A high-order embedded-boundary method based on smooth extension and RBFs for solving elliptic equations in multiply connected domains

N. Mai-Duy[1,*] and Y.T. Gu[2]
[1] School of Engineering,
University of Southern Queensland, Toowoomba, QLD 4350, Australia
[2] School of Mechanical, Medical and Process Engineering,
Queensland University of Technology, Brisbane, QLD 4000, Australia

Submitted to *EABE*, March 2023; revised, May 2023

ABSTRACT This paper presents a new high-order embedded-/immersed-boundary method, based on point collocation, smooth extension of the solution and integrated radial basis functions (IRBFs), for solving the elliptic partial differential equation (PDE) defined in a domain with holes. The PDE is solved in the domain without holes, where the construction of the IRBF approximations is based on a fixed Cartesian grid and local five-point stencils, and the inner/immeresed boundary conditions are included in the discretized equations. More importantly, nodal values of the second-, third- and fourth-order derivatives of the field variable are incorporated into the IRBF approximations, and the forcing term defined in the holes is constructed in a form that gives a globally smooth solution. These features enable the proposed scheme to achieve high level of sparseness of the system matrix, and high level of accuracy of the solution together. Numerical verification is carried out for problems with smooth and non-smooth inner boundaries. Highly accurate results are obtained using relatively coarse grids.

Keywords: embedded boundary, smooth extension, integrated radial basis function, elliptic

*Corresponding author E-mail: nam.mai-duy@usq.edu.au, Telephone +61-7-46312748, Fax +61-7-46312529

equation, multiply connected domain, high-order discretization

1 Introduction

In the context of numerical solution of elliptic PDEs, the development of efficient and robust methods for nontrivial domains has still received a great deal of attention. Various mesh-based discretizations were developed and they can be classified into boundary-fitted-mesh and embedded-/immersed-boundary methods.

In the boundary-fitted-mesh category including the finite-element and boundary-element methods, unstructured meshes are typically used. Boundary-fitted meshes can be used to accurately represent a geometrically complex domain. However, generating a high-quality mesh is a time-consuming process.

In the embedded-/immersed-boundary category, the physical domain is first embedded in a simpler domain and the PDE is then solved by using efficient and robust structured-grid discretizations. For the Immersed Interface [1], Ghost Fluid [2] and Volume Penalty [3] methods, the embedded boundary conditions are directly introduced into the discretized equations. For the Immersed Boundary (IB) method, which was proposed by Peskin [4] for fluid flows with immersed elastic bodies, the effect of the IB on the fluid flow is imposed by the introduction of a forcing term into the continuous governing equations, which is a singular function with support only along the IB. Its advantages include the ability to effectively handle moving boundaries and straightforward extension to higher dimensions. However, the method is only first-order accurate. To carry out the discretization on a rectangular domain, effective strategies based on the smooth extension of the forcing function or the solution were proposed (e.g. [5,6,7]).

Radial basis functions (RBFs) have become an important tool in numerical analysis, particularly in areas of function approximation and numerical solution of PDEs (e.g. [8,9,10,11,12,13,14]). Examples of RBFs that are of particular interest in practice are the multiquadric, inverse

multiquadric and Gaussian functions. For interpolating data, these types of RBFs are capable of having a spectral convergence rate. In the conventional approach, a function is first represented by a linear combination of RBFs, and its derivatives are then obtained by differentiating the RBFs. In the integral (IRBF) approach, the RBFs are employed to represent a derivative and then integrated to obtain approximations for lower-order derivatives and a function itself. The integral approach enables the RBF approximations to avoid a reduction in the convergence rate caused by differentiation. In addition, the presence of the integration constants allows for the exact imposition of extra information such as nodal derivative values in the system of converting the RBF space into the physical space. Recently, we proposed new approximations for one-dimensional 3-point and two-dimensional 5-point IRBF stencils for solving second-order differential problems [15,16]. The second-order derivatives of the field variable at the central node of a stencil are expressed in terms of the nodal values of not only the variable and but also its derivatives of second order and higher ones at the end nodes. The inclusion of nodal derivative values of orders up to 4 or higher can bring several benefits: a significant improvement in accuracy and independency of the solution accuracy from the RBF width. The latter helps overcome the challenging issue of finding the optimal value of the RBF width. Since the approximations are based on RBFs, these stencils can work with non-constant intervals, which allows for the discretization being carried out on non-rectangular domains, and enables the grid density to be higher in areas where the slopes of the solution are steep The high-order IRBF stencils were then successfully applied for the solution of the first biharmonic equation using the non-coupled approach [17] and the coupled-equation approach [18]. Our previous IRBF works about the use of high-order IRBF stencils for arbitrarily shaped domains [15,16,17,18] were all based on boundary-fitted grids. In this study, the elliptic PDE is solved on a Cartesian grid that covers the physical multiply-connected domain and the region of the holes. It can be seen that the grid generally does not conform to the geometry of the holes. In order to achieve high-order accuracy, apart from applying the high-order IRBF stencils [15], we also adopt the smooth extension approach [6] to convert problems defined in multiply connected domains into those in rectangles, where the solution in the computational domain is forced to be globally smooth. The focus of this study lies in the handling of inner boundaries in the context of embedded-boundary

methods..

The remainder of the paper is organized as follows. In Section 2, the proposed method is presented, including the mathematical formulation for the smooth extension approach, the IRBF discretization on a Cartesian grid, the modification of the discretized equations to account for the IB presence, and the construction of a smooth extension of the solution into the computational domain. In Section 3, numerical verification is carried out, where test problems with analytic solutions and no exact solutions are considered. Section 4 gives some concluding remarks.

2 Proposed method

2.1 Problem formulation

Consider the second-order elliptic problem with Dirichlet boundary conditions

$$\mathcal{L}\bar{u} = f \quad \text{in } \Omega, \tag{1}$$

$$\bar{u} = g \quad \text{on } \Gamma, \tag{2}$$

where $\mathcal{L}$ is a second-order elliptic operator, f and g are some smooth functions, Ω is the physical multiply-connected domain and Γ is its boundary. Let H be the region of the holes, called the extension domain, and C be the domain that is comprised of Ω and H, called the computational domain.

Let u^p and u^h be the approximate solution in Ω and H, respectively. The approximate solution u in C is then defined to be

$$u = \chi_\Omega u^p + \chi_H u^h, \tag{3}$$

where χ_X denotes the characteristic function of the domain X. The solution u reduces to u^p

when restricted to the physical domain Ω and u^h when restricted to the extension domain H.

To achieve high-order accuracy, the solution u^h in a hole should be constructed to be smooth across the hole's boundary. Assume that u^p is a known function that is C^k continuous in Ω. In the smooth extension approach [6], the solution u^h is found by solving the following Kth-order PDE in a hole

$$\begin{cases} \mathcal{H}^K u^h = 0, \\ \frac{\partial^j u^h}{\partial n^j} = \frac{\partial^j u^p}{\partial n^j}, \end{cases} \tag{4}$$

where $j = (0, 1, \ldots, k)$, $K = 2(k+1)$ and $\partial^j u / \partial n^j$ is the jth normal derivative of u on the hole's boundary. As discussed in [6], the system matrix becomes ill-conditioned for large k. In this study, the differential problem is solved iteratively. The solution at the previous iteration is known and we utilize it to derive the boundary conditions for the smooth extension problem (4). We also investigate the case of using $K < 2(k+1)$ and it will be shown that high-order accuracy is still achieved for $K = 4$ when $k \geq 2$. The lower the value of K the better the matrix condition number will be. After solving (4), we can compute a forcing function F_h associated with u^h:

$$F_h = \mathcal{L} u^h. \tag{5}$$

The extended forcing function $\tilde{f}$ becomes

$$\tilde{f} = \chi_\Omega f + \chi_H F_h. \tag{6}$$

The problem to be solved is now defined as

$$\mathcal{L} u = \chi_\Omega f + \chi_H F_h \quad \text{in } C, \tag{7}$$

$$u = g \quad \text{on } \Gamma. \tag{8}$$

When $x \in \Omega$, the new problem (7)-(8) reduces to the physical problem of interest.

2.2 IRBF approximations

We describe the proposed method for its use in solving the Poisson equation

$$\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} = f, \tag{9}$$

with the physical domain being a square with several holes (Figure 1). Cartesian grids are used to represent the computational domain C (the square without holes). The approximations used in the proposed method are based on one-dimensional IRBFs. Let η represent the independent variables x and y, and v the dependent variables u and u^h. In the η direction, the variable v along a grid line is represented by using the following IRBF scheme

$$\frac{\partial^q v(\eta)}{\partial \eta^q} = \sum_{k=1}^{N_\eta} w_k G_k(\eta) = \sum_{k=1}^{N_\eta} w_k I_k^{(q)}(\eta), \tag{10}$$

$$\frac{\partial^{q-1} v(\eta)}{\partial \eta^{q-1}} = \sum_{k=1}^{N_\eta} w_k I_k^{(q-1)}(\eta) + c_1, \tag{11}$$

$$\frac{\partial^{q-2} v(\eta)}{\partial \eta^{q-2}} = \sum_{k=1}^{N_\eta} w_k I_k^{(q-2)}(\eta) + c_1\eta + c_2, \tag{12}$$

$$\cdots \quad \cdots \quad \cdots$$

$$\frac{\partial v(\eta)}{\partial \eta} = \sum_{k=1}^{N_\eta} w_k I_k^{(1)}(\eta) + c_1 \frac{\eta^{q-2}}{(q-2)!} + c_2 \frac{\eta^{q-3}}{(q-3)!} + \cdots + c_{q-1}, \tag{13}$$

$$v(\eta) = \sum_{k=1}^{N_\eta} w_k I_k^{(0)}(\eta) + c_1 \frac{\eta^{q-1}}{(q-1)!} + c_2 \frac{\eta^{q-2}}{(q-2)!} + \cdots + c_{q-1}\eta + c_q, \tag{14}$$

where N_η is the number of RBF centers (grid points) under consideration, $G_k(\eta)$ is the RBF, $I_k^{(q-1)}(\eta) = \int I_k^{(q)}(\eta) d\eta$, $I_k^{(q-2)}(\eta) = \int I_k^{(q-1)}(\eta) d\eta, \ldots, I_k^{(0)}(\eta) = \int I_k^{(1)}(\eta) d\eta$, $(w_1, w_2, \ldots, w_{N_\eta})$ the RBF coefficients, and $(c_1, c_2, \ldots, c_q)$ the integration constants. For the multiquadric function, $G_k(\eta) = \sqrt{(\eta - \eta_k)^2 + a_k^2}$, where η_k is the center and a_k is the width/shape-parameter. In (10)-(14), RBFs are integrated q times and we refer to it as an IRBF scheme of order q, denoted by IRBFq.

Figure 2 shows a flowchart describing the numerical procedure of the proposed method.

The constants arising from the process of integrating the RBFs provide an effective means of including nodal derivative values of the field variable into the approximations. This inclusion can bring many benefits. For local stencils used in discretizing the PDE, it results in a significant improvement in the solution accuracy and enables the IRBF solution to be not influenced much by the RBF width. For the smooth extension problem, it enables the high-order PDE to be enforced at every interior grid node.

It should be pointed out that the goal of the extension problem is to achieve some level of smoothness of the solution across the IB rather than the solution itself, and thus one can have some flexibility. For example, for easy implementation, the boundary nodes used in solving the extension problem (4) can be chosen to be grid nodes near the IB rather than points on the actual boundary, and the cross derivative terms in the high-order PDE can be set aside. Since the solution in a hole is not part of the solution to the physical problem, such implementations are expected not to affect much the accuracy of the solution of the physical problem.

