} - 56u_{i-3} + 114u_{i-2} - 104u_{i-1} + 35u_i}{12h^2} + O(h^4) \tag{123}$$

The coefficients in the equations above are:

$$\begin{aligned}
d_{-4} &= \frac{1}{D}\left\{ c_\alpha\left(\frac{137}{2}\sigma + \frac{157}{12}\sigma^2 - 74\sigma^3 + \frac{19}{24}\sigma^4 + \frac{75}{4}\sigma^5 - \frac{25}{12}\sigma^6 - \frac{5}{4}\sigma^7 + \frac{5}{24}\sigma^8\right)\right.\\
&\quad \left. -c_\beta\left(-30 - \frac{83}{2}\sigma + \frac{1259}{12}\sigma^2 + \frac{1}{2}\sigma^3 - \frac{587}{12}\sigma^4 + \frac{51}{4}\sigma^5 + \frac{23}{6}\sigma^6 - \frac{7}{4}\sigma^7 + \frac{1}{6}\sigma^8\right)\right\}\\
d_{-3} &= \frac{1}{D}\left\{ c_\alpha\left(-\frac{1096}{3}\sigma - \frac{118}{3}\sigma^2 + \frac{1171}{3}\sigma^3 - \frac{107}{3}\sigma^4 - \frac{535}{6}\sigma^5 + \frac{95}{6}\sigma^6 + \frac{25}{6}\sigma^7 - \frac{5}{6}\sigma^8\right)\right.\\
&\quad \left. -c_\beta\left(160 + \frac{584}{3}\sigma - \frac{1676}{3}\sigma^2 + \frac{173}{3}\sigma^3 + \frac{1369}{6}\sigma^4 - \frac{473}{6}\sigma^5 - \frac{53}{6}\sigma^6 + \frac{13}{2}\sigma^7 - \frac{2}{3}\sigma^8\right)\right\}\\
d_{-2} &= \frac{1}{D}\left\{ c_\alpha\left(822\sigma - \frac{97}{2}\sigma^2 - \frac{1649}{2}\sigma^3 + \frac{799}{4}\sigma^4 + \frac{255}{2}\sigma^5 - \frac{65}{2}\sigma^6 - 5\sigma^7 + \frac{5}{4}\sigma^8\right)\right.\\
&\quad \left. -c_\beta\left(-360 - 318\sigma + 1208\sigma^2 - \frac{717}{2}\sigma^3 - \frac{627}{2}\sigma^4 + \frac{219}{2}\sigma^5 + \frac{9}{2}\sigma^6 - 9\sigma^7 + \sigma^8\right)\right\}\\
d_{-1} &= \frac{1}{D}\left\{ c_\alpha\left(-1096\sigma + \frac{1838}{3}\sigma^2 + \frac{5}{9}\sigma^3 - \frac{707}{3}\sigma^4 - \frac{135}{2}\sigma^5 + \frac{155}{6}\sigma^6 + \frac{5}{2}\sigma^7 - \frac{5}{6}\sigma^8\right)\right.\\
&\quad \left. -c_\beta\left(480 - 56\sigma - \frac{2804}{3}\sigma^2 + 457\sigma^3 + \frac{919}{6}\sigma^4 - \frac{213}{2}\sigma^5 + \frac{13}{6}\sigma^6 + \frac{11}{2}\sigma^7 - \frac{2}{3}\sigma^8\right)\right\}\\
d_0 &= \frac{1}{D}\left\{ c_\alpha\left(274 + \frac{725}{6}\sigma - \frac{3395}{12}\sigma^2 - \frac{425}{6}\sigma^3 + \frac{1819}{24}\sigma^4 + \frac{125}{12}\sigma^5 - \frac{85}{12}\sigma^6 - \frac{5}{12}\sigma^7 + \frac{5}{24}\sigma^8\right)\right.\\
&\quad \left. -c_\beta\left(-250 - \frac{1325}{6}\sigma + \frac{2165}{12}\sigma^2 - \frac{470}{3}\sigma^3 - \frac{227}{12}\sigma^4 + \frac{325}{12}\sigma^5 - \frac{5}{3}\sigma^6 - \frac{5}{4}\sigma^7 + \frac{1}{6}\sigma^8\right)\right\}\\
d_1 &= \frac{1}{D}\left\{-600 + 1540\sigma - \frac{10189}{6}\sigma^2 + \frac{6307}{6}\sigma^3 - \frac{9593}{24}\sigma^4 + \frac{287}{3}\sigma^5 - \frac{169}{12}\sigma^6 + \frac{7}{6}\sigma^7 - \frac{1}{24}\sigma^8\right\} \qquad (124)\\
d_2 &= \frac{1}{D}\left\{600 - 2140\sigma + \frac{18529}{6}\sigma^2 - \frac{7093}{3}\sigma^3 + \frac{6383}{6}\sigma^4 - \frac{874}{3}\sigma^5 + \frac{287}{6}\sigma^6 - \frac{13}{3}\sigma^7 + \frac{1}{6}\sigma^8\right\}\\
d_3 &= \frac{1}{D}\left\{-400 + 1560\sigma - 2501\sigma^2 + 2151\sigma^3 - \frac{4353}{4}\sigma^4 + 333\sigma^5 - 605\sigma^6 + 6\sigma^7 - \frac{1}{4}\sigma^8\right\}\\
d_4 &= \frac{1}{D}\left\{150 - 610\sigma + \frac{6181}{6}\sigma^2 - \frac{2831}{3}\sigma^3 + \frac{3083}{6}\sigma^4 - \frac{512}{3}\sigma^5 + \frac{203}{6}\sigma^6 - \frac{11}{3}\sigma^7 + \frac{1}{6}\sigma^8\right\}\\
d_5 &= \frac{1}{D}\left\{-24 + 100\sigma - \frac{1045}{6}\sigma^2 + \frac{995}{6}\sigma^3 - \frac{2273}{24}\sigma^4 + \frac{100}{3}\sigma^5 - \frac{85}{12}\sigma^6 + \frac{5}{6}\sigma^7 - \frac{1}{24}\sigma^8\right\}\\
d_A &= \frac{1}{\alpha^+ D}\{274 - 450\sigma + 255\sigma^2 - 60\sigma^3 + 5\sigma^4\}\\
d_B &= -\frac{1}{\beta^+ D}\{-120 + 274\sigma - 225\sigma^2 + 85\sigma^3 - 15\sigma^4 + \sigma^5\}
\end{aligned}$$

