SCIPEDIA Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería

Revista Internacional Métodos numéricos para cálculo y diseño en ingeniería RIMNI

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH In cooperation with CIMNE

A new sixth-order immersed interface method for solving Poisson equations with straight interfaces

R. Itza Balam[12], M. Uh Zapata[12], J. Montalvo-Urquizo[3]

1 Consejo Nacional de Humanidades, Ciencias y Tecnologías, CONAHCYT, México

2 Centro de Investigación en Matemáticas A. C., CIMAT Unidad Mérida, México

3 School of Science and Engineering, Tecnológico de Monterrey, México

Abstract

This paper introduces a sixth-order Immersed Interface Method (IIM) for addressing 2D Poisson problems characterized by a discontinuous forcing function with straight interfaces. In the presence of this discontinuity, the problem exhibits a non-smooth solution at the interface that divides the domain into two regions. Here, the IIM is employed to compute the solution on a fixed Cartesian grid. This method integrates necessary jump conditions resulting from the interface into the numerical schemes. In order to achieve a sixth-order method, the proposed approach combines implicit finite differences with the IIM. The proposed scheme is efficient because the matrix arising from discretization remains the same as in the smooth problem, and changes are made to the resulting linear system by introducing new terms on the right side. These supplementary terms account for the discontinuities in the solution and its derivatives, with calculations restricted near the interface. The paper demonstrates the accuracy of the proposed method through various numerical examples.

OPEN ACCESS

Published: 28/12/2023

Accepted: 18/12/2023

Submitted: 05/09/2023

DOI: 10.23967/j.rimni.2023.12.002

Keywords:
Poisson equation
immersed interface method
finite difference
sixth-order of accuracy
implicit finite difference 1 Introduction

1 Introduction

Developing advanced algorithms for solving the Poisson equation holds great significance in numerous research domains, including computational fluid dynamics, wave propagation, and theoretical physics [1]. The discontinuous problem emerges in scenarios marked by abrupt changes at interfaces that separate different regions within a given domain [2,3,4]. In this article, we particularly focus on linear interfaces, which are commonly encountered in layered phenomena (for example, as seen in [5,6,7] and the cited references). Additionally, the pursuit of high-order methods for addressing such challenges is highly advantageous since their enhanced precision permits the use of coarser grids, subsequently reducing computational expenses.

This paper presents high-order finite-difference schemes up to sixth-order for the Poisson equation for straight interfaces. The problem is given by [2,3,4].

$$\Delta u(x) = f(x), \quad x \in \Omega \setminus \Gamma, \quad \Omega = \Omega^+ \cup \Omega^- \tag{1}$$

$$[u]_\Gamma = p(x), \quad x \in \Gamma, \tag{2}$$

$$[u_n]_\Gamma = q(x), \quad x \in \Gamma. \tag{3}$$

where u and f are the solution and known right-hand side function, respectively. We divide Ω in two regions, Ω^+ and Ω^-, separated by an immersed interface Γ. The computational domain can be one-dimensional (1D) (with several interface points), or a two-dimensional (2D) region with straight interfaces. We use Dirichlet boundary conditions on $\partial\Omega$. We assume that the solution, the right-hand side function, and their derivatives may have discontinuities at Γ. Thus, we require jump conditions as additional inputs. The principal jump conditions $[u] = p$ and $[u_n] = q$ are known functions and are defined on Γ. Here, u_n is the derivative in the normal direction.

Many numerical methods have been proposed to solve accurately the Poisson equation; however most of these methods are limited to smooth solutions. For instance, many developments have been made to get fourth- and sixth-order finite-difference methods for the Poisson equation (1), see for example [8,9,10,11,12,13,14]. On the other hand, to overcome the interface issue, several approaches exist to approximate discontinuous solutions, such as the level set method [15], immersed boundary method [16], ghost fluid method [17], interpolation matched interface method [18], Galerkin finite element method [19], boundary condition capturing [20], and interface neural networks [21]. However, there are not many available high-order discretizations of the Poisson problems with interfaces and many of these algorithms are only second-order accuracy. From these methods, the immersed interface method [22,23,24,4,25] is one popular option to solve (1)-(3) accurately by simple modifications of standard finite differences.

There only exist a few high-order immersed interface methods to solve elliptic equations. Fourth-order interface methods for elliptic equations with discontinuous solutions or discontinuous

Correspondence: R. Itza Balam (reymundo.itza@cimat.mx), M. Uh Zapata (angeluh@cimat.mx), J. Montalvo-Urquizo (jmontalvo@tec.mx).

coefficients are investigated in [26,27,28,29,30]. Lately, Feng and Li [31] presented a third-order IIM for elliptic interface problems, but it is limited to straight interfaces lying at grid points. Pan et al. [32] proposed a third-order IIM to solve elliptic problems on irregular domains, and Colnago et al. [33] developed a fourth-order approximation. More recently, Feng et al. [34] presented a sixth-order IIM to solve Poisson interface problem with singular sources based on the undetermined coefficients technique.

This paper presents a new high-order implicit finite-difference immersed interface method (HIFD-IIM) up to sixth-order accuracy to solve the 2D Poisson problem (1)-(3) with straight interfaces. The implicit finite-difference methodology is based on calculating the unknown variable and its corresponding derivatives simultaneously [35,36]. The system is changed by adding new terms on the right side, named jump contributions. These terms include the jumps of the solution and its derivatives, and they are calculated near the interface. In this context, the straight interfaces allows to calculate the jumps contributions directly from principal jump conditions (2) and (3) without any other calculation. The modifications are only performed at grid points where the method's stencil intersects the interface. Moreover, the matrix in the linear system is the same as the smooth problem. This formulation makes the method attractive as it is easy to implement and does not require other convergence restrictions than the ones from standard methods for smooth solutions. To the best authors' knowledge, there is not other methods using the proposed formulation.

The paper is organized as follows. Section 2 introduces the implicit formulation. The main theoretical results are presented in this section. Section 3 deals with the 1D Poisson problem. Section 4 shows how to implement the high order methods for 2D problems with straight interfaces. Section 5 contains several examples to test the algorithm's capacity. Finally, in Section 6, we present the conclusions and future work.

2 Implicit finite-difference formulation

We begin our analysis by considering the 1D finite-difference (FD) scheme for the second derivative of a real-valued function u. In this work, the numerical solution is approximated using a uniform grid. An interval $[a, b]$ is divided into N sub-intervals, using $x_i = a + (i-1)h$, $i = 1, 2, ..., N, N+1$, where the grid size is given by $h = (b-a)/N$. For simplicity, we set $x = x_\alpha$ between two consecutive grid points $x_I \le x_\alpha < x_{I+1}$. We called x_I and x_{I+1} irregular points; meanwhile the reminder points are referred as regular. Besides h, the discretization needs the definitions of $h_L = x_I - x_\alpha$, and $h_R = x_{I+1} - x_\alpha$. Note that h_R is a positive value, meanwhile h_L is negative, see Fig. 1.

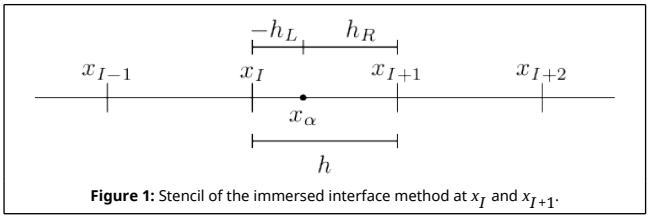

Figure 1: Stencil of the immersed interface method at x_I and x_{I+1}.

Finally, we denote as $\varphi_i = \varphi\,|_i = \varphi(x_i)$ the evaluation of function φ at the ith-point of the grid.

2.1 Approximation on regular points

It is well-known that the central finite difference gives us a second-order approximation, and can be written as

$$u_{xx}\,|_i = \delta^2 u\,|_i + O(h^2), \tag{4}$$

where

$$\delta^2 u\,|_i := \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2}, \tag{5}$$

for a given small value $h > 0$.

One way to increase the order of accuracy in (4) is using a FD operator with more grid points. However, there is another way to get high-order approximations without changing the length of the FD operator (5). It is by considering a high-order implicit finite-difference (HIFD) formulation, as presented in [13]. Thus, using Taylor series expansions, it directly follows that the sixth-order formula is given by

$$\partial^4 u_{xx}\,|_i = \delta^2 u\,|_i + O(h^6), \qquad i \neq I, I+1. \tag{6}$$

where the FD operator δ^2 is given by (5) and the partial operator is defined as

$$\partial^4(\,\cdot\,) := (\,\cdot\,) + b\,h^2(\,\cdot\,)_{xx} + e\,h^4(\,\cdot\,)_{xxxx}, \tag{7}$$

$b = 1/12$ and $e = 1/360$. The above formulation (6) turns in a fourth-order method if we choose $b = 1/12$ and $e = 0$. Moreover, if we select $b = 0$ and $e = 0$ the standard second-order approximation is recovered.

We remark that (6) can only be applied in cases where the problem's solution has enough regularity, see [37]. To overcome this issue, we combine the HIFD with the IIM to solve problems with discontinuous solutions, as described in the next section.