2.2.1 Discretizing the PDE without regard to the immersed boundary

Consider an interior node and its associated four neighbouring nodes (Figure 3). We apply the 1D IRBF scheme (10)-(14) to compute $\partial^2 u/\partial x^2$ and $\partial^2 u/\partial y^2$ at the central node (x_i, y_j). In the η direction, the IRBF approximation involves a set of 3 grid points $(\eta_{i-1}, \eta_i, \eta_{i+1})$ $(N_\eta = 3)$. For IRBFq, there are q integration constants and we utilise them to add q extra equations to the conversion of the RBF space into the physical space. It was shown in [17] that the IRBF solution is not influenced much by the RBF width when $q \geq 6$. In this work, $q = 6$ is implemented. In the following conversion system, the equations are employed to impose $(\partial^2 u/\partial \eta^2, \partial^3 u/\partial \eta^3, \partial^4 u/\partial \eta^4)$ at the two end-nodes.

$$\widehat{u} = \mathcal{C}\widehat{w}, \tag{15}$$

where

$$\widehat{u} = \left(u_{i-1}, u_i, u_{i+1}, \frac{\partial^2 u_{i-1}}{\partial \eta^2}, \frac{\partial^2 u_{i+1}}{\partial \eta^2}, \frac{\partial^3 u_{i-1}}{\partial \eta^3}, \frac{\partial^3 u_{i+1}}{\partial \eta^3}, \frac{\partial^4 u_{i-1}}{\partial \eta^4}, \frac{\partial^4 u_{i+1}}{\partial \eta^4} \right)^T,$$

$$\widehat{w} = \left(w_1, w_2, w_3, c_1, c_2, c_3, c_4, c_5, c_6 \right)^T,$$

and $\mathcal{C}$ is the matrix of size 9-by-9 (called the conversion matrix)

$$\mathcal{C} = \begin{bmatrix} I_1^{(0)}(\eta_{i-1}) & I_2^{(0)}(\eta_{i-1}) & I_3^{(0)}(\eta_{i-1}) & \frac{\eta_{i-1}^5}{5!} & \frac{\eta_{i-1}^4}{4!} & \frac{\eta_{i-1}^3}{3!} & \frac{\eta_{i-1}^2}{2} & \eta_{i-1} & 1 \\ I_1^{(0)}(\eta_i) & I_2^{(0)}(\eta_i) & I_3^{(0)}(\eta_i) & \frac{\eta_i^5}{5!} & \frac{\eta_i^4}{4!} & \frac{\eta_i^3}{3!} & \frac{\eta_i^2}{2} & \eta_i & 1 \\ I_1^{(0)}(\eta_{i+1}) & I_2^{(0)}(\eta_{i+1}) & I_3^{(0)}(\eta_{i+1}) & \frac{\eta_{i+1}^5}{5!} & \frac{\eta_{i+1}^4}{4!} & \frac{\eta_{i+1}^3}{3!} & \frac{\eta_{i+1}^2}{2} & \eta_{i+1} & 1 \\ I_1^{(2)}(\eta_{i-1}) & I_2^{(2)}(\eta_{i-1}) & I_3^{(2)}(\eta_{i-1}) & \frac{\eta_{i-1}^3}{3!} & \frac{\eta_{i-1}^2}{2} & \eta_{i-1} & 1 & 0 & 0 \\ I_1^{(2)}(\eta_{i+1}) & I_2^{(2)}(\eta_{i+1}) & I_3^{(2)}(\eta_{i+1}) & \frac{\eta_{i+1}^3}{3!} & \frac{\eta_{i+1}^2}{2} & \eta_{i+1} & 1 & 0 & 0 \\ I_1^{(3)}(\eta_{i-1}) & I_2^{(3)}(\eta_{i-1}) & I_3^{(3)}(\eta_{i-1}) & \frac{\eta_{i-1}^2}{2} & \eta_{i-1} & 1 & 0 & 0 & 0 \\ I_1^{(3)}(\eta_{i+1}) & I_2^{(3)}(\eta_{i+1}) & I_3^{(3)}(\eta_{i+1}) & \frac{\eta_{i+1}^2}{2} & \eta_{i-1} & 1 & 0 & 0 & 0 \\ I_1^{(4)}(\eta_{i-1}) & I_2^{(4)}(\eta_{i-1}) & I_3^{(4)}(\eta_{i-1}) & \eta_{i-1} & 1 & 0 & 0 & 0 & 0 \\ I_1^{(4)}(\eta_{i+1}) & I_2^{(4)}(\eta_{i+1}) & I_3^{(4)}(\eta_{i+1}) & \eta_{i+1} & 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Solving (15) yields

$$\widehat{w} = \mathcal{C}^{-1} \widehat{u}. \tag{16}$$

The second derivative of u at η_i is thus calculated by

$$\frac{\partial^2 u_i}{\partial \eta^2} = \mathcal{D}_{2\eta} \widehat{u}, \tag{17}$$

where $\mathcal{D}_{2\eta}$ is a row matrix of 9 coefficients that is defined as

$$\mathcal{D}_{2\eta} = \begin{bmatrix} I_1^{(2)}(\eta_i) & I_2^{(2)}(\eta_i) & I_3^{(2)}(\eta_i) & \frac{\eta_i^3}{3!} & \frac{\eta_i^2}{2} & \eta_i & 1 & 0 & 0 \end{bmatrix} \mathcal{C}^{-1}.$$

In practice, the coefficient set $\mathcal{D}_{2\eta}$ is acquired by using Gaussian elimination to solve the following algebraic equation set:

$$\mathcal{C}^T \mathcal{D}_{2\eta}^T = \left(I_1^{(2)}(\eta_i), I_2^{(2)}(\eta_i), I_3^{(2)}(\eta_i), \frac{\eta_i^3}{3!}, \frac{\eta_i^2}{2}, \eta_i, 1, 0, 0 \right)^T. \tag{18}$$

2.2.2 Modifying the discretized system to account for the presence of the immersed boundary

The stencils near the immersed boundary are modified to account for its presence. In each direction, we construct the approximation based on 4 nodes (three grid nodes: $(\eta_{i-1}, \eta_i, \eta_{i+1})$ and one boundary node: η_b) by also using IRBF6 (i.e. $q = 6$). Two configurations of nodes are shown in Figure 4; they share a common feature: there are two grid nodes in Ω and one grid node in H. The conversion system takes the form

$$\widehat{U} = \mathcal{C}_b \widehat{W}. \tag{19}$$

where

$$\widehat{U} = \left(u_{i-1}, u_i, u_{i+1}, u_b, \frac{\partial^2 u_{i-1}}{\partial \eta^2}, \frac{\partial^2 u_{i+1}}{\partial \eta^2}, \frac{\partial^3 u_{i-1}}{\partial \eta^3}, \frac{\partial^3 u_{i+1}}{\partial \eta^3}, \frac{\partial^4 u_{i-1}}{\partial \eta^4}, \frac{\partial^4 u_{i+1}}{\partial \eta^4} \right)^T,$$

$$\widehat{W} = \left(w_1, w_2, w_3, w_4, c_1, c_2, c_3, c_4, c_5, c_6 \right)^T,$$

and $\mathcal{C}_b$ is the matrix of size 10-by-10

$$\mathcal{C} = \begin{bmatrix}
I_1^{(0)}(\eta_{i-1}) & I_2^{(0)}(\eta_{i-1}) & I_3^{(0)}(\eta_{i-1}) & I_4^{(0)}(\eta_{i-1}) & \frac{\eta_{i-1}^5}{5!} & \frac{\eta_{i-1}^4}{4!} & \frac{\eta_{i-1}^3}{3!} & \frac{\eta_{i-1}^2}{2} & \eta_{i-1} & 1 \\
I_1^{(0)}(\eta_i) & I_2^{(0)}(\eta_i) & I_3^{(0)}(\eta_i) & I_4^{(0)}(\eta_i) & \frac{\eta_i^5}{5!} & \frac{\eta_i^4}{4!} & \frac{\eta_i^3}{3!} & \frac{\eta_i^2}{2} & \eta_i & 1 \\
I_1^{(0)}(\eta_{i+1}) & I_2^{(0)}(\eta_{i+1}) & I_3^{(0)}(\eta_{i+1}) & I_4^{(0)}(\eta_{i+1}) & \frac{\eta_{i+1}^5}{5!} & \frac{\eta_{i+1}^4}{4!} & \frac{\eta_{i+1}^3}{3!} & \frac{\eta_{i+1}^2}{2} & \eta_{i+1} & 1 \\
I_1^{(0)}(\eta_b) & I_2^{(0)}(\eta_b) & I_3^{(0)}(\eta_b) & I_4^{(0)}(\eta_b) & \frac{\eta_b^5}{5!} & \frac{\eta_b^4}{4!} & \frac{\eta_b^3}{3!} & \frac{\eta_b^2}{2} & \eta_b & 1 \\
I_1^{(2)}(\eta_{i-1}) & I_2^{(2)}(\eta_{i-1}) & I_3^{(2)}(\eta_{i-1}) & I_4^{(2)}(\eta_{i-1}) & \frac{\eta_{i-1}^3}{3!} & \frac{\eta_{i-1}^2}{2} & \eta_{i-1} & 1 & 0 & 0 \\
I_1^{(2)}(\eta_{i+1}) & I_2^{(2)}(\eta_{i+1}) & I_3^{(2)}(\eta_{i+1}) & I_4^{(2)}(\eta_{i+1}) & \frac{\eta_{i+1}^3}{3!} & \frac{\eta_{i+1}^2}{2} & \eta_{i+1} & 1 & 0 & 0 \\
I_1^{(3)}(\eta_{i-1}) & I_2^{(3)}(\eta_{i-1}) & I_3^{(3)}(\eta_{i-1}) & I_4^{(3)}(\eta_{i-1}) & \frac{\eta_{i-1}^2}{2} & \eta_{i-1} & 1 & 0 & 0 & 0 \\
I_1^{(3)}(\eta_{i+1}) & I_2^{(3)}(\eta_{i+1}) & I_3^{(3)}(\eta_{i+1}) & I_4^{(3)}(\eta_{i+1}) & \frac{\eta_{i+1}^2}{2} & \eta_{i-1} & 1 & 0 & 0 & 0 \\
I_1^{(4)}(\eta_{i-1}) & I_2^{(4)}(\eta_{i-1}) & I_3^{(4)}(\eta_{i-1}) & I_4^{(4)}(\eta_{i-1}) & \eta_{i-1} & 1 & 0 & 0 & 0 & 0 \\
I_1^{(4)}(\eta_{i+1}) & I_2^{(4)}(\eta_{i+1}) & I_3^{(4)}(\eta_{i+1}) & I_4^{(4)}(\eta_{i+1}) & \eta_{i+1} & 1 & 0 & 0 & 0 & 0
\end{bmatrix}.$$

The only difference between $\widehat{U}$ in (19) and $\widehat{u}$ in (15) is that the former contains an extra value, u_b, which is the boundary condition on the IB. The second-order derivative of u at the

central node of the stencil, i.e. $\partial^2 u_i/\partial\eta^2$, is now expressed in terms of u at $(\eta_{i-1}, \eta_i, \eta_{i+1}, \eta_b)$ and its nodal derivatives at only (η_{i-1}, η_{i+1}). When compared to (17), there is an additional term associated with u_b, which represents the effect of the IB on the discretization in the entire computational domain.

2.2.3 Constructing a smooth extension of the solution u into the computational domain C

Two configurations of nodes representing the hole are considered: one involving the physical boundary points and grid nodes, and the other involving only grid nodes (Figure 5). The boundary nodes used in solving (4) are chosen to be the physical boundary points for the former, and outside grid points adjacent to the boundary for the latter. It is easier to implement the discretization for the latter. It is noted that (i) the extension problem is solved in the same way for the two node configurations, and (ii) the boundary conditions for the physical problem are already imposed at the physical boundary points in the process of discretizing the PDE in the computational domain. For the global C^k regularity, the boundary values of u^h and its derivatives $(\partial u^h/\partial\eta, \ldots, \partial^k u^h/\partial\eta^k)$ are imposed. It is noted that $(\partial u^h/\partial x, \ldots, \partial^k u^h/\partial x^k)$ are for the horizontal grid lines, while $(\partial u^h/\partial y, \ldots, \partial^k u^h/\partial y^k)$ are for the vertical grid lines. These boundary values are obtained after the IRBF approximations of u on the entire grid lines are carried out.