where

$$\begin{aligned}
D = \frac{1}{100}\{&c_\beta(-2880 - 5424\sigma + 9400\sigma^2 + 3510\sigma^3 - 5125\sigma^4 - 181\sigma^5 + 800\sigma^6 - 70\sigma^7 - 35\sigma^8 + 5\sigma^9)\\
&- c_\alpha(6576\sigma + 2900\sigma^2 - 6790\sigma^3 - 1700\sigma^4 + 1819\sigma^5 + 250\sigma^6 - 170\sigma^7 - 10\sigma^8 + 5\sigma^9)\}
\end{aligned}$$

References

[1] L. Adams, Z. Li, The immersed interface/multigrid methods for interface problems, SIAM J. Sci. Comput. 24 (2002) 463–479.

[2] L. Adams, T.P. Chartier, New geometric immersed interface multigrid solvers, SIAM J. Sci. Comput. 25 (2004) 1516–1533.

[3] A.N. Almgren, J.B. Bell, P. Colella, T. Marthaler, A Cartesian grid projection method for the incompressible Euler equations in complex geometries, SIAM J. Sci. Comput. 18 (1997) 1289–1309.

[4] R.P. Beyer, R.J. LeVeque, Analysis of a one-dimensional model for the immersed boundary method, SIAM J. Numer. Anal. 29 (1992) 332–364.

[5] D. Calhoun, A Cartesian grid method for solving the two-dimensional streamfunction–vorticity equations in irregular regions, J. Comput. Phys. 176 (2002) 231–275.

[6] Y.C. Chang, T.Y. Hou, B. Merriman, S. Osher, A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, J. Comput. Phys. 124 (1996) 449–464.

[7] S. Deng, K. Ito, Z. Li, Three-dimensional elliptic solvers for interface problems and applications, J. Comput. Phys. 184 (2003) 215–243.

[8] M.A. Dumett, J.P. Keener, An immersed interface method for solving anisotropic elliptic boundary value problems in three dimensions, SIAM J. Sci. Comput. 25 (2003) 348–367.

[9] B. Engquist, A.-K. Tornberg, R. Tsai, Discretization of Dirac delta function in level set methods, J. Comput. Phys. 207 (2005) 28–51.

[10] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interface in multimaterial flows (the Ghost Fluid Method), J. Comput. Phys. 152 (1999) 457–492.

[11] Z.C. Feng, L.G. Leal, Nonlinear bubble dynamics, Ann. Rev. Fluid Mech. 29 (1997) 201–243.

[12] A.L. Fogelson, J.P. Keener, Immersed interface methods for Neumann and related problems in two and three dimensions, SIAM J. Sci. Comput. 22 (2000) 1630–1654.

[13] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domain with applications to the Stefan problem, J. Comput. Phys. 202 (2005) 577–601.

[14] J. Glimm et al., A critical analysis of Rayleigh–Taylor growth rates, J. Comput. Phys. 169 (2001) 652–677.

[15] B.E. Griffith, C.S. Peskin, On the order of accuracy of the immersed boundary method: Higher order convergence rates for sufficiently smooth problems, J. Comput. Phys. 208 (2005) 75–105.

[16] B.T. Helenbrook, L. Martinelli, C.K. Law, A numerical method for solving incompressible flow problems with a surface of discontinuity, J. Comput. Phys. 148 (1999) 366–396.

[17] T.Y. Hou, Z. Li, S. Osher, H. Zhao, A hybrid method for moving interface problems with application to the Hele–Shaw flow, J. Comput. Phys. 134 (1997) 236–252.