2.2 Approximation on irregular points

This paper proposes a new formulation named HIFD-IIM that is based on the combination of HIFD and IIM. To derive high-order schemes, the IIM requires additional conditions at $i = I, I+1$. These are known as jump conditions at x_α. Thus, for a function φ, they are given by

$$[\varphi] = \varphi^+ - \varphi^-, \qquad \varphi^+ = \varphi(x_\alpha^+) = \lim_{x \to x_\alpha^+} \varphi(x), \qquad \varphi^- = \varphi(x_\alpha^-) = \lim_{x \to x_\alpha^-} \varphi(x). \tag{8}$$

The IIM contribution requires to include high-order jump derivatives. However, we can reduce the number of jumps by applying a less accurate scheme at the irregular points. As other IIMs proposed by different authors [22,24,38,39,40], the global order is $O(h^2)$, even if the local truncation error at $i = I$ and $i = I+1$ is one order lower, i.e., $O(h)$ [41]. Recently, Pan et al. [32] proposed a global third-order IIM using $O(h^3)$ and $O(h^2)$ local truncation errors at regular and irregular grid points, respectively. Following similar ideas, the main result of this paper is presented in Theorem 1.

Theorem 1: HIFD-IIM. Let us consider the known jump conditions

$$[u],\quad [u_x],\quad [u_{xx}],\quad [u_{xxx}],\quad [u_{xxxx}],\quad [u_{xxxxx}],\quad [u_{xxxxxx}], \tag{9}$$

on x_α such that $x_I \le x_\alpha < x_{I+1}$. Then u_{xx} can be approximated at x_I and x_{I+1} by the finite-difference scheme

$$\partial^4 u_{xx}\,|_i = \delta^2 u\,|_i + C_i + O(h^5), \quad i = I, I+1, \tag{10}$$

where

$$C_I := -\frac{1}{h^2}\{[u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{3!}[u_{xxx}] + \frac{h_R^4}{4!}[u_{xxxx}] + \frac{h_R^5}{5!}[u_{xxxxx}] + \frac{h_R^6}{6!}[u_{xxxxxx}]\}, \tag{11}$$

$$C_{I+1} := \frac{1}{h^2}\{[u] + h_L[u_x] + \frac{h_L^2}{2}[u_{xx}] + \frac{h_L^2}{3!}[u_{xxx}] + \frac{h_L^4}{4!}[u_{xxxx}] + \frac{h_L^6}{5!}[u_{xxxxx}] + \frac{h_L^6}{6!}[u_{xxxxxx}]\}. \tag{12}$$

We obtain a fifth-order scheme for at x_I and x_{I+1} following similar ideas as the generalized Taylor expansion proposed by Xu & Wang [38] and the IIM for elliptic interface problems with straight interfaces proposed by Feng & Li [39].

We initially consider extended solutions of u, named u_ℓ and u_r, as shown in Fig. 2. The idea is to have smooth functions such that we can apply the standard central scheme to x_I and x_{I+1}. These functions are defined as

$$u_\ell(x) = \begin{cases} u(x), & x < x_\alpha \\ u^- + h_R u_{\bar{x}} + \frac{h_R^2}{2}u_{\bar{x}x} + \frac{h_R^3}{3!}u_{\bar{x}xx} + \frac{h_R^4}{4!}u_{\bar{x}xxx} + \frac{h_R^5}{5!}u_{\bar{x}xxxx} + \frac{h_R^6}{6!}u_{\bar{x}xxxxx}, & x_\alpha < x, \end{cases} \tag{13}$$

$$u_r(x) = \begin{cases} u^+ + h_L u_x^+ + \frac{h_L^2}{2}u_{xx}^+ + \frac{h_L^3}{3!}u_{xxx}^+ + \frac{h_L^4}{4!}u_{xxxx}^+ + \frac{h_L^5}{5!}u_{xxxxx}^+ + \frac{h_L^6}{6!}u_x^+, & x < x_\alpha \\ u(x), & x_\alpha \le x, \end{cases} \tag{14}$$

where u^- and u^+ are defined as in equation (8).

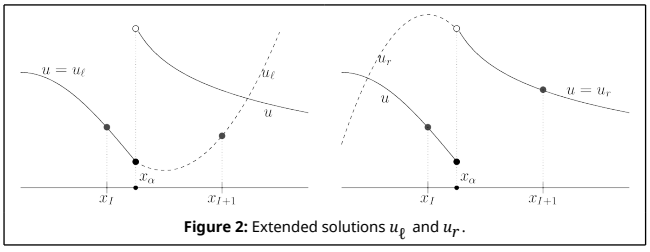

Figure 2: Extended solutions u_ℓ and u_r.

Taylor series expansions of u_{I+1} around x_α yields

$$u_{I+1} = u^+ + h_R u_x^+ + \frac{h_R^2}{2}u_{xx}^+ + \frac{h_R^3}{3!}u_{xxx}^+ + \frac{h_R^4}{4!}u_{xxxx}^+ + \frac{h_R^5}{5!}u_{xxxxx}^+ + \frac{h_R^6}{6!}u_{xxxxxx}^+ + O(h^7).$$

Using the definition of jumps (9), it follows

$$u_{I+1} = (u^- + h_R u_{\bar{x}} + \frac{h_R^2}{2}u_{\bar{x}x} + \frac{h_R^3}{3!}u_{\bar{x}xx} + \frac{h_R^4}{4!}u_{\bar{x}xxx} + \frac{h_R^5}{5!}u_{\bar{x}xxxx} + \frac{h_R^6}{6!}u_{\bar{x}xxxxx}) + [u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{3!}[u_{xxx}] + \frac{h_R^4}{4!}[u_{xxxx}] + \frac{h_R^5}{5!}[u_{xxxxx}] + \frac{h_R^6}{6!}[u_{xxxxxx}] + O(h^7).$$

Next, we use u_ℓ defintion in (13) to obtain

$$u_{I+1} = u_{\ell\, I+1} + [u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{3!}[u_{xxx}] + \frac{h_R^4}{4!}[u_{xxxx}] + \frac{h_R^5}{5!}[u_{xxxxx}] + \frac{h_R^6}{6!}[u_{xxxxxx}] + O(h^7).$$

Thus,

$$u_{I+1} = u_{\ell\, I+1} - h^2 C_I + O(h^7), \tag{15}$$

where C_I is defined as equation (11). Substituting (15) into a standard central scheme for the second-order derivative, it yields

$$\delta^2 u\,|_I = \frac{1}{h^2}u_{I+1} - \frac{2}{h^2}u_I + \frac{1}{h^2}u_{I-1} = \frac{1}{h^2}u_{\ell I+1} - \frac{2}{h^2}u_{\ell I} + \frac{1}{h^2}u_{\ell I-1} - C_I + O(h^5).$$

On the other hand, using Taylor series of $(u_\ell)_{xx}$, we have

$$(u_\ell)_{xx}\,|_I = \frac{1}{h^2}u_{\ell I+1} - \frac{2}{h^2}u_{\ell I} + \frac{1}{h^2}u_{\ell I-1} - bh^2(u_\ell)_{xxxx} - eh^4(u_\ell)_{xxxxxx} + O(h^6).$$

Thus,

$$\delta^2 u\,|_I = (u_\ell)_{xx}\,|_I + bh^2(u_\ell)_{xxxx}\,|_I + eh^4(u_\ell)_{xxxxxx}\,|_I - C_I + O(h^5).$$

Finally, we get $\partial^4 u_{xx}\,|_I = \delta_x^2 u\,|_I + C_I + O(h^5)$. This completes the proof. The same procedure can be applied for the proof at x_{I+1} using definition (14).

It is important to remark that C_I and C_{I+1} are constants computed from the jump derivatives of u and we assume that those values are known. To emphasize that the contribution includes all jump derivatives up to sixth-order we write C_I^6 instead C_I.

Remark 1: We can rewrite the contributions (11) and (12) as follows

$$C_I^6 := -\frac{1}{h^2}\{[u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}] + b\left(2h_R^3[u_{xxx}] + \frac{h_R^4}{2}[u_{xxxx}]\right) + e\left(3h_R^5[u_{xxxxx}] + \frac{h_R^6}{2}[u_{xxxxxx}]\right)\}, \quad (16)$$

$$C_{I+1}^6 := \frac{1}{h^2}\{[u] + h_L[u_x] + \frac{h_L^2}{2}[u_{xx}] + b\left(2h_L^3[u_{xxx}] + \frac{h_L^4}{2}[u_{xxxx}]\right) + e\left(3h_L^5[u_{xxxxx}] + \frac{h_L^6}{2}[u_{xxxxxx}]\right)\}, \quad (17)$$

where b and e are defined as in (7).

Remark 2: If the solution is smooth, then all contributions C_I^6 and C_{I+1}^6 are equal to zero and the standard sixth-order method [37] is recovered in equation (6).