For each grid line in the extension domain, we employ IRBFq with $q = 2k$ to accommodate $2k$ boundary derivative values at the two end-nodes of the grid line. The conversion system takes the form

$$\widehat{u}_h = \mathcal{C}_h \widehat{w}_h. \tag{20}$$

For $k = 1$,

$$\widehat{u}_h = \left(u_1^h, u_2^h, \ldots, u_{N_\eta}^h, \frac{\partial u_1^h}{\partial\eta}, \frac{\partial u_{N_\eta}^h}{\partial\eta} \right)^T,$$

$$\widehat{w}_h = \left(w_1, w_2, \ldots, w_{N_\eta}, c_1, c_2 \right)^T,$$

and $\mathcal{C}_h$ is the matrix of size $(N_\eta+2)$-by-$(N_\eta+2)$

$$\mathcal{C}_h = \begin{bmatrix} I_1^{(0)}(\eta_1) & I_2^{(0)}(\eta_1) & \dots & I_{N_\eta}^{(0)}(\eta_1) & \eta_1 & 1 \\ I_1^{(0)}(\eta_2) & I_2^{(0)}(\eta_2) & \dots & I_{N_\eta}^{(0)}(\eta_2) & \eta_2 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ I_1^{(0)}(\eta_{N_\eta}) & I_2^{(0)}(\eta_{N_\eta}) & \dots & I_{N_\eta}^{(0)}(\eta_{N_\eta}) & \eta_{N_\eta} & 1 \\ I_1^{(1)}(\eta_1) & I_2^{(1)}(\eta_1) & \dots & I_{N_\eta}^{(1)}(\eta_1) & 1 & 0 \\ I_1^{(1)}(\eta_{N_\eta}) & I_2^{(1)}(\eta_{N_\eta}) & \dots & I_{N_\eta}^{(1)}(\eta_{N_\eta}) & 1 & 0 \end{bmatrix}.$$

For $k \geq 2$

$$\widehat{u}_h = \left(u_1^h, u_2^h, \dots, u_{N_\eta}^h, \frac{\partial u_1^h}{\partial \eta}, \frac{\partial u_{N_\eta}^h}{\partial \eta}, \dots, \frac{\partial^k u_1^h}{\partial \eta^k}, \frac{\partial^k u_{N_\eta}^h}{\partial \eta^k} \right)^T,$$

$$\widehat{w}_h = \left(w_1, w_2, \dots, w_{N_\eta}, c_1, c_2, \dots, c_{2k-1}, c_{2k} \right)^T,$$

and $\mathcal{C}_h$ is the matrix of size $(N_\eta+2k)$-by-$(N_\eta+2k)$

$$\mathcal{C}_h = \begin{bmatrix} I_1^{(0)}(\eta_1) & I_2^{(0)}(\eta_1) & \dots & I_{N_\eta}^{(0)}(\eta_1) & \frac{\eta_1^{2k-1}}{(2k-1)!} & \dots & \eta_1 & 1 \\ I_1^{(0)}(\eta_2) & I_2^{(0)}(\eta_2) & \dots & I_{N_\eta}^{(0)}(\eta_2) & \frac{\eta_2^{2k-1}}{(2k-1)!} & \dots & \eta_2 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ I_1^{(0)}(\eta_{N_\eta}) & I_2^{(0)}(\eta_{N_\eta}) & \dots & I_{N_\eta}^{(0)}(\eta_{N_\eta}) & \frac{\eta_{N_\eta}^{2k-1}}{(2k-1)!} & \dots & \eta_{N_\eta} & 1 \\ I_1^{(1)}(\eta_1) & I_2^{(1)}(\eta_1) & \dots & I_{N_\eta}^{(1)}(\eta_1) & \frac{\eta_1^{2k-2}}{(2k-2)!} & \dots & 1 & 0 \\ I_1^{(1)}(\eta_{N_\eta}) & I_2^{(1)}(\eta_{N_\eta}) & \dots & I_{N_\eta}^{(1)}(\eta_{N_\eta}) & \frac{\eta_{N_\eta}^{2k-2}}{(2k-2)!} & \dots & 1 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ I_1^{(k)}(\eta_1) & I_2^{(k)}(\eta_1) & \dots & I_{N_\eta}^{(k)}(\eta_1) & \frac{\eta_1^{k-1}}{(k-1)!} & \dots & 0 & 0 \\ I_1^{(k)}(\eta_{N_\eta}) & I_2^{(k)}(\eta_{N_\eta}) & \dots & I_{N_\eta}^{(k)}(\eta_{N_\eta}) & \frac{\eta_{N_\eta}^{k-1}}{(k-1)!} & \dots & 0 & 0 \end{bmatrix}.$$

Solving (15) yields

$$\widehat{w}_h = \mathcal{C}_h^{-1} \widehat{u}_h. \tag{21}$$

It is noted that value of N_η employed here is typically much smaller than the number of grid nodes of the entire grid line. Making use of (21), the derivatives of any order of the variable u^h with respect to x and y can be computed in terms of the nodal values of u^h and the boundary values of its derivatives. In solving (4), the Kth-order derivatives are required.

In this study, we consider K in range of 4 to $2(k+1)$. The advantage of incorporating the boundary conditions into the IRBF approximations is that it allows the equation in (4) to be enforced at every interior node. For 1D problems, the extension problem (4) can be directly solved in the RBF space. Since the high-order ordinary differential equation can be enforced at the two boundary nodes, there are two additional equations needed and we employ IRBFq with $q = 2(k+1)$ for the global C^k regularity.

2.2.4 Solving the resultant algebraic system

We apply the Newton method with finite-difference Jacobian to solve the resultant algebraic system. At each iteration, we carry out the following calculation tasks: (i) computes nodal values of derivatives of second order and higher ones along the grid lines using the 1D IRBF scheme [16], from which the boundary conditions (i.e. continuity conditions) for the Kth-order PDE in the extension domain are acquired; (ii) solves the extension problem for u^h, from which the forcing function in the extension domain (i.e. F_h) is derived; (iii) modifies the forcing function by including new F_h; and (iv) updates the solution u in the computational domain. The approximation of the derivatives along a grid line involves only its grid nodes and thus the differentiation matrices stay the same whatever the number of holes inside the domain.

3 Examples

We test the proposed method in several 1D and 2D problems. For the latter, both smooth and non-smooth inner boundaries are considered. The RBF width is simply chosen as 0.0001 for all IRBF calculations. Let h be the grid size. We employ M uniform grids, $(h_1, h_2, \cdots, h_M)$, to represent the computational domain. For each grid, the solution accuracy is measured in the form of relative L_2-norm

$$N_e = \frac{\sqrt{\sum_{i=1}^{N} \left(u_i^p - \bar{u}_i\right)^2}}{\sqrt{\sum_{i=1}^{N} (\bar{u}_i)^2}}, \tag{22}$$

where N is the number of grid points in the physical domain, and $\bar{u}$ is the exact solution. We fit power function $N_e = bh^m$, where m and b are the two coefficients, to the obtained data (i.e. (h_i, N_{e_i}), $i = (1, 2, \cdots, M)$). The value of m is considered as the (average) rate of convergence. The computations are carried out using MATLAB®R2022a on a laptop with Processor: Intel(R) Core(TM) i7-10610U CPU @ 1.80 GHz 2.30 GHz, and 16GB RAM.

3.1 Example 1

Consider the following one-dimensional problem

$$\frac{d^2\bar{u}}{dx^2} = -4\pi^2 \sin(2\pi x) \quad \text{in } \Omega = [x_A, x_B] \setminus [x_C, x_D], \tag{23}$$

$$\bar{u} = 0 \quad \text{on } \Gamma = (x_C, x_D), \tag{24}$$

$$\frac{d^r\bar{u}_A}{dx^r} = \frac{d^r\bar{u}_B}{dx^r}, \tag{25}$$

where $r = (0, 1, 2, \dots)$, $x_A = 0$, $x_B = 1$, $x_C = 0.3$ and $x_D = 0.7$ (Figure 6).

First, we analytically solve the equation in the physical domain Ω and in the extension domain H, separately. The latter takes the same forcing function as the former. Figure 7 displays the two obtained analytical solutions. It can be seen that the direct use of a given forcing function in the non-physical portion $[x_C, x_D]$ produces a solution in the computational domain that is not smooth across the interfaces C and D, which leads to low-order convergence in numerical schemes.

To achieve high-order accuracy, we construct a smooth extension of the solution into the computational domain. Different levels of the global regularity (i.e. different values of k in C^k) are considered. Figure 8 shows a forcing function in $[x_C, x_D]$ for $k = 2$. The solution u is seen to be globally smooth in the computational domain, while the corresponding forcing function is no longer smooth across the interfaces.

Figure 9 shows IRBF solutions in the computational domain obtained with $k = (0, 1, 2, 3, 4)$.

It can be seen that the solutions produced in the non-physical portion are quite different in terms of the peak value.

To study the convergence of the proposed method with respect to grid size, several uniform data sets, namely $N_x = (30, 40, \ldots, 160)$, are employed. Figures 10 and 11 show the behaviour of the solution error for two different ways of choosing the inner boundary nodes in solving the extension problem (4). In the first approach, the continuity conditions are directly imposed at x_C and x_D. The solution u_p converges as $O(h^{1.0})$, $O(h^{2.0})$, $O(h^{2.9})$, $O(h^{3.6})$, $O(h^{4.4})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively (Figure 10). In the second approach, the continuity conditions are imposed at the grid nodes adjacent to x_C and x_D. The solution u_p converges as $O(h^{1.0})$, $O(h^{2.0})$, $O(h^{2.9})$, $O(h^{3.6})$, $O(h^{4.2})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively (Figure 11). The two approaches yield similar rates of convergence. It is easier to carry out the discretization for the second approach. From now on, only the second approach is considered.

Figure 12 shows the first-, second-, third- and fourth-order derivatives of the solution u in the computational domain for $k = 4$. They are all continuous across the interfaces C and D as expected.

Figure 13 shows the second derivatives of the solution u in the computational domain when different values of k are imposed. The solution u is discontinuous for $k = 1$ and continuous for $k \geq 2$ as expected. The smoothness of the solution is improved with an increase in k.

3.2 Example 2

The PDE is chosen as

$$\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} = -4 \tag{26}$$

defined in the square $0 \leq x, y \leq 2\pi$ with one circular hole located at (π, π) and of radius 0.4. The exact solution to this problem is a second-order polynomial

$$\bar{u} = 4 - (x - \pi)^2 - (y - \pi)^2, \tag{27}$$

which is a C^∞ function with $\partial^r \bar{u}/\partial x^r = \partial^r \bar{u}/\partial y^r = 0$ for $r \geq 3$. The boundary values of $\bar{u}$ are derived from (27).

The problem is solved with different uniform grids, $N_x \times N_y = (11 \times 11, 21 \times 21, \ldots, 91 \times 91)$, and different values of k (different levels of the global regularity). The obtained results are shown in Figure 14. The solution in the physical domain converges as $O(h^{2.1})$, $O(h^{6.4})$, $O(h^{6.4})$, $O(h^{6.4})$ and $O(h^{6.4})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively. Since the exact solution is a polynomial of order 2, the method is able to achieve a high rate of convergence with $k = 1$, and higher values of k only give similar results.

In solving for the smooth extension of the solution, there are different ways of solving the extension problem [6]. $K = 2(k + 1)$ is sufficient for standard discretization schemes to impose the C^k continuity condition. For the proposed method, the presence of the integration constants allows for the addition of nodal derivative values into the conversion system. Making use of it, the C^k-continuity condition ($k \geq 2$) can be imposed with $K = 4$. Here, we also investigate the case of using $K < 2(k + 1)$, i.e. $K = (4, \ldots, 2k)$. The obtained results are shown in Figure 15. For this problem, it can be seen that the order K does not affect the solution accuracy. The sixth-order accuracy is achieved for $K = 4$.

3.3 Example 3

The PDE and the exact solution are respectively given by

$$\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} = \exp(\sin(x))(\cos(x)^2 - \sin(x)) - \cos(y), \tag{28}$$

$$\bar{u} = \exp(\sin(x)) + \cos(y). \tag{29}$$

Unlike Example 2, the exact solution here is a function involving exponential and trigonometric functions. The domain of interest is the square $0 \leq x, y \leq 2\pi$ with 9 holes located at

$$(\pi/2 - \pi/10, \pi/2 - \pi/10), (\pi/2, \pi - \pi/20), (\pi/2 - \pi/10, 3\pi/2 + \pi/20),$$

$$(\pi - \pi/20, \pi/2 + \pi/20), (\pi + 3\pi/40, \pi + \pi/20), (\pi + \pi/20, 3\pi/2 + 3\pi/40),$$

$$(3\pi/2 - \pi/20, \pi/2 - \pi/20), (3\pi/2 + 3\pi/40, \pi - \pi/20), (3\pi/2 + \pi/20, 3\pi/2 - \pi/20),$$

and of radius 0.4 (Figure 16).

This problem is also solved with different uniform grids, $N_x \times N_y = (21 \times 21, 31 \times 31, \ldots, 91 \times 91)$, and different values of k (different levels of the global regularity). The obtained results are shown in Figure 17. The solution in the physical domain converges as $O(h^{2.4})$, $O(h^{3.9})$, $O(h^{6.0})$, $O(h^{8.1})$ and $O(h^{7.9})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively. For a given grid size, an increase in k results in a better accuracy of the solution.

We also study the effect of the order K of the PDE in the extension domain. The obtained results are shown in Figures 18 and 19. With a decrease in K, one has a less accurate solution (Figure 18) but better condition number of matrix $\mathcal{H}^K$ (Figure 19). Nevertheless, the fifth-order accuracy is still achieved for $K = 4$. It should be pointed out that the condition number of the system matrix associated with the computational domain is low, in range of $O(10^1)$ to $O(10^4)$ (Figure 20). It appears that the error does not reduce smoothly with a decrease in grid size. The reason for it could be that the boundary nodes used to account for the presence of the holes do not coincide with the grid nodes. When compared with 1D problems, the proposed method yields a much faster rate of convergence for the same level of the global regularity.

To study the effect of the number of holes on the solution accuracy, we put 9 holes into 3 sets, namely set 1: holes (1,2,3), set 2: holes $(1, 2 \ldots, 6)$ and set 3: holes $(1, 2 \ldots, 9)$. The obtained results are shown in Figure 21. The solution accuracy is not influenced by the number of holes. For $k = 4$ and $K = 10$, all cases yield $O(h^{7.7})$.