[18] H. Huang, Z. Li, Convergence analysis of the immersed interface method, IMA J. Numer. Anal. 19 (1999) 583–608.

[19] K. Ito, Z. Li, Solving a nonlinear problem in magneto-rheological fluids using the immersed interface method 19 (2003) 253–266.

[20] H. Johansen, P. Colella, A Cartesian grid embedded boundary method for Poisson's equation on irregular domains, J. Comput. Phys. 147 (1998) 60–85.

[21] M. Kang, R.P. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323–360.

[22] M.-C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2000) 705–719.

[23] L. Lee, R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (2003) 832–856.

[24] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1001–1025.

[25] R. LeVeque, Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput. 18 (1997) 709–735.

[26] Z. Li, Immersed interface methods for moving interface problem, Numer. Algorithms 14 (1997) 269–293.

[27] Z. Li, A fast iterative algorithm for elliptic interface problems, SIAM J. Numer. Anal. 35 (1998) 230–254.

[28] Z. Li, K. Ito, Maximum principle preserving schemes for interface problems with discontinuous coefficients, SIAM J. Sci. Comput. 23 (2001) 339–361.

[29] Z. Li, M.-C. Lai, The immersed interface method for the Navier–Stokes equations with singular forces, J. Comput. Phys. 171 (2001) 822–842.

[30] S.P. Lin, R.D. Reitz, Drop and spray formation from a liquid jet, Ann. Rev. Fluid Mech. 30 (1998) 85–105.

[31] M.N. Linnick, F.F. Hermann, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, J. Comput. Phys. 204 (2004) 157–192.

[32] B. Lombard, J. Piraux, Numerical treatment of two-dimensional interfaces for acoustic and elastic waves, J. Comput. Phys. 195 (2004) 90–116.

[33] P. McCorquodale, P. Colella, H. Johansen, A Cartesian grid embedded boundary method for the heat equation on irregular domains, J. Comput. Phys. 173 (2001) 620–635.

[34] A. Mayo, The fast solution of Poisson's and biharmonic equations on irregular regions, SIAM J. Numer. Anal. 21 (1984) 285–299.

[35] S. Osher, R. Fedkiw, Level set methods: an overview and some recent results, J. Comput. Phys. 169 (2001) 463–502.

[36] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer-Verlag, New York, 2003.

[37] C. Peskin, The immersed boundary method, Acta Numer. (2002) 479–517.

[38] J. Piraux, B. Lombard, A new interface method for hyperbolic problems with discontinuous coefficients: one-dimensional acoustic example, J. Comput. Phys. 168 (2001) 227–248.

[39] C. Pozrikidis, Interfacial dynamics for Stokes flow, J. Comput. Phys. 169 (2001) 250–301.

[40] D. Russell, Z.J. Wang, A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow, J. Comput. Phys. 191 (2003) 177–205.

[41] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, Ann. Rev. Fluid Mech. 31 (1999) 567–603.

[42] J.A. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, Cambridge, UK, 1999.

[43] J.A. Sethian, Evolution, implementation, and application of level set and fast marching methods for advancing fronts, J. Comput. Phys. 169 (2001) 503–555.

[44] H.A. Stone, Dynamics of drop deformation and breakup in viscous fluids, Ann. Rev. Mech. 26 (1994) 65–102.

[45] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–154.

[46] M. Sussman, A second-order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles, J. Comput. Phys. 187 (2003) 110–136.

[47] M. Tateneni, X. Zhong, Numerical study of two-phase flows in microchannels using the level set method, AIAA 2004, paper 2004-0929.

[48] M. Tateneni, X. Zhong, Numerical simulations of two-phase flows in micro gas/liquid sections, AIAA 2005, paper 2005-1392.

[49] A.K. Tornberg, B. Engquist, Numerical approximations of singular source terms in differential equations 200 (2004) 462–488.

[50] G. Tryggvasson, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front-tracking method for the computations of multiphase flow, J. Comput. Phys. 169 (2001) 708–759.

[51] W. Tsai, D.K. Yue, Computation of nonlinear free-surface flows, Ann. Rev. Fluid Mech. 28 (1996) 249–278.

[52] H.S. Udaykumar, R. Mittal, P. Rampunggoon, A. Khanna, A sharp interface Cartesian grid method for simulating flows with complex moving boundaries, J. Comput. Phys. 174 (2001) 345–380.

[53] H.S. Udaykumar, R. Mittal, W. Shyy, Computation of solid–liquid phase fronts in the sharp interface limit on fixed grids, J. Comput. Phys. 153 (1999) 534–574.

[54] H.S. Udaykumar, W. Shyy, M.M. Rao, Elafint: a mixed Eulerian–Lagrangian method for fluid flows with complex and moving boundaries, Int. J. Numer. Methods Fluids 22 (1996) 691–705.

[55] A. Wiegmann, K.P. Bube, The immersed interface method for nonlinear differential equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 35 (1998) 177–200.

[56] 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 (2000) 827–862.

[57] Y.C. Zhou, S. Zhao, M. Feig, G.W. Wei, High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources, J. Comput. Phys. 213 (2006) 1–30.