Corollary 1: If we consider $b = 1/12$ and $e = 0$ in (1) and (17), then we obtain a third-order scheme for $i = I, I+1$ (global fourth-order method) as follows

$$u_{xx}|_i + bh^2(u_{xx})_{xx}|_i = \delta^2 u|_i + C_i^4 + O(h^3), \quad i = I, I+1, \quad (18)$$

where

$$C_I^4 = -\frac{1}{h^2}\{[u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{3!}[u_{xxx}] + \frac{h_R^4}{4!}[u_{xxxx}]\}, \quad (19)$$

$$C_{I+1}^4 = \frac{1}{h^2}\{[u] + h_L[u_x] + \frac{h_L^2}{2}[u_{xx}] + \frac{h_L^3}{3!}[u_{xxx}] + \frac{h_L^4}{4!}[u_{xxxx}]\}. \quad (20)$$

Corollary 2: If $b = e = 0$ in (1) and (17), the method represents an explicit finite-difference scheme of first-order of accuracy for $i = I, I+1$ (global second-order method) as follows

$$u_{xx}|_i = \delta^2 u|_i + C_i^2 + O(h), \quad i = I, I+1, \quad (21)$$

where

$$C_I^2 = -\frac{1}{h^2}\{[u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}]\}, \quad (22)$$

$$C_{I+1}^2 = \frac{1}{h^2}\{[u] + h_L[u_x] + \frac{h_L^2}{2}[u_{xx}]\}. \quad (23)$$

In this case, we only require to explicitly know jump conditions $[u]$, $[u_x]$, and $[u_{xx}]$.

Observe that previous corollaries include the superscript 2 and 4 to emphasize contributions upto second- and fourth-order derivatives, respectively.

3 One-dimensional problem

In this section, we use u_{xx} approximation from Theorem 1 to study the 1D Poisson problem given by

$$u_{xx}(x) = f(x), \quad x \in \Omega = (a, x_\alpha) \cup (x_\alpha, b), \quad (24)$$

where u and f can be discontinuous functions at a given point $x = x_\alpha$, and the principal jump conditions $[u] = p$, $[u_x] = q$ are known values at x_α. For the boundary conditions, we impose the Dirichlet type. Although this technique can be applied to several interface points, we only focus on one point $x_\alpha \in [x_I, x_{I+1})$ to simplify our exposition.

If we apply the partial operator (7) at both sides of (24), then we get

$$\partial^4 u_{xx}|_i = \partial^4 f|_i, \qquad i = 2, \ldots, I, I+1, \ldots, N. \quad (25)$$

Substituting equations (6) and (10) into the left hand-side of equation (25), we get

$$\partial^4 u_{xx}|_i = \delta^2 u|_i + C_i^6\{u\}, \qquad i = 2, \ldots, I, I+1, \ldots, N, \quad (26)$$

where C_i^6 corresponds to the contribution term of u at x_i given by

$$C_i^6\{u\} = \begin{cases} -\frac{1}{h^2}\{[u] + h_R[u_x] + \frac{h_R^2}{2}[u_{xx}] + \frac{h_R^3}{3!}[u_{xxx}] + \frac{h_R^4}{4!}[u_{xxxx}] + \frac{h_R^5}{5!}[u_{xxxxx}] + \frac{h_R^6}{6!}[u_{xxxxxx}]\}, & i = I, \\ \frac{1}{h^2}\{[u] + h_L[u_x] + \frac{h_L^2}{2}[u_{xx}] + \frac{h_L^2}{3!}[u_{xxx}] + \frac{h_L^4}{4!}[u_{xxxx}] + \frac{h_L^6}{5!}[u_{xxxxx}] + \frac{h_L^6}{6!}[u_{xxxxxx}]\}, & i = I+1, \\ 0, & \text{erwise}. \end{cases} \quad (27)$$

Here, we introduce the notation $C_i^6\{u\}$ to emphasize that the contribution depends on the high-order jump derivatives and it is computed from the function u. It is necessary because in next section we will require to obtain contributions from different functions.

If we explicitly know the function f and its derivatives, then the right-hand side of (25) can be calculated as:

$$\partial^4 f|_i = f_i + bh^2 f_{xx}|_i + eh^4 f_{xxxx}|_i.$$

On the other hand, if we only know values of f on the grid, we have to approximate the second- and fourth-order derivative of f at x_i. As with other implicit schemes [13,36], the right-hand side derivatives are calculated using a central finite-difference method. The discretization of the right-hand side for $i \neq I, I+1$ is given by

$$\partial^4 f|_i = f_i + bh^2 f_{xx}|_i + eh^4 f_{xxxx}|_i = df_{i-2} + (b - 4d)f_{i-1} + (1 - 2b + 6d)f_i + (b - 4d)f_{i+1} + df_{i+2} + O(h^5),$$

where $d =: e - b^2 = -1/240$. For $i = I, I+1$, we need to compute the derivatives using the IIM technique which is described as follows. Notice that the second derivative of f in $bh^2 f_{xx}$ requires a discretization of third-order accuracy to obtain a local error of $O(h^5)$ because it is already multiplied by a factor h^2. Thus, using (1), we obtain

$$f_{xx}\,|_i = \delta^2 f\,|_i - bh^2 f_{xxxx}\,|_i + C_i^4\{f\} + O(h^3). \tag{28}$$

On the other hand, the fourth-order derivative of f in $eh^4 f_{xxxx}$ only requires an approximation of the first-order accuracy because its coefficient includes the term h^4; thus, we still have a local $O(h^5)$ to keep a global sixth-order accurate method. Now applying (1) and (2) we get

$$\begin{aligned} f_{xxxx}\,|_i &= \delta^2 f_{xx}\,|_i + C_i^2\{f_{xx}\} + O(h) \\ &= \delta^2\left(\left[\delta^2 f - bh^2\partial_{xx} f_{xx}\,|_i + C_i^4\{f\} + O(h^3)\right) + C_i^2\{f_{xx}\} + O(h), \end{aligned}$$

then,

$$f_{xxxx}\,|_i = \delta^4 f\,|_i + \delta^2 C_i^4\{f\} + C_i^2\{f_{xx}\} + O(h). \tag{29}$$

where

$$\delta^4 f = \frac{1}{h^4}\left(f_{i+2} - 4f_{i+1} + 6f_i - 4f_{i-1} + f_{i-2}\right).$$

Note that now the contribution notation includes from which function f or f_{xx} it comes. We also remark that term $bh^2\delta^2 f_{xxxx}$ will be small if we guarantee $\delta^2 f_{xxxx} \approx f_{xxxxxx}$ is bounded, which is always true in our analysis. Now, using identities (28) and (29), we obtain the discretization of the right-hand side as follows

$$\begin{aligned} f_i + bh^2 f_{xx}\,|_i + eh^4 f_{xxxx}\,|_i = {} & df_{i-2} + (b - 4d)f_{i-1} + (1 - 2b + \\ & 6d)f_i + (b - 4d)f_{i+1} + df_{i+2} \\ & + bh^2 C_i^4\{f\} + dh^4\left(C_i^2\{f_{xx}\} + \delta^2 C_i^4\{f\}\right) + O(h^5). \end{aligned} \tag{30}$$

Finally, the HIFD-IIM for the 1D Poisson equation (24) at x_i is given by

$$\begin{aligned} \delta^2 u\,|_i = {} & df_{i-2} + (b - 4d)f_{i-1} + (1 - 2b + 6d)f_i + (b - \\ & 4d)f_{i+1} + ef_{i+2} + C + O(h^5), \end{aligned} \tag{31}$$

where $C = (C_f)_i - C_i^6\{u\}$ and

$$(C_f)_i := bh^2 C_i^4\{f\} + dh^4 C_i^2\{f_{xx}\} + dh^2\left(C_{i+1}^4\{f\} - 2C_i^4\{f\} + C_{i-1}^4\{f\}\right),$$

for $i = 2, \ldots, N$. Note that Dirichlet boundary conditions are directly applied at $i = 1$ and $i = N + 1$.

Using definitions (19), (20), (22), and (23), we can simplify contribution $(C_f)_i$ as

$$(C_f)_i = \begin{cases} -d\left([f] + h_R[f_x] + \frac{h_R^2}{2}[f_{xx}] + \frac{h_R^3}{3!}[f_{xxx}] + \frac{h_R^2}{4!}[f_{xxxx}]\right) \\ \quad -\left(a_0[f] + r_1[f_x] + r_2[f_{xx}] + r_3[f_{xxx}] + r_4[f_{xxxx}]\right) \\ a_0[f] + l_1[f_x] + l_2[f_{xx}] + l_3[f_{xxx}] + l_4[f_{xxxx}], \\ d\left([f] + h_L[f_x] + \frac{h_L^2}{2}[f_{xx}] + \frac{h_L^3}{3!}[f_{xxx}] + \frac{h_L^4}{4!}[f_{xxxx}]\right) \\ 0 \end{cases} \begin{matrix} i = \\ \\ i = \\ i = \\ \text{oth} \end{matrix} \tag{32}$$

where $a_0 = b - d$, and

$$\begin{aligned} r_1 &= bh_R + d(h_L - 2h_R), & l_1 &= bh_L + d(h_R - 2h_L), \\ r_2 &= \tfrac{1}{2}\left(bh_R^2 + d(h_L^2 - 2h_R^2 + 2h^2)\right), & l_2 &= \tfrac{1}{2}\left(bh_L^2 + d(h_R^2 - 2h_L^2 + 2h^2)\right), \\ r_3 &= \tfrac{1}{3!}\left(bh_R^3 + d(h_L^3 - 2h_R^3 + 6h^2 h_R)\right), & l_3 &= \tfrac{1}{3!}\left(bh_L^3 + d(h_R^3 - 2h_L^3 + 6h^2 h_L)\right), \\ r_4 &= \tfrac{1}{4!}\left(bh_R^4 + d(h_L^4 - 2h_R^4 + 24h^2 h_R^2)\right), & l_4 &= \tfrac{1}{4!}\left(bh_L^4 + d(h_R^4 - 2h_L^4 + 24h^2 h_L^2)\right). \end{aligned}$$