3.4 Example 4

Consider the following PDE

$$\frac{\partial^2 \bar{u}}{\partial x^2} + \frac{\partial^2 \bar{u}}{\partial y^2} = \sin(x)\sin(y), \tag{30}$$

defined in the square $0 \leq x, y \leq 2\pi$ with 9 holes of radius 0.4 (Figure 16), and subjected to the boundary condition $\bar{u} = 0$. For this problem, the analytical solution is not available. Our results are compared with the one produced by the finite element method (FEM). We use the Matlab PDE toolbox (version R2022a); the FEM solution, where the mesh consists of 3356 triangles and 1795 mesh points, is shown in Figure 22. Figure 23 displays IRBF solutions obtained with four grids, $(41 \times 41, 46 \times 46, 51 \times 51, 61 \times 61)$. The smoothness of the contour lines is improved with an increase in the grid density. It can be seen that the two methods yield similar results. Unlike the FEM case, the IRBF solution is defined everywhere in the square as shown in Figures 23 and 24.

3.5 Example 5

This example is concerned with finding the electrostatic potential in an air-filled annular quadrilateral frame. This problem is one of the examples considered in the Matlab PDE Toolbox (version R2022a). The governing equation is

$$\frac{\partial}{\partial x}\left(\varepsilon \frac{\partial \bar{V}}{\partial x}\right) + \frac{\partial}{\partial y}\left(\varepsilon \frac{\partial \bar{V}}{\partial y}\right) = -\rho, \tag{31}$$

where ρ is the space charge density, and ε is the absolute dielectric permittivity of the material. Assuming that there is no charge in the domain, the Poisson equation simplifies to the Laplace equation. The electrostatic potential is specified as 1000V at the inner boundary and 0V at the outer boundary.

Unlike previous examples, the hole here is a square (convex polygon, non-smooth immersed

boundary). It is noted that traditional fixed grid methods can suffer from poor performance when applied to non-smooth boundaries [7]. Since the analytical solution is not available, our results are compared with the one produced by the FEM. Figure 25 displays the FEM solution using a grid of 1292 nodes and 596 quadratic triangles, and the IRBF solution using a uniform grid of 30×30. It can be seen that the IRBF method performs well for the non-smooth boundary case with convex holes. There is no special treatment required. Unlike the FEM solution, the IRBF solution is also defined in the square hole.

3.6 Example 6

The non-smooth immersed boundary case is further considered in this example. Here, the PDE is the Poisson equation with $f = -5\sin(x/2\pi)\cos(y/2\pi)$ defined in the square $0 \leq x, y \leq 1$, and subjected to the boundary condition $\bar{u} = 0$. The hole is of L shape (non-convex polygon) located at $(0.3, 0.3)$ and having outer sides of 0.4 and width 0.1 (Figure 26). There is no exact solution to this problem. Figure 26 compares the IRBF solution using a uniform grid of 45×45 with the FEM solution using a mesh of 2484 nodes and 4736 triangular elements. It can be seen that the two solutions are in good agreement, indicating that the proposed method performs well for the non-smooth boundary case with concave holes.

Discussion: All problems in this study are linear; however, due to the presence of derivative terms in the IRBF formulation, it is necessary to employ an iterative procedure to solve the resultant algebraic systems of the proposed method. Numerical results show that the Newton method needs a few iterations to get a converged solution. Table 1 shows the number of iterations as well as the CPU times of the IRBF method and FEM for Example 1 (1D problem), Example 4 (2D problem, 9 circular holes) and Example 6 (2D problem, L-shaped hole). In case of IRBFs, the function tolerance is set to 10^{-6}, and grids chosen are a uniform set of 40 points for Example 1, 41×41 for Example 4 and 45×45 for Example 6. It can be seen that most of the CPU time was spent on solving the algebraic system in the IRBF method, and generating a FE mesh in FEM. It can also be seen that the IRBF method and

FEM yielded efficient solutions; the CPU times of the IRBF method are larger than those of FEM as the former requires the employment of an iterative procedure to find nodal derivative values. When coming to nonlinear problems such as a suspension of rigid particles in a non-Newtonian fluid and the flow past a set of cylinders over a range of Reynolds numbers, an iterative scheme is needed for any numerical method, and the computation of derivatives in the IRBF 5-point stencils can be simply embedded in those iterative steps without extra loop required. It is expected that the two main advantages of the proposed method, namely high-order accuracy and efficient pre-processing, can become very helpful – the proposed method particularly suits such nonlinear problems.

4 Concluding remarks

We have developed a new high-order discretization scheme, based on point collocation, immersed boundary smooth extension and integrated RBFs, for solving the elliptic equation in a multiply connected domain. The discretization is based on a Cartesian grid, which does not conform to the geometry of the holes; the resultant algebraic systems have only 5 nonzero entries per row. High-order accuracy is achieved by including nodal derivative values in the local IRBF approximations, and forcing the solution to be globally smooth in the entire computational domain. Since the proposed IRBF discretization is based on only one-dimensional approximations, the process of obtaining the discretized equations is relatively simple. In solving for the smooth extension of the solution, the constants arising from the process of integrating the RBFs are utilized to include the boundary values of the derivatives in the IRBF approximations, which enables the high-order PDE in the extension domain to be enforced at every interior node. Numerical experiments shows that for a high level of the global regularity, high-order accuracy (≥ 4) is still achieved by simply considering the fourth-order PDE in the extension domain. It is also observed that the method performs better for 2D problems than for 1D problems. This works further shows a great potential of compact RBF stencils in solving differential problems defined on geometrically complex domains.

References

1. LeVeque RJ, Li Z. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM Journal on Numerical Analysis 1994;31(4):1019-1044.

2. Marques AN, Nave J-C, Rosales RR. A correction function method for Poisson problems with interface jump conditions. Journal of Computational Physics 2011;230(20):7567-7597.

3. Angot P, Bruneau CH, Fabrie P. A penalization method to take into account obstacles in incompressible viscous flows. Numer. Math. 1999;81:497–520.

4. Peskin CS. Flow patterns around heart valves: A numerical method. Journal of Computational Physics 1972;10(2):252-271.

5. Boyd JP. Fourier embedded domain methods: extending a function defined on an irregular region to a rectangle so that the extension is spatially periodic and C^{∞}. Applied Mathematics and Computation 2005;161(2):591-597.

6. Stein DB, Guy RD, Thomases B. Immersed boundary smooth extension: A high-order method for solving PDE on arbitrary smooth domains using Fourier spectral methods. Journal of Computational Physics 2016;304:252-274.

7. Qadeer S, Griffith BE. The smooth forcing extension method: A high-order technique for solving elliptic equations on complex domains. Journal of Computational Physics 2021;439:110390.

8. Fasshauer GE. Meshfree Approximation Methods with MATLAB, Interdisciplinary mathematical sciences, Volume 6. Hackensack: World Scientific; 2007.

9. Mai-Duy N, Tanner RI. Computing non-Newtonian fluid flow with radial basis function networks. International Journal for Numerical Methods in Fluids 2005;48:1309-1336.

10. Ebrahimijahan A, Dehghan M, Abbaszadeh M. Compact local integrated radial basis functions (Integrated RBF) method for solving system of non–linear advection-diffusion-reaction equations to prevent the groundwater contamination. Engineering Analysis with Boundary Elements 2020;121:50-64.

11. Ebrahimijahan A, Dehghan M, Abbaszadeh M. Simulation of Maxwell equation based on an ADI approach and integrated radial basis function-generalized moving least squares (IRBF-GMLS) method with reduced order algorithm based on proper orthogonal decomposition. Engineering Analysis with Boundary Elements 2022;143:397-417.

12. Abbaszadeh M, Ebrahimijahan A, Dehghan M. Application of compact local integrated RBF (CLI-RBF) for solving transient forward and backward heat conduction problems with continuous and discontinuous sources. Engineering Analysis with Boundary Elements 2023;146:733-748.

13. Beggs KW, Divo E, Kassab AJ. A meshless multiscale method for simulating hemodynamics. Engineering Analysis with Boundary Elements 2023;150:167-179.

14. Shi CZ, Zheng H, Wen PH, Hon YC. The local radial basis function collocation method for elastic wave propagation analysis in 2D composite plate. Engineering Analysis with Boundary Elements 2023; 150:571-582.

15. Mai-Duy N, Strunin D. New approximations for one-dimensional 3-point and two-dimensional 5-point compact integrated RBF stencils. Engineering Analysis with Boundary Elements 2021;125:12-22.

16. Mai-Duy N, Strunin D, Karunasena W. Computing high-order derivatives in compact integrated-RBF stencils. Engineering Analysis with Boundary Elements 2022;135:369-381.

17. Mai-Duy N, Strunin D, Karunasena W. A new high-order nine-point stencil, based on integrated-RBF approximations, for the first biharmonic equation. Engineering Analysis with Boundary Elements 2022;143:687-699.

18. Mai-Duy N, Tien CMT, Strunin D, Karunasena W. An effective high-order five-point stencil, based on integrated-RBF approximations, for the first biharmonic equation and its applications in fluid dynamics. International Journal of Numerical Methods for Heat & Fluid Flow, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/HFF-11-2022-0673.

Table 1: Number of iteration (niter) and CPU time.

	IRBF				FEM		
		CPU time (seconds)				CPU time (seconds)	
Example	niter	Generating a grid	Constructing the system	Solving the system	niter	Generating a mesh	Solving the system
1	4	negligible	< 1	< 1			
4	3	negligible	< 1	10.86	0	4.84	< 1
6	2	negligible	< 1	8.69	0	2.99	< 1

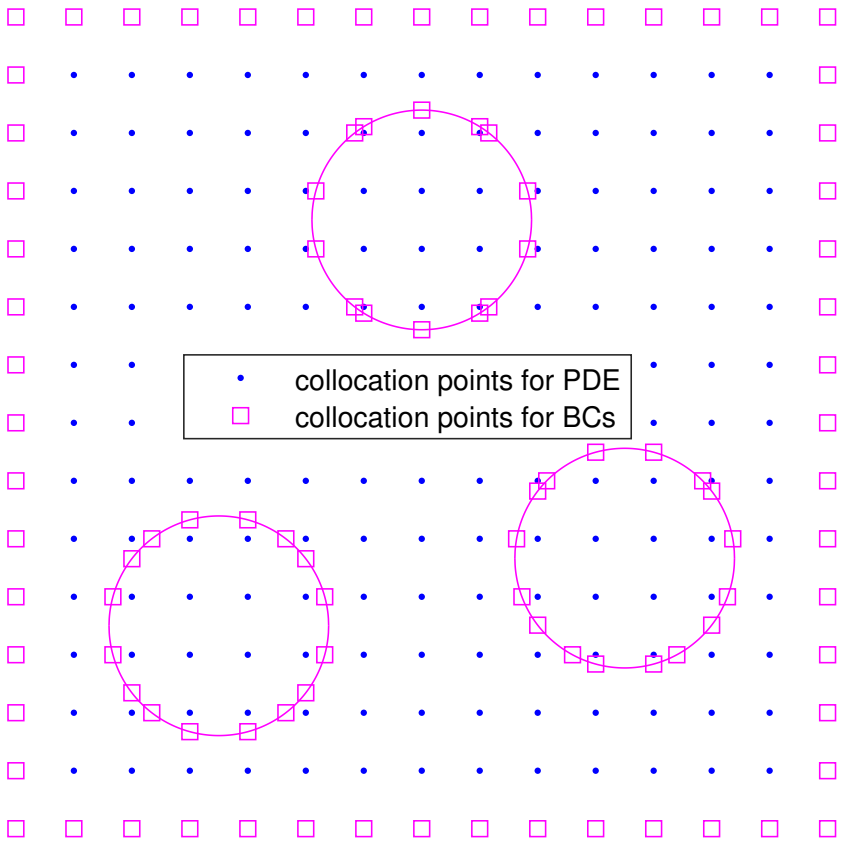


Figure 1: A multiply connected domain: embedded boundaries, Cartesian grid and collocation points. The problem is solved in the domain without holes. Collocation points for enforcing the PDE are all grid nodes inside the outer boundary (blues points), while collocation points for imposing the boundary conditions are points where the grid lines cross the outer and inner boundaries (magenta squares).