3.1 1D HIFD-IIM and the principal jump conditions

Note that $[u]$, $[u_x]$, $[u_{xx}]$, $[u_{xxx}]$, $[u_{xxxx}]$, $[u_{xxxxx}]$ and $[u_{xxxxxx}]$ must be known to apply the proposed sixth-order HIFD-IIM. Thus, it seems that more jump conditions of u rather than the principal jump conditions are required to have a sixth-order accurate method. However, we can use the Poisson equation (24) to obtain relations between u, f and their derivatives as follows

$$[u_{xx}] = [f], \quad [u_{xxx}] = [f_x], \quad [u_{xxxx}] = [f_{xx}], \quad [u_{xxxxx}] = [f_{xxx}], \quad [u_{xxxxxx}] = [f_{xxxx}].$$

Thus, the total jump contribution for the 1D problem, C, is given by

$$C = \begin{cases} -d\left([f] + h_R[f_x] + \frac{h_R^2}{2}[f_{xx}] + \frac{h_R^3}{3!}[f_{xxx}] + \frac{h_R^2}{4!}[f_{xxxx}]\right), & i = \\ \frac{1}{h^2}\left\{[u] + h_R[u_x]\right\} - \left(R_0[f] + R_1[f_x] + R_2[f_{xx}] + R_3[f_{xxx}] + R_4[f_{xxxx}]\right), & i \\ -\frac{1}{h^2}\left\{[u] + h_L[u_x]\right\} + L_0[f] + L_1[f_x] + L_2[f_{xx}] + L_3[f_{xxx}] + L_4[f_{xxxx}], & i = \\ d\left([f] + h_L[f_x] + \frac{h_L^2}{2}[f_{xx}] + \frac{h_L^3}{3!}[f_{xxx}] + \frac{h_L^4}{4!}[f_{xxxx}]\right), & i = \\ 0, & \text{oth} \end{cases}$$

where

$$R_0 = a_0 - \frac{h_R^2}{2h^2}, \quad R_k = r_k - \frac{h_R^{k+2}}{(k+2)!h^2}, \quad L_0 = a_0 - \frac{h_L^2}{2h^2}, \quad L_k = l_k - \frac{h_L^{k+2}}{(k+2)!h^2}.$$

for $k = 1, 2, 3, 4$. Thus, the contribution $C = (C_f)_i - C_i^6\{u\}$ depends only on the principal jump conditions and right-hand side jumps $[f]$, $[f_x]$, $[f_{xx}]$, $[f_{xxx}]$, and $[f_{xxxx}]$.

Remark 3: For the 1D Poisson problem, a second-order IIM ($b = 0, d = 0$) only requires knowing the principal jump conditions $[u]$, $[u_x]$, and $[f]$. On the other hand, a fourth-order IFD-IIM ($d = 0$) is obtained knowing the principal jump conditions $[u]$, $[u_x]$, $[f]$, and additionally $[f_x]$ and $[f_{xx}]$. However, the extra jumps are from the right-hand side, which is already known analytically or can be approximated using the current values of f. In this paper, we will assume that we know them.

Remark 4: We can achieve sixth or fourth-order approximation in some particular grids even if we do not know high-order derivative jumps. For example, for the sixth-order HIFD-IIM, if $h_R/h = 1$, then $h_L = 0$, and both weight terms next to the fourth-order derivative jump of f are equal to zero. Thus, we do not require to know the jump condition $[f_{xxxx}]$ to obtain a sixth-order method when the interface is located at a grid point. Similarly, we can get a fourth-order scheme even if we do not know jump condition $[f_{xx}]$ when the interface is located at a grid point.

4 Two-dimensional problem

We now apply the methodology to 2D problems and straight interfaces. We study the 2D Poisson problem given by

$$u_{xx}(x,y) + u_{yy}(x,y) = f(x,y), \quad (x,y) \in \Omega = (a,x_\alpha) \cup (x_\alpha, b) \times (c,d), \tag{33}$$

where u and f can be discontinuous functions at $x = x_\alpha$, and the principal jump conditions are known functions in the variable y and specified as follows

$$[u] = p(x_\alpha,y), \quad \text{and} \quad [u_n] = q(x_\alpha,y), \qquad c \le y \le d\,. \tag{34}$$

The numerical domain is discretized using an uniform mesh, $h = (b-a)/N$ (N sub-intervals in the x-direction) and assuming that $d - c = Mh$ (M sub-intervals in the y-direction). We denote as $u_{ij} = u(x_i,y_j)$ and $f_{ij} = f(x_i,y_j)$ where the ijth point is given by

$$(x_i,y_j) = (a + (i-1)h, c + (j-1)h), \qquad i = 1,\ldots,N,N+1, j = 1,\ldots,M,M+1.$$

The grid points are also classified into two types: regular and irregular. If the straight interface, $x = x_\alpha$, intersects the FD stencil surrounding the ijth point, the center point is called irregular; otherwise, the grid point is regular, see Fig. 3.

Finally, let us define the discrete operators δ_x^2 and δ_x^4 at the ijth point as

$$\delta_x^2 u(x_i,y_j) = \frac{u_{i+1j} - 2u_{ij} + u_{i-1j}}{h^2},$$
$$\delta_x^4 u(x_i,y_j) = \delta_x^2\delta_x^2 u(x_i,y_j) = \frac{u_{i+2j} - 4u_{i+1j} + 6u_{ij} - 4u_{i-1j} + u_{i-2j}}{h^4}.$$

Formulas δ_y^2 and δ_y^4 are defined similarly.

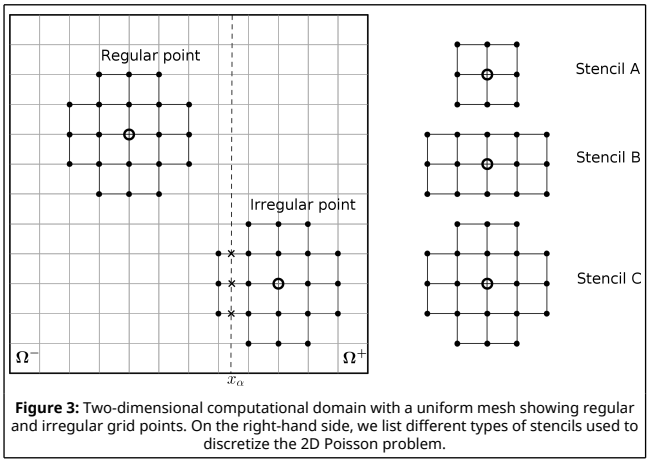

Figure 3: Two-dimensional computational domain with a uniform mesh showing regular and irregular grid points. On the right-hand side, we list different types of stencils used to discretize the 2D Poisson problem.

We develop the discretization around the ijth point. However, to simplify our exposition we drop the evaluation at ijth point. We start the discretization by applying the HIFD operator (7) in x- and y-direction to the Poisson equation (33), as

$$\partial_y^4 u_{xx} + \partial_y^4 u_{yy} = \partial_y^4 f, \quad \text{and} \quad \partial_x^4 u_{xx} + \partial_x^4 u_{yy} = \partial_x^4 f, \tag{35}$$

where

$$\partial_x^4(\cdot) := (\cdot) + bh^2(\cdot)_{xx} + eh^4(\cdot)_{xxxx}, \quad \text{and} \quad \partial_y^4(\cdot) := (\cdot) + bh^2(\cdot)_{yy} + eh^4(\cdot)_{yyyy}\,. \tag{36}$$

Adding both equations in (35),

$$\partial_x^4 u_{xx} + \partial_y^4 u_{yy} + \partial_y^4 u_{xx} + \partial_x^4 u_{yy} = \partial_x^4 f + \partial_y^4 f\,. \tag{37}$$

In the following sections, we will describe the discretization of (37) for regular and irregular grid points.

4.1 The HIFD-IIM at regular points

Using one-dimensional formula (6), equations (36) and some algebraic simplifications, the sixth-order implicit method applied on the regular grid points is given by

$$\delta_x^2 u + \delta_y^2 u + bh^2\left(\delta_y^2\delta_x^2 u + \delta_x^2\delta_y^2 u\right) + (e - 2b^2)h^4\left(\delta_x^4\delta_y^2 u + \delta_y^4\delta_x^2 u\right) = f + bh^2(f_{xx} + f_{yy}) + eh^4(f_{xxxx} + f_{yyyy}) + O(h^6)\,. \tag{38}$$

The implicit methods have not only high accuracy, but they are also more efficient in terms of the number of iterations required to solve the linear system of the discretization [13]. In the next section we work with the irregular grind points.

4.2 The HIFD-IIM at irregular points

As the interface is a vertical line, the left-hand side terms without cross derivatives of (37) can be approximated using the HIFD-IIM formulation as follows

$$\partial_x^4 u_{xx} = \delta_x^2 u + C^6\{u\} + O(h^5), \quad \text{and} \quad \partial_y^4 u_{yy} = \delta_y^2 u + O(h^6)\,. \tag{39}$$

The left-hand side terms with cross derivatives of (37) require more attention. Using the Poisson equation and Theorem 1, we can show that the following equation holds

$$\begin{aligned}\partial_y^4 u_{xx} + \partial_x^4 u_{yy} &= f + bh^2\left(\delta_y^2\delta_x^2 u + \delta_x^2\delta_y^2 u\right) + (e - 2b^2)h^4\left(\delta_x^4\delta_y^2 u + \delta_y^4\delta_x^2 u\right)\\ &+ bh^2\delta_y^2 C^6\{u\} + bh^2\left(C^4\{u_{yy}\} + bh^2C^2\{u_{yyyy}\}\right)\\ &+ (e - 2b^2)h^4\left(\delta_x^2 C^4\{u_{yy}\} + bh^2\delta_x^2 C^2\{u_{yyyy}\} + C^2\{u_{yyxx}\} + \delta_y^4 C^6\{u\}\right) + O(h^5)\,.\end{aligned} \tag{40}$$

On the other hand, using the definition of operator ∂^4 given by (36), the right-hand side can be written as follows

$$\partial_x^4 f + \partial_y^4 f = 2f + bh^2(f_{xx} + f_{yy}) + eh^4(f_{xxxx} + f_{yyyy})\,. \tag{41}$$

Substituting equations (39)-(41) into (37) we obtain

$$\delta_x^2 u + \delta_y^2 u + bh^2\left(\delta_y^2\delta_x^2 u + \delta_x^2\delta_y^2 u\right) + (e - 2b^2)h^4\left(\delta_x^4\delta_y^2 u + \delta_y^4\delta_x^2 u\right) = f + bh^2(f_{xx} + f_{yy}) + eh^4(f_{xxxx} + f_{yyyy}) + C_u + O(h^5), \tag{42}$$

where

$$C_u = -C^6\{u\} - bh^2\delta_y^2 C^6\{u\} - bh^2\left(C^4\{u_{yy}\} + bh^2C^2\{u_{yyyy}\}\right) - (e-2b^2)h^4\left(\delta_x^2C^4\{u_{yy}\} + bh^2\delta_x^2C^2\{u_{yyyy}\} + C^2\{u_{yyxx}\} + \delta_y^4C^6\{u\}\right) + O(h^5). \quad (43)$$

All above terms are evaluated at the ijth point (omitted to avoid misunderstandings). If we known explicitly f, then the discretization is complete. However, there are problems which the Poisson implementation only allows to known the right-hand side at the discretization points. In the following section, we focus on this case.

4.3 Right-hand side approximation

It is possible that f is only know at the grid points. If this is the case, we only require to compute terms f_{xx} and f_{xxxx} using HIFD-IIM because the interface is a vertical line, see Fig. 3. Moreover, formulas (28) and (29) are valid in this case. Then,

$$f + bh^2(f_{xx}+f_{yy}) + eh^4(f_{xxxx}+f_{yyyy}) = f + bh^2\left(\delta_x^2f + \delta_y^2f\right) + dh^4\delta_x^4f + eh^4\delta_y^4f + bh^2C^4\{f\} + dh^4\left(\delta_x^2C^4\{f\} + C^2\{f_{xx}\}\right) + O(h^5), \quad (44)$$

where $d = e - b^2$. Finally, putting together equations (42) and (44), we obtain the *fully discretized 2D problem* as follows

$$\delta_x^2u + \delta_y^2u + bh^2\left(\delta_y^2\delta_x^2u + \delta_x^2\delta_y^2u\right) + (e-2b^2)h^4\left(\delta_x^4\delta_y^2u + \delta_y^4\delta_x^2u\right) = f + bh^2\left(\delta_x^2f + \delta_y^2f\right) + dh^4\delta_x^4f + eh^4\delta_y^4f + C_f + C_u + O(h^5), \quad (45)$$

where $C_f = bh^2C^4\{f\} + dh^4\left(\delta_x^2C^4\{f\} + C^2\{f_{xx}\}\right)$.

If the 2D Poisson problem admits a smooth solution, then all contribution terms in (45) are equal to zero, and a sixth-order method is recovered. In the presence of an interface, the contributions C^2, C^4, and C^6 on regular grid points are always zero. This scheme requires a stencil type C, see Fig. 3

Remark 5: Taking $e = 0$ into equations (36) and only considering approximations up to four-order accurate, we obtain the IFD-IIM

$$\delta_x^2u + \delta_y^2u + bh^2\left(\delta_y^2\delta_x^2u + \delta_x^2\delta_y^2u\right) = f + bh^2\left(\delta_x^2f + \delta_y^2f\right) + \hat{C}_f + \hat{C}_u + O(h^3), \quad (46)$$

where $\hat{C}_f = C^4\{f\}$ and $\hat{C}_u = -C^4\{u\} - bh^2\delta_y^2C^4\{u\} - bh^2C^2\{u_{yy}\}$. In addition, if the 2D Poisson problem admits a smooth solution, all contribution terms in (5) vanish, and we get a fourth-order implcit scheme. In fact, the contributions C^k are always zero on regular grid points. This scheme requires a stencil type A, see Fig. 3. Furthermore, if we consider that $b = 0$ in (5) then we obtain the standard explicit immersed interface method, called IIM, which is second order accurate.

Remark 6: It is possible to develop a discrete formulation for the 2D Poisson problem when the straight interface is horizontal. The resulting scheme is

$$\delta_x^2u + \delta_y^2u + bh^2\left(\delta_y^2\delta_x^2u + \delta_x^2\delta_y^2u\right) + (e-2b^2)h^4\left(\delta_x^4\delta_y^2u + \delta_y^4\delta_x^2u\right) = f + bh^2\left(\delta_x^2f + \delta_y^2f\right) + dh^4\delta_x^4f + eh^4\delta_y^4f + C + O(h^5),$$

where

$$C = bh^2C^4\{f\} + dh^4\left(\delta_y^2C^4\{f\} + C^2\{f_{yy}\}\right) - C^6\{u\} - bh^2\delta_x^2C^6\{u\} - bh^2\left(C^4\{u_{xx}\} + bh^2C^2\{u_{xxxx}\}\right) - (e-2b^2)h^4\left(\delta_y^2C^4\{u_{xx}\} + bh^2\delta_x^2C^2\{u_{xxxx}\} + C^2\{u_{xxyy}\} + \delta_x^4C^6\{u\}\right).$$

The constants C^2, C^4, and C^6 are computed using jump derivatives in the y-direction.

4.4 Boundary treatment

The grid points close to the interface require special treatment because the HIFD-IIM formulation of the 21-point stencil C cannot be applied there. For regular grid points, we use the following fourth-order method

$$\delta_x^2u + \delta_y^2u + bh^2\left(\delta_y^2\delta_x^2u + \delta_x^2\delta_y^2u\right) = f + bh^2\left(\delta_x^2f + \delta_y^2f\right) + O(h^4).$$

The deduction of this scheme, stencil type A, can be found in [13].

It is necessary to develop a new fourth-order method for the grid points near both boundary and interface. Following the same ideas to develop the sixth-order method, we obtain a new scheme with stencil of type B

$$\delta_x^2u + \delta_y^2u + bh^2\left(\delta_x^2\delta_y^2u + \delta_y^2\delta_x^2u\right) + (e-2b^2)h^4\delta_x^4\delta_y^2u = f + bh^2\left(\delta_x^2f + \delta_y^2f\right) + h^4\delta_x^4f - b^2h^4\delta_y^4f + C_{boundary} + O(h^4),$$

where

$$C_{boundary} = bh^2C^4\{f\} + dh^4\left(\delta_x^2C^4\{f\} + C^2\{f_{xx}\}\right) - C^6\{u\} - bh^2\delta_y^2C^6\{u\} - bh^2\left(C^4\{u_{yy}\} + bh^2C^2\{u_{yyyy}\}\right) - (e-2b^2)h^4\left(\delta_x^2C^4\{u_{yy}\} + bh^2\delta_x^2C^2\{u_{yyyy}\} + C^2\{u_{yyxx}\}\right) + O(h^4).$$

Remark 7: We emphasized that the proposed methods are high-order accuracy regardless of the position of the interface concerning the grid. Moreover, the scheme does not assure restrictions in the jumps, such as the natural jump conditions $[u_x] = 0$, $[u_y] = 0$). This characteristic is a significant advantage of the proposed HIFD-IIM, besides the higher-order, compared with the fourth-order simplified immersed interface method developed by Feng et al.[39].