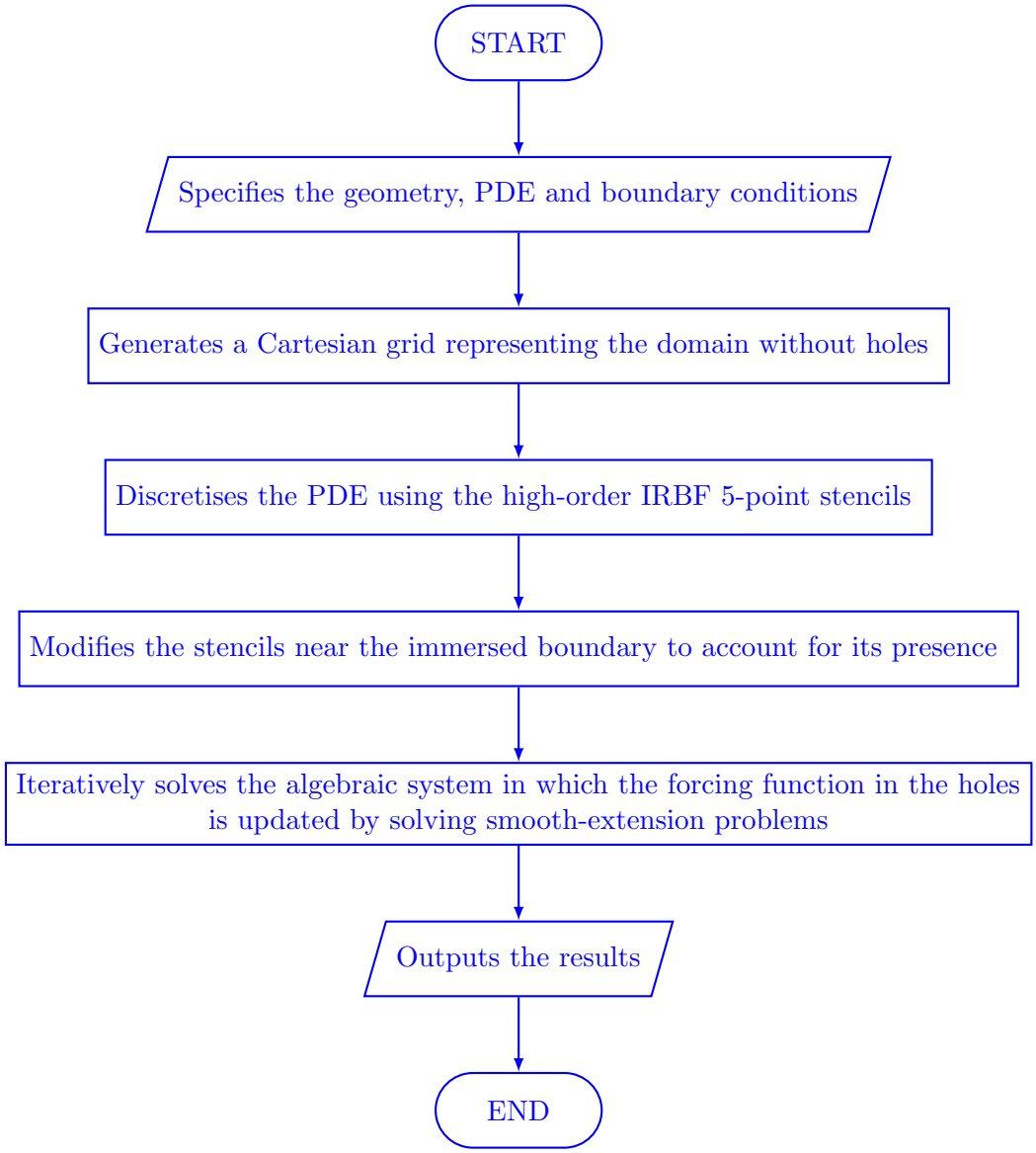


Figure 2: A flowchart of the numerical procedure of the proposed method.

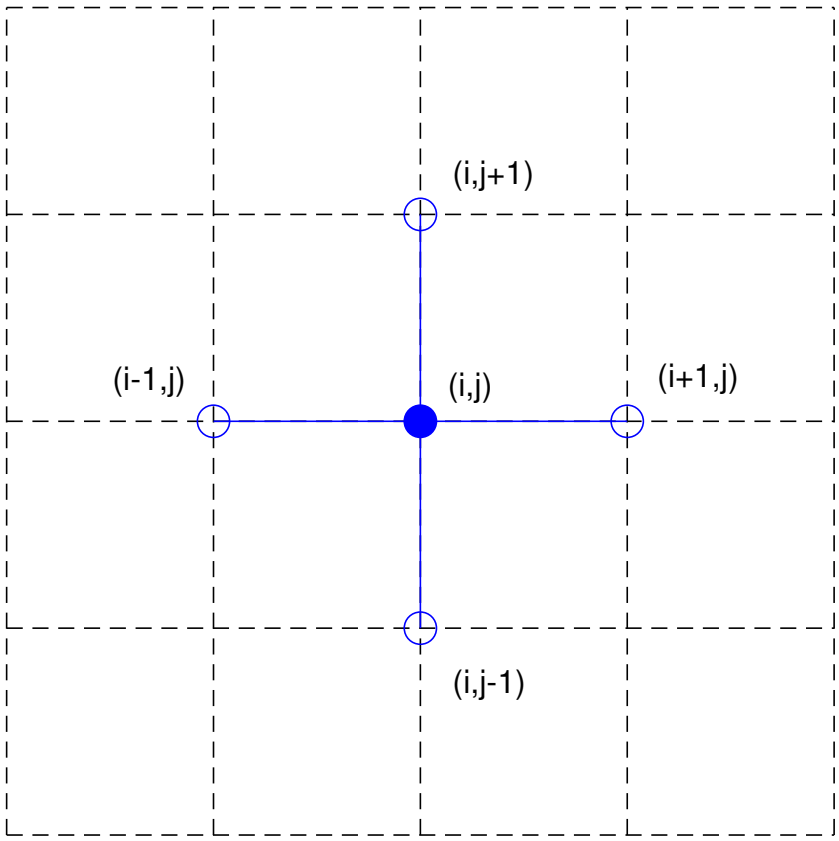


Figure 3: A five-point stencil.

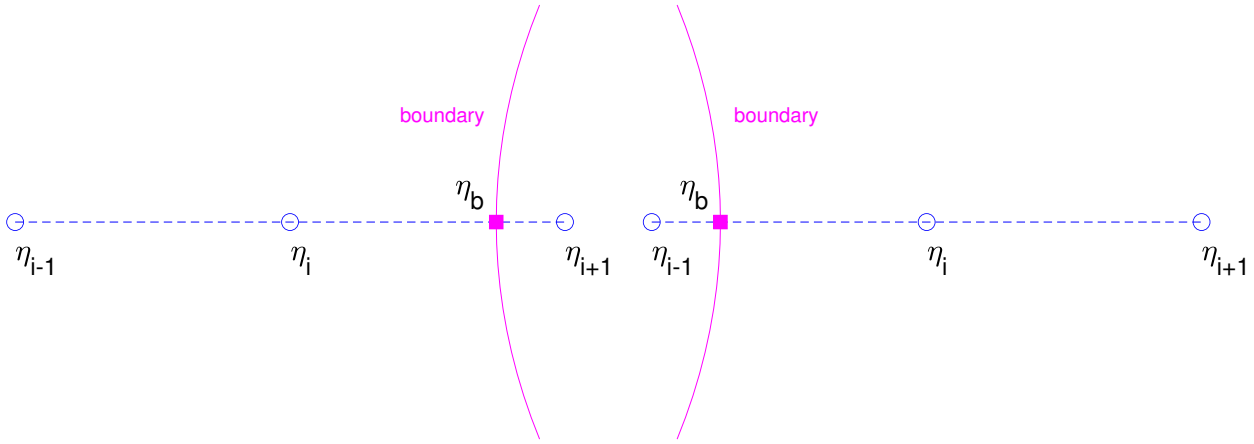


Figure 4: Two possible configurations for a stencil near the immersed boundary ($(\eta_{i-1}, \eta_i, \eta_{i+1})$: grid nodes and η_b: boundary node). They share a common feature: there are two grid nodes in the physical domain and one grid node in the extension domain.

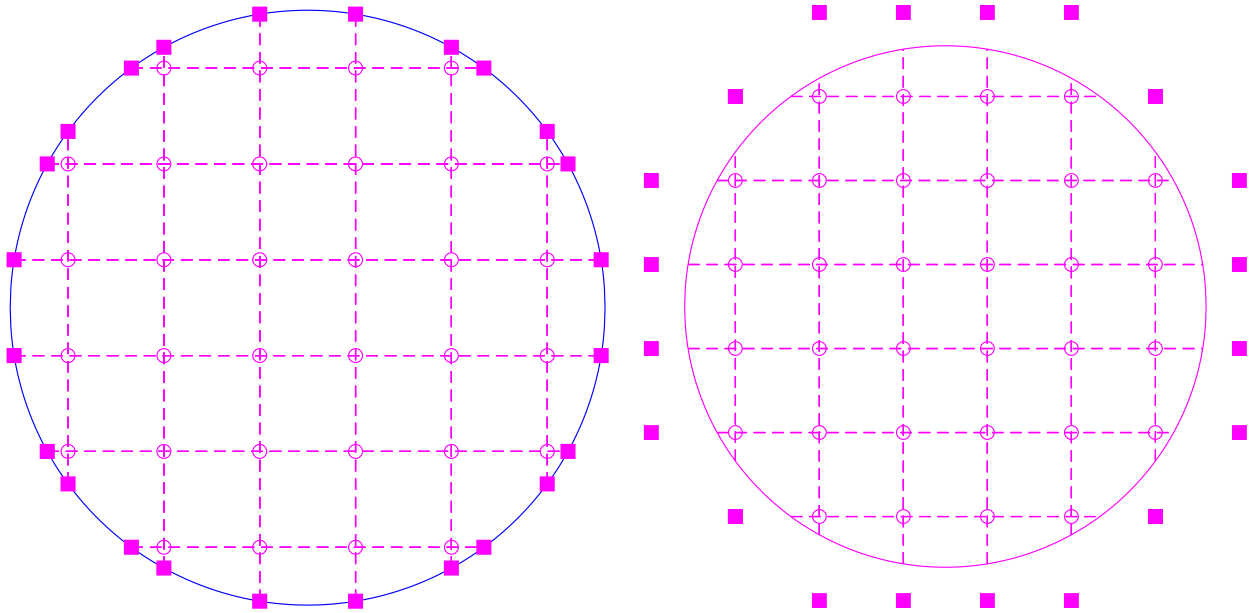


Figure 5: Two configurations of nodes representing the hole considered: one involving the physical boundary points and grid nodes (left) and the other involving only grid nodes (right). The boundary nodes (filled squares) are the physical boundary points for the former and outside grid points adjacent to the boundary for the latter. It is easier to implement the discretization for the latter.

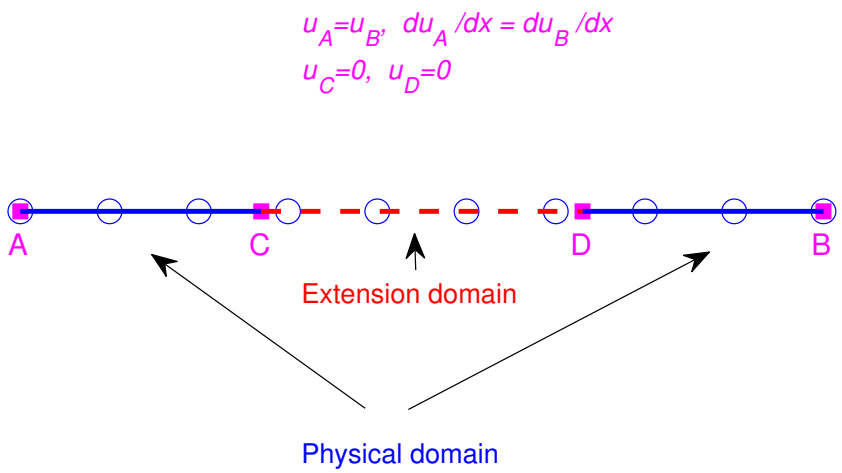


Figure 6: A one-dimensional domain and its associated discretization. The computational domain is comprised of the physical and extension domains. A, B, C and D are boundary points.

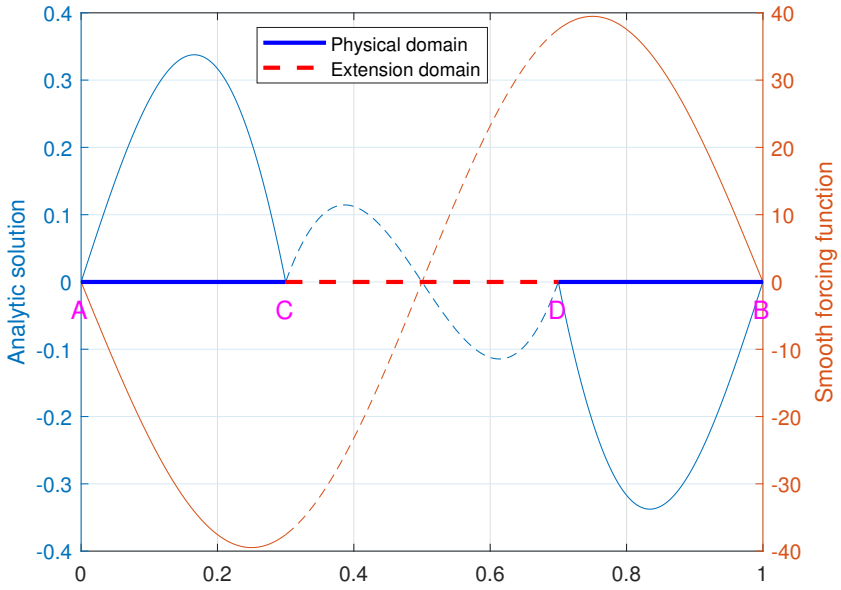


Figure 7: 1D problem, $d^2u/dx^2 = -4\pi^2 \sin(2\pi x)$: analytic solution for the case of using a given smooth forcing function $f(x)$ in the entire computational domain. It can be seen that the analytic solution is not smooth across the interfaces C and D.

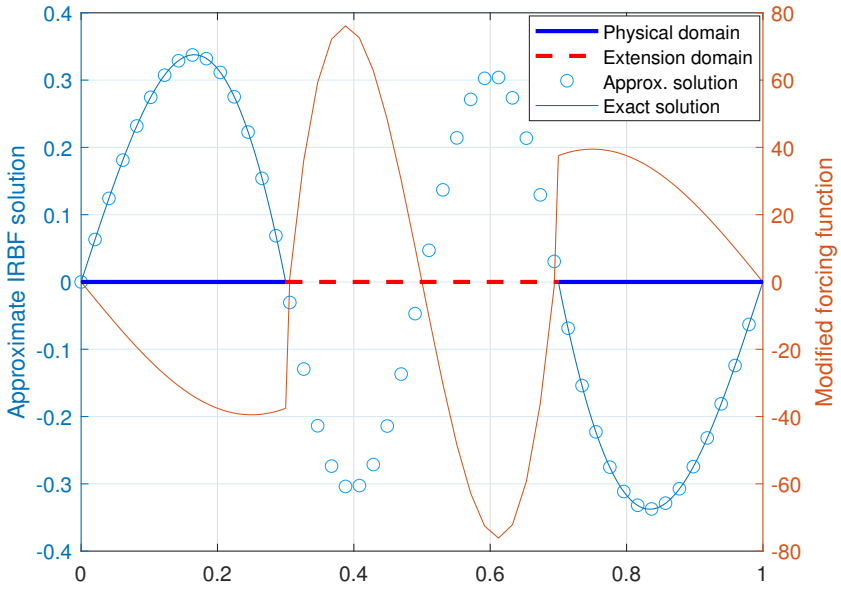


Figure 8: 1D problem, $d^2u/dx^2 = -4\pi^2 \sin(2\pi x)$, $N_x = 50$: A forcing function in the extension domain is modified to give a solution that is C^2 continuous across the two interfaces. It can be seen that the forcing function is no longer smooth across the interfaces.

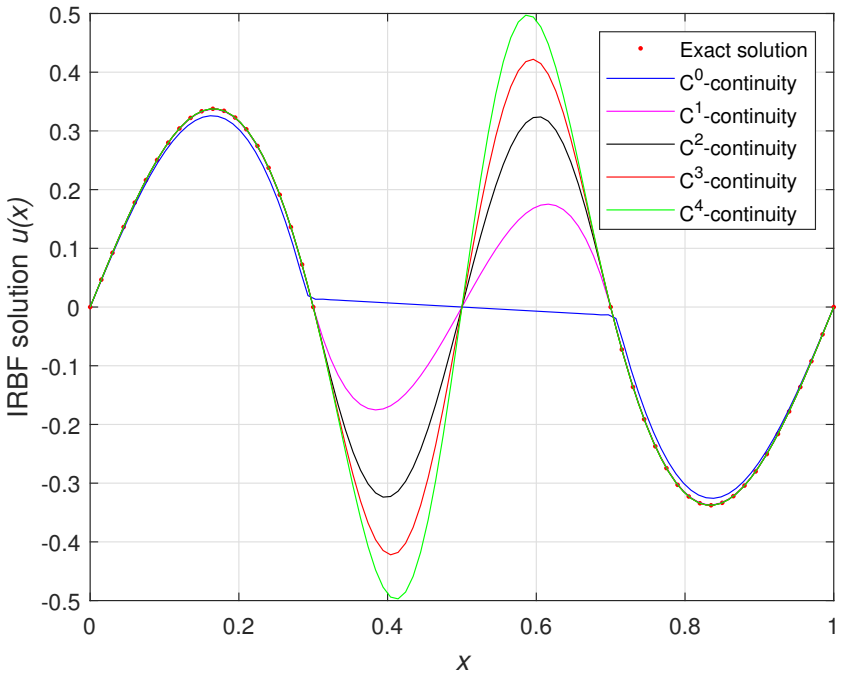


Figure 9: 1D problem, $N_x = 100$: Approximate solutions for the case of using non-smooth forcing functions. A forcing function on the extension domain is modified to give solutions that are C^0, C^1, C^2, C^3 and C^4 continuous across the two interfaces. The exact solution in the physical domain is also included for comparison purposes.

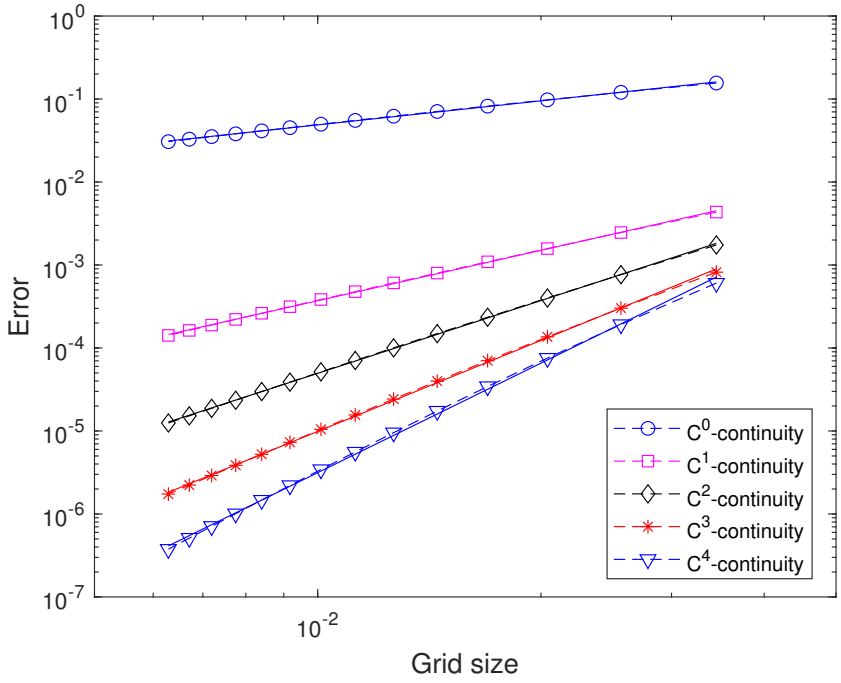


Figure 10: 1D problem, $N_x = (30, 60, \cdots, 160)$: Error versus grid size for different levels of continuity of the solution across the interfaces. The solution in the physical domain converges as $O(h^{1.00})$, $O(h^{2.0})$, $O(h^{2.9})$, $O(h^{3.6})$, $O(h^{4.4})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively. For these results, the continuity conditions are imposed at the inner boundary points.

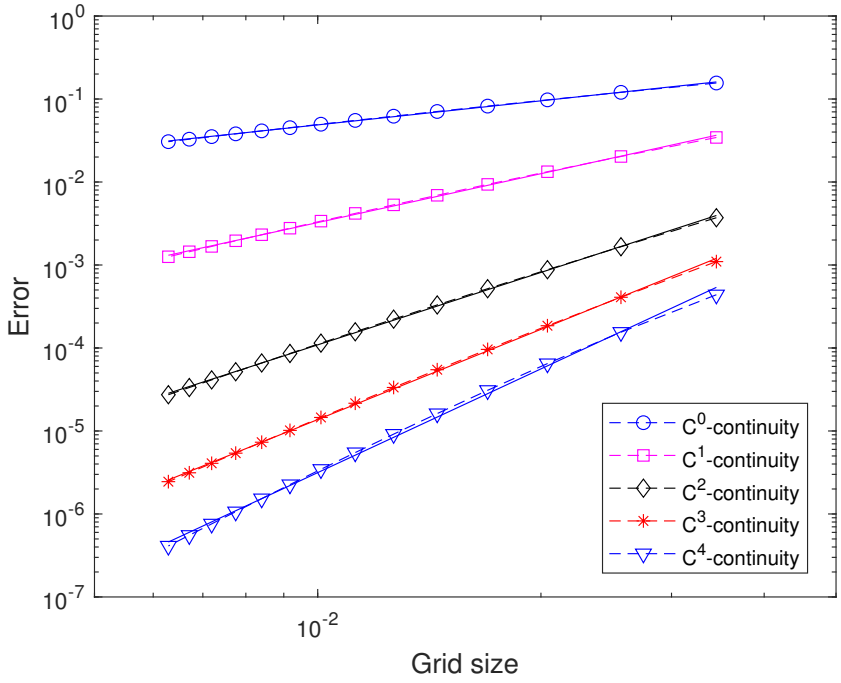


Figure 11: 1D problem, $N_x = (30, 60, \cdots, 160)$: Error versus grid size for different levels of continuity of the solution across the interfaces. The solution in the physical domain converges as $O(h^{1.0})$, $O(h^{2.0})$, $O(h^{2.9})$, $O(h^{3.6})$, $O(h^{4.2})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively. For these results, the continuity conditions are imposed at the grid nodes adjacent to the inner boundary.

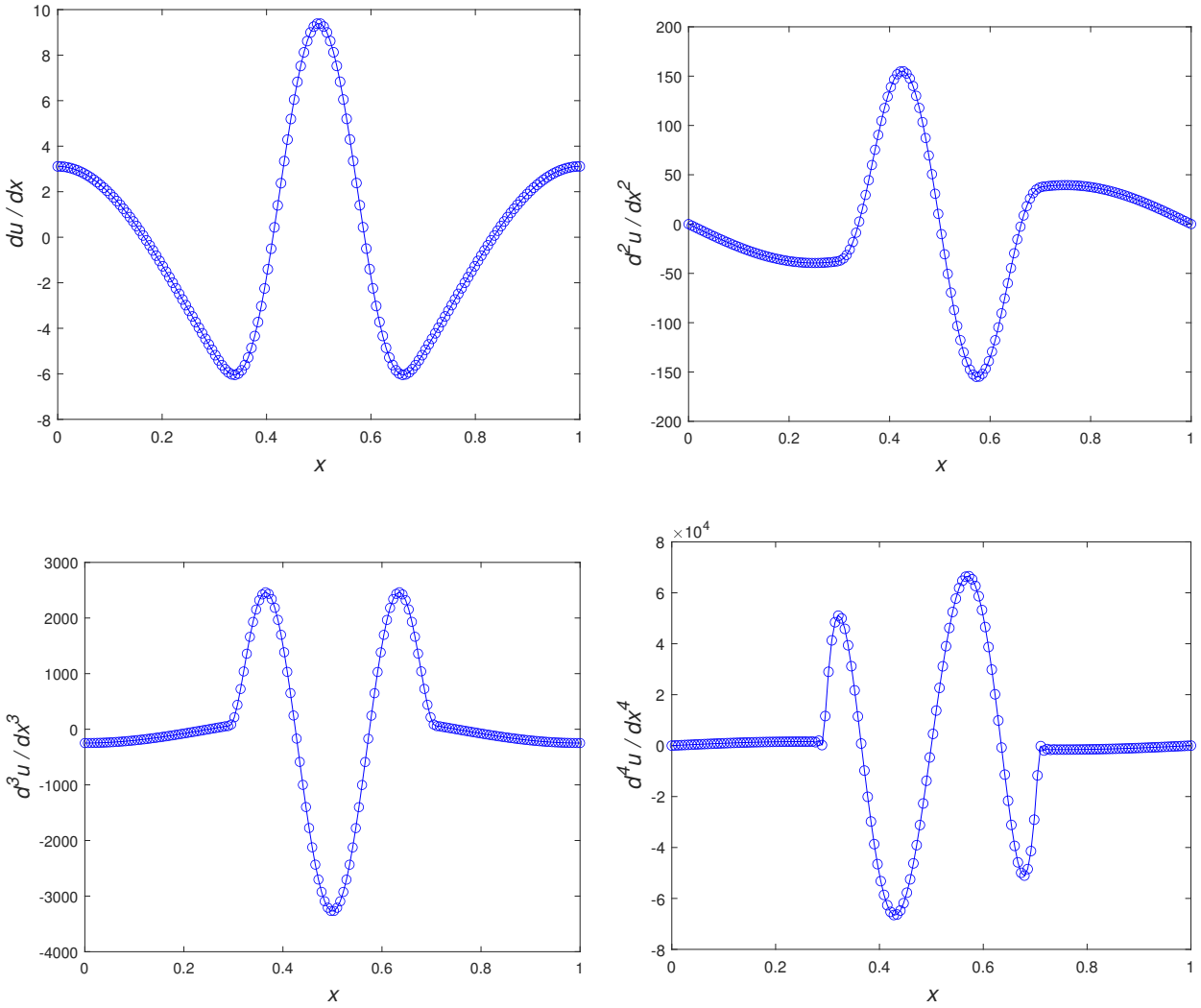


Figure 12: 1D problem, $N_x = 160$: the first-, second-, third- and fourth-order derivatives of the solution in the computational domain when the global C^4 regularity is imposed.