5 Numerical results

This section tests the HIFD-IIM for different one- and two-dimensional examples with straight interfaces. In the following simulations, we numerically solve the Poisson equation for a given right-hand side function and compare it with its analytic solution. We present different examples to test the HIFD-IIM capabilities. First, we investigate the method's accuracy for one-dimensional problems. Next, we validate the HIFD-IIM for two-dimensional solutions with straight interfaces.

The errors are reported utilizing the L_∞-norm, as $\| e \|_\infty = \max_{i,j} | u_{ij} - U_{ij} |$, where u_{ij} and U_{ij} corresponds to the exact and numerical solution at (x_i, y_j), respectively. The estimated order of accuracy is computed as

$$\text{Order} := \log\left(\| e \|_{N_1} / \| e \|_{N_2} \right) / \log(N_2/N_1),$$

where N_1 and N_2 indicates the different number of sub-intervals. In all tables, the last row shows the numerical order calculated by the regression-line slope based on a Least Squares Method (LSM).

5.1 One-dimensional examples

We initially consider the 1D problem (24) to analyze the order of the proposed implicit methods. The numerical method is tested using three different tests. Example 1.A was designed to verify the high-order implicit method for smooth solutions. Example 1.B studies a Poisson equation with a discontinuous solution in a single interface point. Example 1.C presents a discontinuous problem with multiple interface points. Thus, the proposed solution is taken from the following list of functions:

$$\text{Example1.A:} \quad u(x) = e^{-4\pi(x-1/4)^2}. \tag{47}$$

$$\text{Example1.B:} \quad u(x) = \begin{cases} \sin(\pi x) & x \le x_\alpha, \\ \cos(\pi x) & x > x_\alpha. \end{cases} \tag{48}$$

$$\text{Example1.C:} \quad u(x) = \begin{cases} \sin(2\pi x) & x \le x_{\alpha_1}, \\ \cos(2\pi x) & x_{\alpha_1} < x \le x_{\alpha_2}, \\ \sin(2\pi x) & x_{\alpha_2} < x. \end{cases} \tag{49}$$

where x_α, x_{α_1} and x_{α_2} are known values corresponding to the interface location. The right-hand side function, f, is obtained directly from (33) and u. For all cases, we impose the Dirichlet boundary conditions according to u. The computational domain is the interval $[0, 1]$, and the grid spacing is $h = 1/N$ for different N numbers.

For Example 1.A, due to the solution's regularity, the jump contributions are equal to zero. Table 1 presents the convergence analysis of Example 1.A for different grid resolutions. Note that the IIM, IFD-IIM, and HIFD-IIM achieve their corresponding order of accuracy. These results match with the ones obtained using an implicit methodology as presented in [13].

Table. 1 Convergence analysis of Example 1.A for IFD-IIM and HIFD-IIM.

N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	7.52e-02	--	9.30e-03	--	1.39e-04	--
20	1.70e-02	2.15	5.05e-04	4.20	1.52e-06	6.51
40	4.15e-03	2.03	3.06e-05	4.04	1.77e-08	6.43
80	1.03e-03	2.01	1.90e-06	4.01	2.34e-10	6.24
160	2.57e-04	2.00	1.18e-07	4.00	3.24e-12	6.18
LSM		**2.04**		**4.06**		**6.34**

Example 1.B shows the capacity of the proposed method to solve a single interface problem located at $x = x_\alpha$. We test two different interface points: $x_\alpha = 0.40$ and $x_\alpha = 0.63$. We initially select the mesh grid given by $N = 10 \times 2^n$, thus the first interface is always located on a grid point ($h_R/h = 1$). For the second case, we have different h_R/h values for the same N numbers. Fig. 4 shows the numerical and exact solution using $N = 40$. As expected, the exact solution is accurately recovered for both cases.

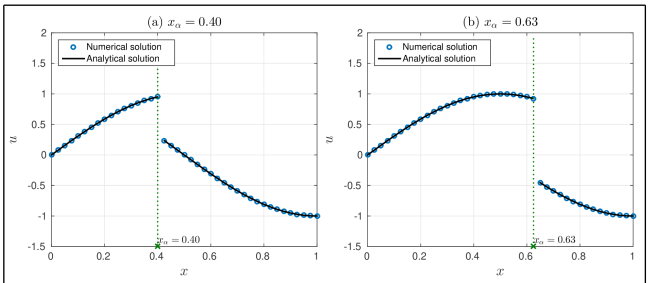

Figure 4: Numerical and exact solution of Example 1.B using N=40 using (a) x_α=0.4, and (b) x_α=0.63.

Table 2 shows the convergence analysis for Example 1.B. As expected, the desired order of accuracy are obtained for the two x_α values. Observe that high-order methods do not depend on the location of the interface. However, their error magnitude presents minor variations due to the interface position. Errors with mesh size close to $N = 160$ have a random behavior due the effect of arithmetic operations close to the machine precision. Fig. 5 shows the error analysis corresponding to interface locations $x_\alpha = 0.40$ and $x_\alpha = 0.63$ for $N = 10, 11, 12, \ldots, 100$. Note that errors of IFD-IIM give a good behavior even if the h_R/h varies and they are close to fourth order. As expected, errors of HIFD-IIM are also near sixth order.

Table. 2 Convergence analysis of Example 1.B using the IFD-IIM and HIFD-IIM.

N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	1.69e-02	--	5.75e-05	--	6.42e-07	--
20	4.21e-03	2.00	3.58e-06	4.00	1.01e-08	6.42
40	1.05e-03	2.00	2.24e-07	4.00	1.59e-10	6.21
80	2.63e-04	2.00	1.40e-08	4.00	2.48e-12	6.11
160	6.57e-05	2.00	8.74e-10	4.00	4.55e-14	5.82
LSM		**2.00**		**4.00**		**5.95**
N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	6.63e-03	--	3.42e-05	--	6.38e-07	--
20	1.48e-03	2.16	1.97e-06	4.12	9.89e-09	6.44
40	4.46e-04	1.73	1.39e-07	3.82	1.58e-10	6.18
80	9.37e-05	2.25	7.75e-09	4.16	2.40e-12	6.15
160	2.75e-05	1.77	5.37e-10	3.85	7.31e-14	5.08
LSM		**1.98**		**3.99**		**5.81**

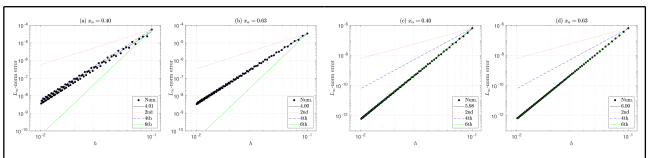

Figure 5: Convergence analysis of Example 1.B for N=10,...,100 using (a),(b) IFD-IIM and (c),(d) HIFD-IIM for x_α=0.40 and x_α=0.63.

We remark that, the contribution formula includes jumps $[u]$, $[u_x]$, $[u_{xx}]$, $[u_{xxx}]$, and $[u_{xxxx}]$ to obtain a fourth-order accurate method. Fig. 6 shows that if we add additional jumps of high-order derivatives to C, such as $[u_{xxxxx}]$, we observe that the error oscillation decreases compared to Fig. 5 results. It is expected because now the method is $O(h^4)$ for the whole computational domain, including the irregular points. Thus, we can mitigate error oscillations due to interface position by adding high-order jumps. A similar behavior is observed for the sixth-order HIFD-IIM if we include the seventh derivative jump

$[u_{xxxxxxx}]$.

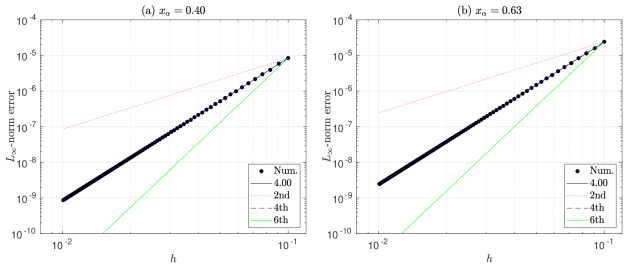

Figure 6: Convergence analysis of Example 1.B for N=10,...,100 using (a) x_α=0.40, and (b) x_α=0.63. The contribution term includes jumps up to fifth-order ($[u_{xxxxx}]$=$[f_{xxx}]$).

Finally, Example 1.C investigates the method's capacity to solve a multiple interface problem. We only focus on two interface points located at $x_{\alpha_1} = 0.30$ and $x_{\alpha_2} = 0.70$. However, the methodology could be applied for several interfaces by doing minor modifications in the implementation. Fig. 7 presents the analytical and numerical solution using $N = 40$. This figure also shows the corresponding absolute error. Fig. 8 shows the error analysis for high-order IIM using different grid resolutions $N = 10, 20, 80, 160$. As expected, the IFD-IIM and HIFD-IIM are fourth- and sixth-order accurate methods, respectively.

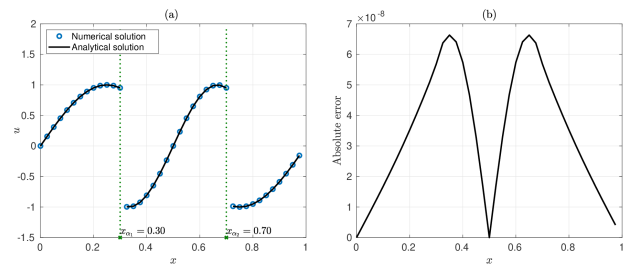

Figure 7: (a) Numerical and exact solution of Example 1.C with multiple interfaces using N=40; (b) Absolute error of the numerical solution using the HIFD-IIM.

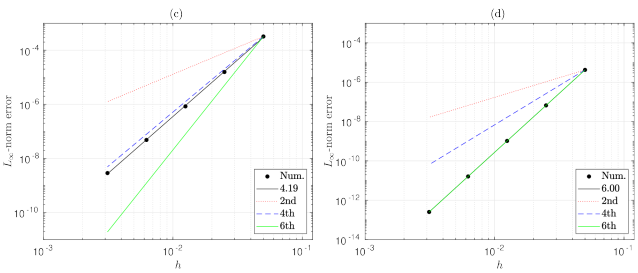

Figure 8: Convergence error analysis of (c) the IFD-IIM, and (d) HIFDM-IIM, using different grid resolutions N=10,20,40,80,160.

5.2 Two-dimensional examples

In this section we study the method's capacities to solve two-dimensional problems with different jump-contribution characteristics. Here we use straight interfaces located at $x = x_\alpha$. First, we consider smooth problem to verify the accurate implementation of the implicit method. Next, we analyze several discontinuous problems and finally, we include a more complex test where the jump derivatives increase rapidly.

For all 2D examples, the right-hand side function and jump conditions are computed from the corresponding exact solution, the computational domain is $\Omega = [-1, 1] \times [-1, 1]$ and the grid size in both directions is the same using N = 10, 20, 40, 80, 160. We impose Dirichlet boundary conditions. This paper uses the Successive-Over Relaxation (SOR) method with $\epsilon = 10^{-14}$ and $\omega = 1.9$ as tolerance and relaxation parameters, respectively.

5.2.1 2D Poisson equation with smooth solution

In the first 2D example, we solve the Poisson equation (33) for a smooth solution to show the correct implementation of the high-order implicit methods. In this case, the exact solution is given by

$$\text{Example2.A:} \quad u(x,y) = \sin(3x)\sin(3y). \tag{50}$$

Table 3 shows the convergence analysis of Example 2.A using different grid resolutions for the fourth- and sixth-order implicit formulation. As expected for a smooth solution, the implicit formulation improves the precision of the standard second-order numerical solution.

Table. 3 Convergence analysis of Example 2.A.

N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	2.65e-02	--	3.07e-04	--	4.64e-05	--
20	7.06e-03	1.91	2.10e-05	3.87	7.54e-07	5.94
40	1.76e-03	2.00	1.32e-06	3.99	1.17e-08	6.01
80	4.40e-04	2.00	8.24e-08	4.00	1.82e-10	6.00
160	1.10e-04	2.00	5.16e-09	4.00	2.95e-12	5.95
LSM		**1.98**		**3.97**		**5.98**

5.2.2 2D Poisson equation with straight interfaces

In this section, we solve the 2D Poisson problem using the following set of functions:

$$\text{Example2.B:} \quad u(x,y) = \begin{cases} \sin(x)e^y & x \le x_\alpha, \\ -x^2e^y + 2 & x_\alpha < x, \end{cases} \quad \text{where } x_\alpha = 0.$$

$$\text{Example2.C:} \quad u(x,y) = \begin{cases} \sin(x)\sin(y) & x \le x_\alpha, \\ e^x\sin(3y) + 1 & x_\alpha < x, \end{cases} \quad \text{where } x_\alpha = 0.4.$$

$$\text{Example2.D:} \quad u(x,y) = \begin{cases} \cos(3x)\cos(3y) & x \le x_\alpha, \\ \sin(3y)\sin(3y) & x_\alpha < x, \end{cases} \quad \text{where} x_\alpha = 0.5.$$

Examples 2.B, 2.C and 2.D investigate the influence on the absolute error and the accuracy over several assumptions of the jump derivatives. Example 2.B analyzes the solution where jump derivatives in the y-direction vanish at x_α. In Example 2.C, the jump derivative in the y-direction changes slower for $x < x_\alpha$ than the one for $x > x_\alpha$. Example 2.D studies the problem without any assumption about the jump derivatives. Note that the interface location x_α is also different for each example.

Fig. 9 shows the numerical solution of these examples using a grid resolution of N = 80. As expected, the HIFD-IIM solves the problem accurately for each case.

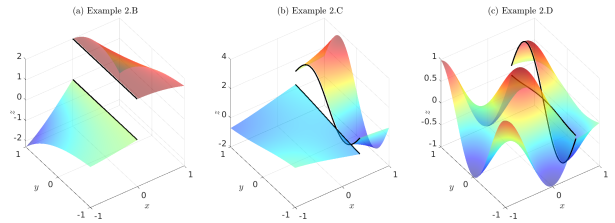

Figure 9: Numerical solution of 2D Poisson equation with a straight interface corresponding to Examples 2.B, 2.C, and 2.D using N=80.

Table 4 shows the convergence analysis for Example 3 using the implicit methods. As expected, the IFD-IIM and HIFD-IIM schemes are close to fourth- and sixth-order, respectively. This table also confirms the standard IIM is second-order accurate. Observe that the order corresponding to the HIFD-IIM with N = 160 in Example 2.B is reduced due to the arithmetic operations close to the machine precision. Fig. 10 shows more details of the convergence analysis using a cloud of points from N = 10 to N = 160.

Table. 4 Convergence analysis of Examples 2.B, 2.C, and 2.D using a straight interface.

N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	5.76e-04	--	5.06e-06	--	2.10e-06	--
20	1.50e-04	1.94	3.19e-07	3.99	3.78e-08	5.79
40	3.77e-05	1.99	2.00e-08	3.99	6.34e-10	5.90
80	9.45e-06	2.00	1.25e-09	4.00	1.03e-11	5.95
160	2.36e-06	2.00	7.84e-11	4.00	1.17e-12	3.13
LSM		**1.99**		**4.00**		**5.34**
N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	2.65e-02	--	8.40e-04	--	3.01e-04	--
20	6.60e-03	2.00	6.58e-05	3.67	6.74e-06	5.48
40	1.61e-03	2.04	4.52e-06	3.86	1.25e-07	5.75
80	3.98e-04	2.01	2.96e-07	3.93	2.14e-09	5.87
160	9.90e-05	2.01	1.90e-08	3.97	3.50e-11	5.93
LSM		**2.02**		**3.87**		**5.77**
N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	4.62e-02	--	5.46e-04	--	2.62e-04	--
20	1.13e-02	2.03	3.94e-05	3.79	2.91e-06	6.49
40	2.81e-03	2.01	2.63e-06	3.91	5.40e-08	5.75
80	7.04e-04	2.00	1.69e-07	3.96	9.12e-10	5.89
160	1.76e-04	2.00	1.07e-08	3.98	1.48e-11	5.95
LSM		**2.01**		**3.91**		**5.98**

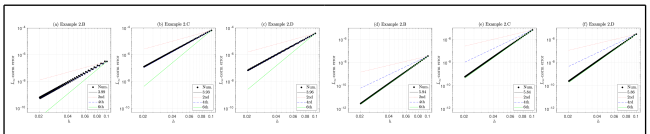

Figure 10: Convergence analysis of Examples 2.B, 2.C, and 2.D using the (a)-(c) IFD-IIM and (d)-(f) HIFD-IIM from N=10 to 100.

5.2.3 2D Poisson solution with large jump derivatives

Finally, we construct the following example to analyze the variations in the errors due to jump derivative magnitudes. Thus, we have

$$\text{Example 2.E: } u(x,y) = \begin{cases} 0 & x \le x_\alpha, \\ 2 + \ln(x(1+y^2)) & x_\alpha < x, \end{cases} \quad \text{where } x_\alpha = 0.5.$$

Here, the jump derivatives for all points at the interface are $[u_x] = 2$, $[u_{xx}] = -4$, $[u_{xxx}] = 16$, $[u_{xxxx}] = -96$, $[u_{xxxxx}] = 768$, and $[u_{xxxxxx}] = -7680$. Is important to remark that opposite to the previous examples, the jump derivatives increase rapidly. This behavior makes the problem challenging to solve.

Table 5 and Fig.11 show the convergence analysis for Example 2.E. Numerical results show that Example 2.E has more variability in error than previous examples. However, the order of each technique is close to the proposed one. These findings also confirm that local truncation error depends not only on h_R and h_L but also on the jump magnitudes.

Table. 5 Convergence analysis of Example 2.E.