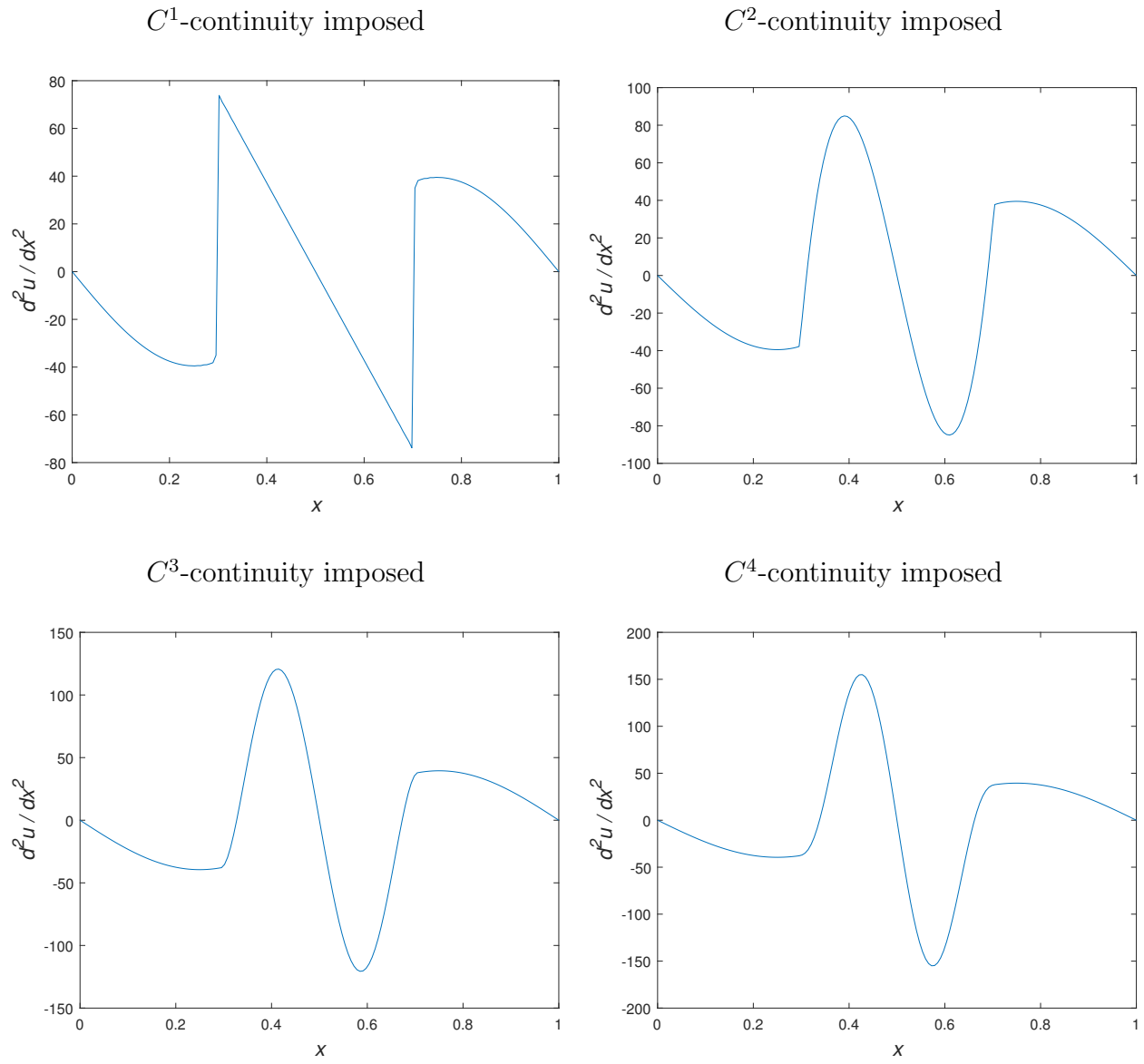


Figure 13: 1D problem, $N_x = 160$: the second-order derivatives of the solution in the computational domain for different levels of the global regularity. The smoothness of the solution is improved with an increase in the level of the global regularity.

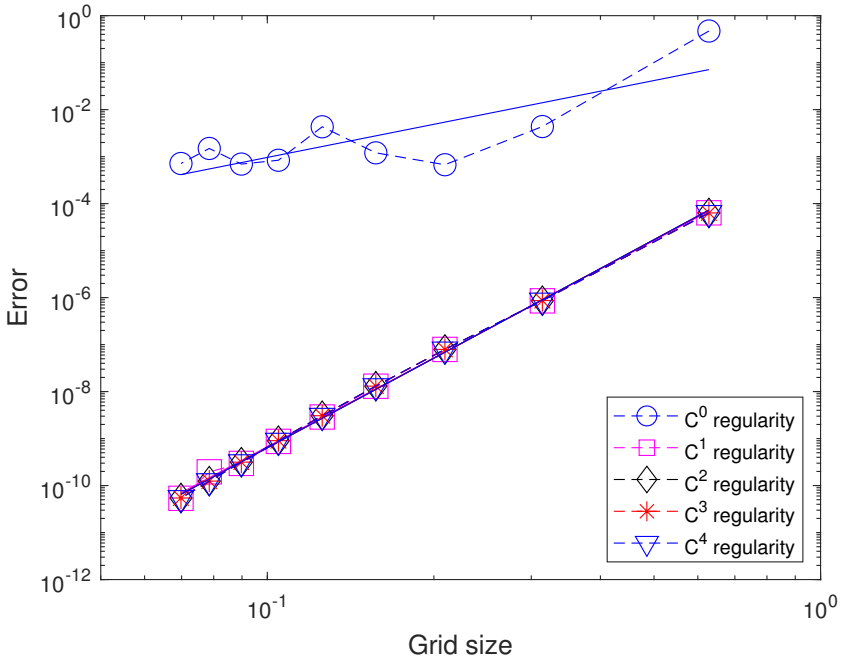


Figure 14: Example 2, $0 \leq x, y \leq 2\pi$, one hole of radius 0.4, $(11\times11, 21\times21, \ldots, 91\times91)$: the solution in the physical domain converges as $O(h^{2.1})$, $O(h^{6.4})$, $O(h^{6.4})$, $O(h^{6.4})$ and $O(h^{6.4})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively.

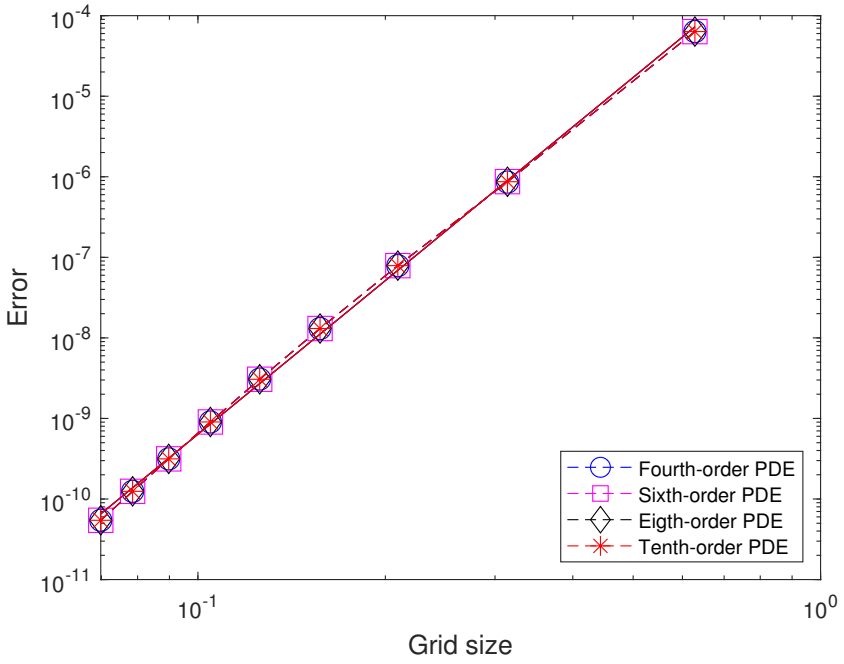


Figure 15: Example 2, $0 \leq x, y \leq 2\pi$, one hole of radius 0.4, $(11 \times 11, 21 \times 21, \ldots, 91 \times 91)$, global C^4 regularity: the solution in the physical domain converges as $O(h^{6.4})$, $O(h^{6.4})$, $O(h^{6.4})$ and $O(h^{6.4})$ when the order of the PDE in the hole is chosen as 4, 6, 8 and 10, respectively.

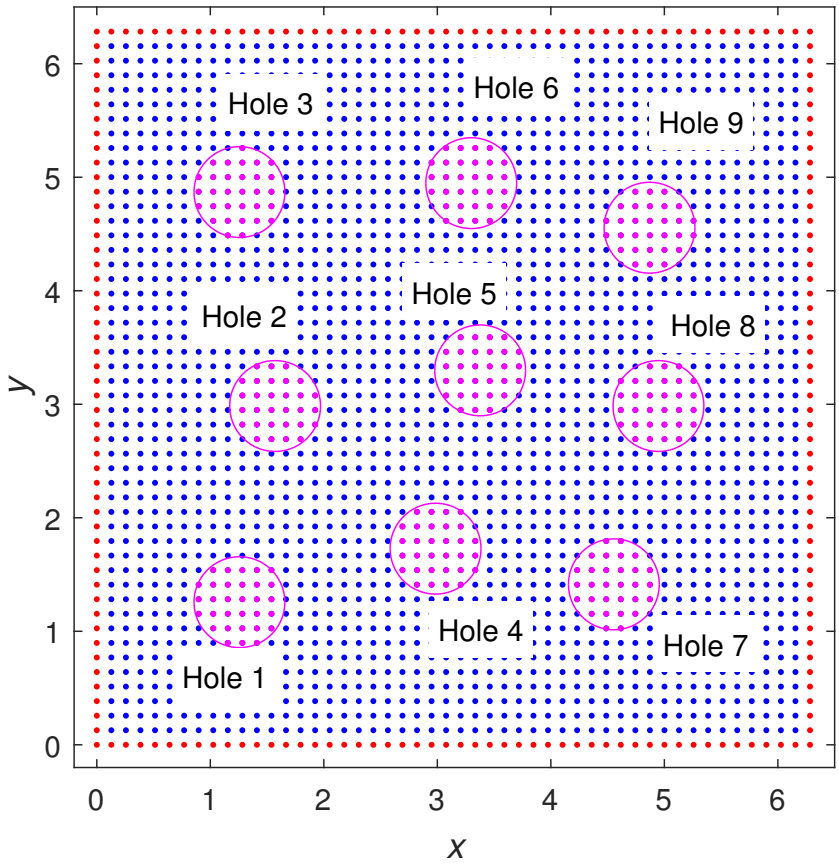


Figure 16: 2D problems, $0 \leq x, y \leq 2\pi$: a domain with 9 holes of radius 0.4

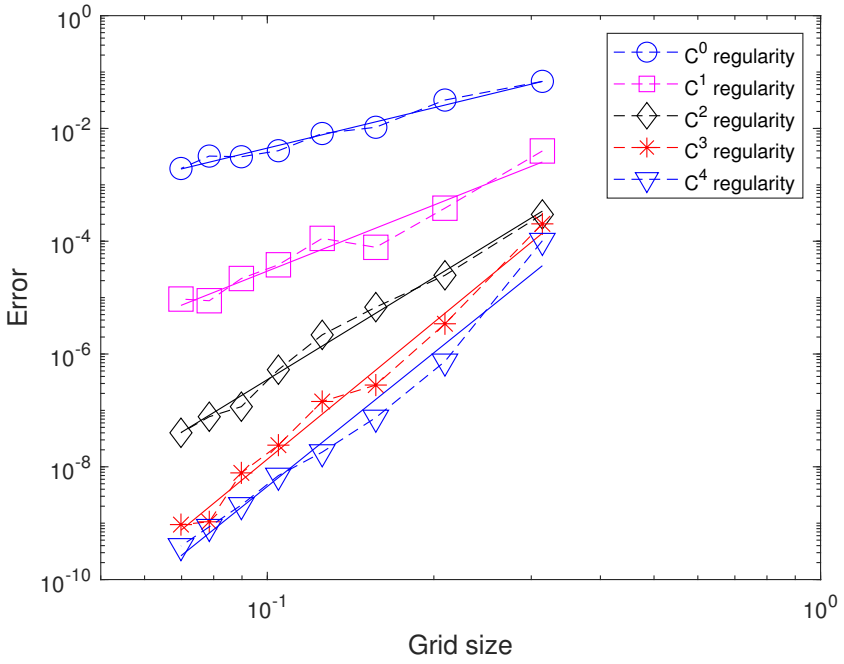


Figure 17: Example 3, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4, $(21\times 21, 31\times 31, \ldots, 91\times 91)$: the solution in the physical domain converges as $O(h^{2.4})$, $O(h^{3.9})$, $O(h^{6.0})$, $O(h^{8.1})$ and $O(h^{7.9})$ for the C^0-, C^1-, C^2-, C^3- and C^4-continuity impositions, respectively.