N	L_∞-norm	Order	L_∞-norm	Order	L_∞-norm	Order
10	1.11e-02	--	2.72e-04	--	1.38e-04	--
20	1.80e-03	2.62	5.61e-05	2.28	2.48e-06	5.80
40	5.64e-04	1.67	3.62e-06	3.96	9.43e-08	4.72
80	1.57e-04	1.85	2.29e-07	3.98	2.61e-09	5.17
160	4.11e-05	1.93	1.43e-08	3.99	6.10e-11	5.42
LSM		**1.97**		**3.64**		**5.21**

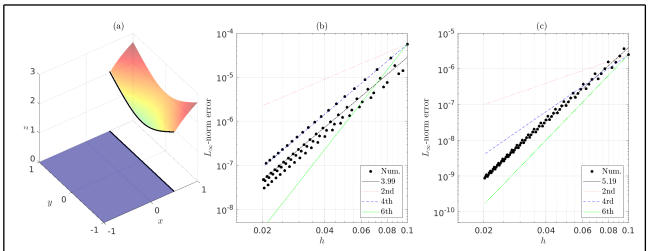

Figure 11: Numerical and exact solution of Example 2.E using N=80; (b),(c) convergence error analysis of IFD-IIM and HIFD-IIM respectively, using different grid resolutions from N=10 to 100.

6 Conclusions

The present paper introduces a new sixth-order immersed interface combined with an implicit finite difference to solve 2D Poisson problems with straight interfaces. The resulting numerical method is $O(h^6)$ at regular points, and $O(h^5)$ at irregular points. Furthermore, a fourth-order immersed interface method is obtained as a particular case of the proposed scheme. This paper also presents a numerical technique to handle the boundaries in the Poisson problem. The global accuracy of the sixth-order was demonstrated using several numerical examples. As expected, this approach does not depend on the interface position. For future work, the proposed approximation will be used to solve more general elliptic equations and interface shapes, and time-dependent problems in higher dimensions.

References

[1] H. J. Diersch, Fletcher, C.A.J. *Computational Techniques for Fluid Dynamics. Vol. I: Fundamental and General Techniques. Vol. II:*

Specific Techniques for Different Flow Categories. Springer-Verlag, 1998.

[2] E. Javierre, C. Vuik, F.J. Vermolen, S. Van der Zwaag, A comparison of numerical models for one-dimensional Stefan problems, J. Comput. Appl. Math. 192 (2) (2006) 445–459.

[3] Y.E. Shi, R.K. Ray, K.D. Nguyen, A projection method-based model with the exact C-property for shallow-water flows over dry and irregular bottom using unstructured finite-volume technique, Comput. Fluids 76 (2013) 178–195.

[4] Li, Z. & Ito, K. *The Immersed Interface Method: Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*, SIAM: Frontiers in Applied Mathematics, 2006.

[5] Z. Li, D. McTigue, and J. Heine, A numerical method for diffusive transport with moving boundaries and discontinuous material properties, Int J Numer Anal Meth Geomech 21 (1997), 653–662.

[6] Z. Li and J. Zou, Theoretical and numerical analysis on a thermo-elastic system with discontinuities, J Comput Appl Math 91 (1998), 1–22.

[7] S. Tsynkov, G. Baruch, G. Fibich, & E. Turkel, Fourth order schemes for time-harmonic wave equations with discontinuous coefficients, Commun Comput Phys 5 (2009), 442–455.

[8] Li M., Fornberg B. & Tang T. A compact fourth order finite difference scheme for the steady incompressible Navier-Stokes equations, *Int. J. Numer. Methods Fluids* **20** (1995), 1137–1151.

[9] Zhang, J. Multigrid method and fourth-order compact scheme for 2D Poisson equation with unequal mesh-size discretization. *J. Comput. Phys.* **179(1)** (2002), 170–179.

[10] Nabavi, M., Siddiqui, M.H.K., & Dargahi J. A new 9-point sixth-order accurate compact finite difference method for the Helmholtz equation, *J. Sound Vib.* **307** (2007), 972–982.

[11] Wang Y., & Zhang J. Sixth order compact scheme combined with multigrid method and extrapolation technique for 2D poisson equation, *J. Comput. Phys.* **228** (2009), 137–146.

[12] Zhai, S., Feng, X., & He, Y. A new method to deduce high-order compact difference schemes for two-dimensional Poisson equation. *Applied Mathematics and Computation* **230** (2014), 9–26.

[13] Uh Zapata, M., & Itzá Balam, R. High-order implicit finite difference schemes for the two-dimensional Poisson equation. *Applied Mathematics and Computation* **309** (2017), 222–244.

[14] Itzá Balam, R., & Uh Zapata, M. A new eighth-order implicit finite difference method to solve the three-dimensional Helmholtz equation. *Computers & Mathematics with Applications* **80(5)** (2020), 1176–1200.

[15] Sethian, J. A. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Geometry, Fluid Mechanics*, Computer Vision, and Materials Sciences, Cambridge University Press, 1999.

[16] Peskin, C. S. The immersed boundary method, *Acta Numer.* **11** (2002), 479–517.

[17] Liu, X. D., & Soderis, T. C. Convergence of the ghost fluid method for elliptic equations with interfaces, *J. Math. Comp.* **72** (2003), 1731–1746.

[18] Hu, H., Pan, K., & Tan, Y. An interpolation matched interface and boundary method for elliptic interface problems, *J. Comput. Appl. Math.* **234** (2010), 73–94.

[19] Mu, L., Wang, J., Ye, X., & Zhao, S. A new weak Galerkin finite element method for elliptic interface problems, *J. Comput. Phys.* **325** (2016), 157–173.

[20] Cho, H., Han, H., Lee, B., Ha, Y., & Kang, M. A second-order boundary condition capturing method for solving the elliptic interface problems on irregular domains, *Journal of Scientific Computing* **81(3)** (2019), 217–251.

[21] Itzá Balam, R., Hernandez-Lopez, F., Trejo-Sánchez, J., & Uh Zapata, M. An immersed boundary neural network for solving elliptic equations with singular forces on arbitrary domains. *Mathematical Biosciences and Engineering: MBE* **18(1)** (2020), 22–56.

[22] Leveque R.J., & Li Z. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, *SIAM J. Numer. Anal.* **31(4)** (1994), 1019-1044.

[23] Wiegmann A., & Bube K.P. The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions, *SIAM J. Numer. Anal.* **37(3)** (2000), 827–862.

[24] Berthelsen, P. A. (2004). A decomposed immersed interface method for variable coefficient elliptic equations with non-smooth and discontinuous solutions. J. Comput. Phys. 197(1), 364–386.

[25] Seo, J. H. & Mittal, R. A high-order immersed boundary method for acoustic wave scattering and low-Mach number flow-induced sound in complex geometries, *J. Comput. Phys.* **230** (2011), 1000–1019.

[26] Ito, K., Li, Z., & Kyei, Y. Higher-order, Cartesian grid based finite difference schemes for elliptic equations on irregular domains. *SIAM Journal on Scientific Computing* **27(1)** (2005), 346-367.

[27] Gibou, F., & Fedkiw, R. A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem. *J. Comput. Phys.* **202(2)** (2005), 577-601.

[28] Linnick, M. N., & Fasel, H. F. A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains. *J. Comput. Phys.* **204(1)** (2005), 157-192.

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

[30] Zhong, X. A new high-order immersed interface method for solving elliptic equations with imbedded interface of discontinuity. *J. Comput. Phys.* **225(1)** (2007), 1066-1099.

[31] Feng, X., Li, Z., & Qiao, Z. High order compact finite difference schemes for the Helmholtz equation with discontinuous coefficients. *J. of Comput. Math.* (2011) 324–340.

[32] Pan, K., He, D., & Li, Z. A high order compact FD framework for elliptic BVPs involving singular sources, interfaces, and irregular domains. *J. Sci. Comput.* **88(3)** (2021), 1–25.

[33] Colnago, M., Casaca, W., & de Souza, L. F. A high-order immersed interface method free of derivative jump conditions for Poisson equations on irregular domains. *J. Comput. Phys.* **423** (2020), 109791.

[34] Feng, Q., Han, B., & Minev, P. Sixth order compact finite difference schemes for Poisson interface problems with singular sources. *Computers & Mathematics with Applications* **99** (2021), 2-25.

[35] Claerbout, J.F. The craft of wave-field extrapolation, in Imaging the Earth's Interior, *Blackwell Scientific Publications, Oxford* (1985), 260-265.

[36] Liu, Y., & Sen, M. K. A practical implicit finite-difference method: examples from seismic modeling. *Journal of Geophysics and Engineering* **6(3)** (2009), 231.

[37] W.F. Spotz, High-order compact finite difference schemes for computational mechanics Doctoral dissertation, The University of Texas at Austin, 1995.

[38] Xu S., & Wang Z.J. Systematic Derivation of Jump Conditions for the Immersed Interface Method in Three-Dimensional Flow Simulation, *J. Sci. Comput.* **27(6)** (2006), 1948-1980.

[39] Feng, X., & Li, Z. Simplified immersed interface methods for elliptic interface problems with straight interfaces. *Num. Meth. for Par. Diff. Eqs.* **28(1)** (2012), 188–203.

[40] Li, Z. Immersed interface methods for moving interface problems. *Numerical algorithms* **14(4)** (1997), 269-293.

[41] Huang, H., & Li, Z. (1999). Convergence analysis of the immersed interface method. IMA Journal of Numerical Analysis, 19(4), 583-608.