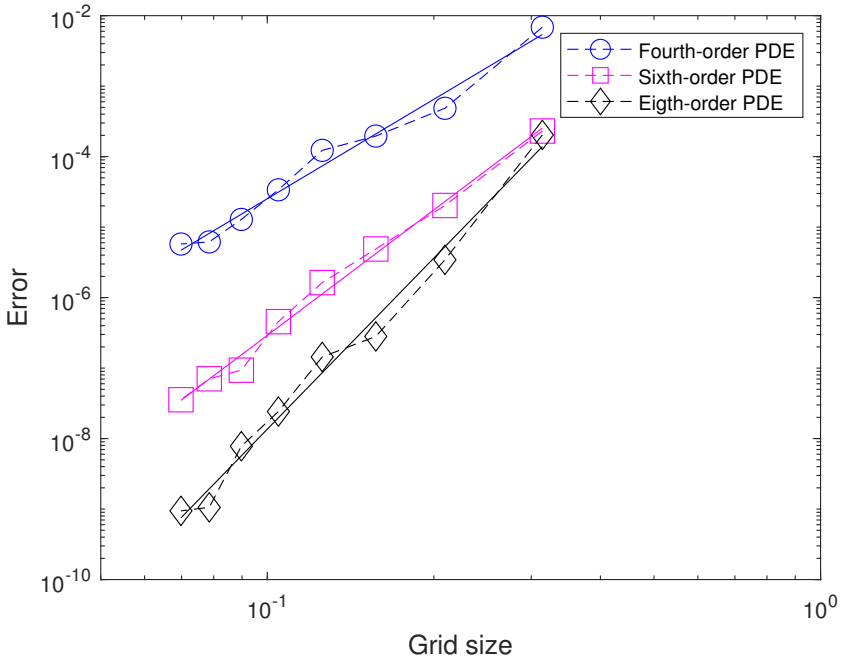


Figure 18: Example 3, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4, global C^3 regularity, $(21 \times 21, 31 \times 31, \ldots, 91 \times 91)$: the solution in the physical domain converges as $O(h^{4.7})$, $O(h^{5.9})$, and $O(h^{8.1})$ when the order of the PDE in the extension domain is chosen as 4, 6 and 8, respectively.

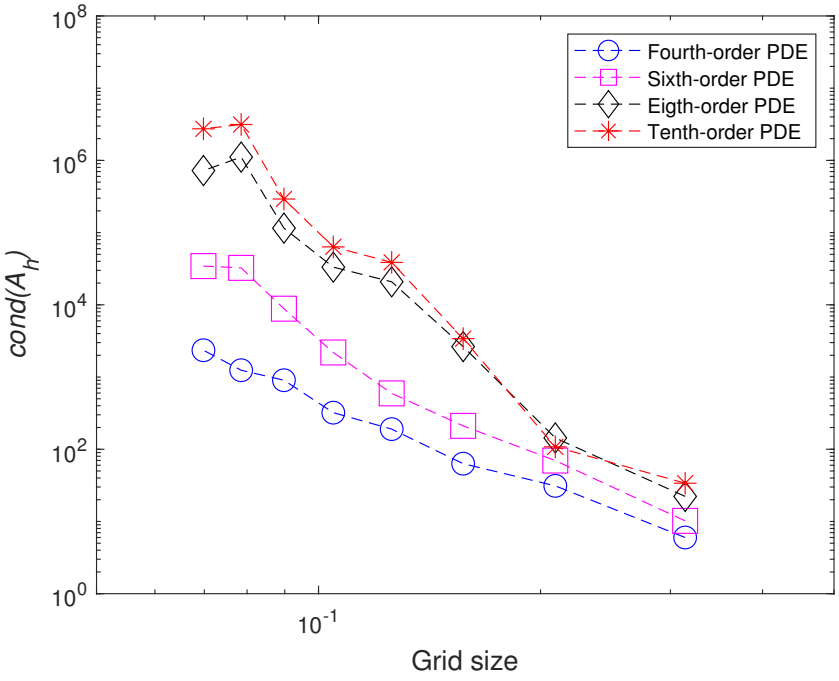


Figure 19: Example 3, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4, $(21 \times 21, 31 \times 31, \ldots, 91 \times 91)$, global C^4 regularity: the condition number of the system matrix associated with Hole 1 when the order of the PDE in the extension domain is chosen as 4, 6, 8 and 10, respectively.

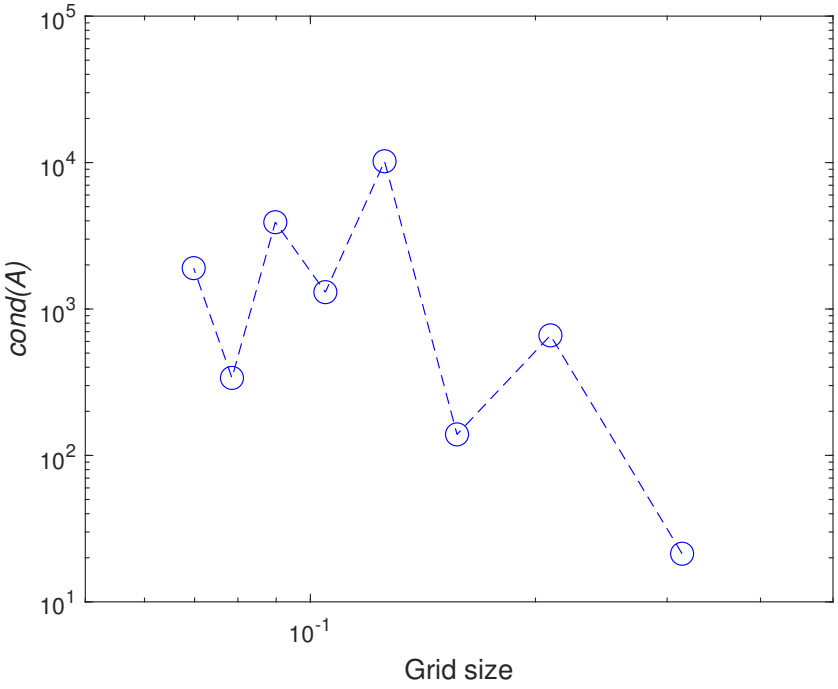


Figure 20: Example 3, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4, $(21 \times 21, 31 \times 31, \ldots, 91 \times 91)$: the condition number of the system matrix associated with the entire computational domain.

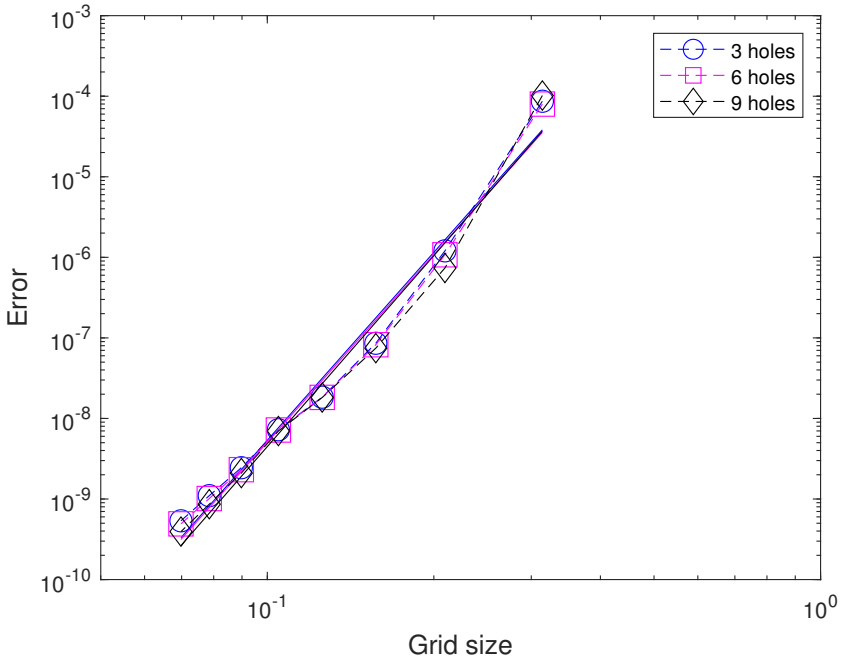


Figure 21: Example 3, $0 \leq x, y \leq 2\pi$, $21 \times 21, \ldots, 91 \times 91$, global C^4 regularity, tenth-order PDE in the extension domain: error versus grid size for the domain with 3, 6 and 9 holes of radius 0.4. They have similar performances, achieving $O(h^{7.7})$.

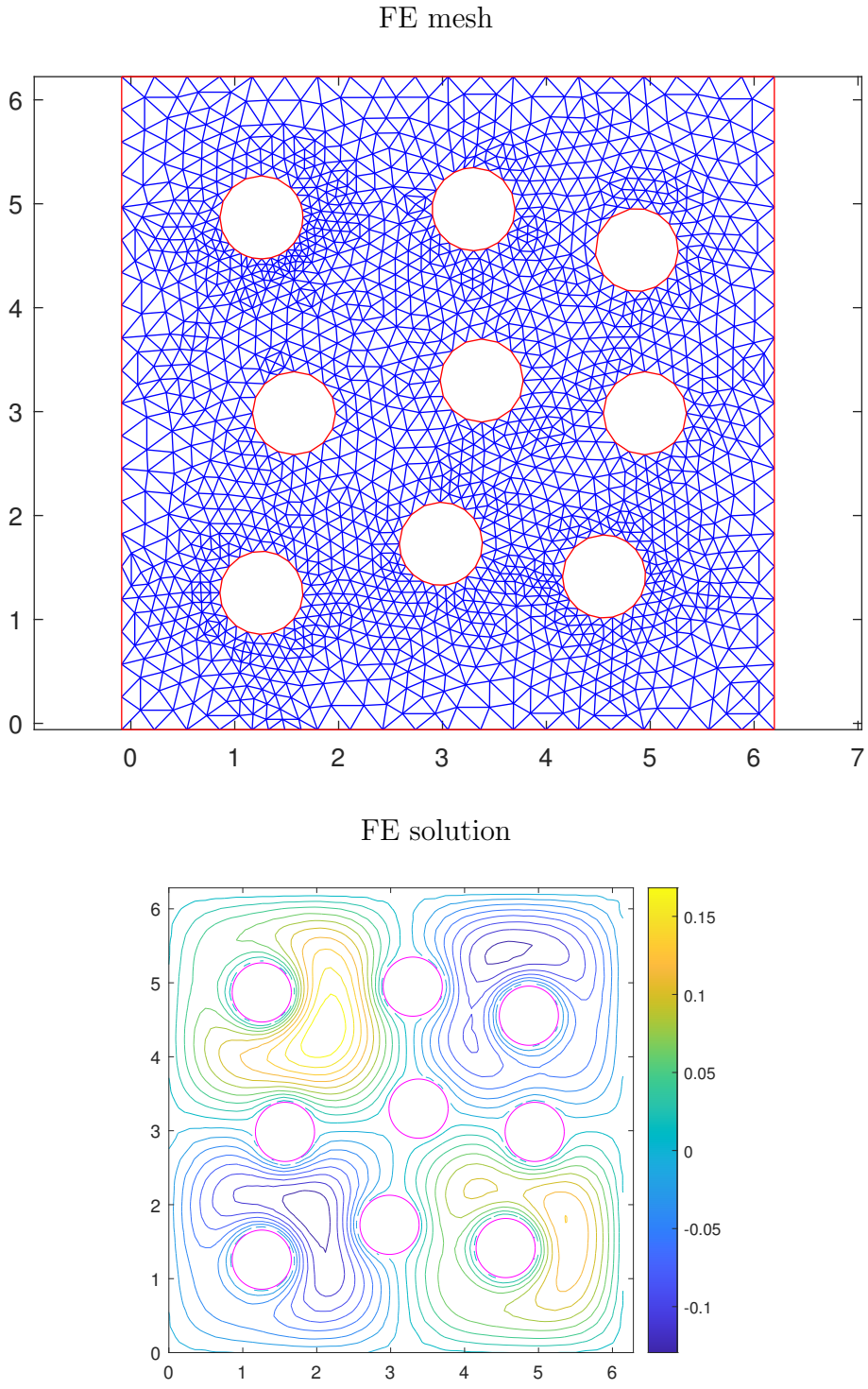


Figure 22: Example 4, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4: Finite-element (FE) results

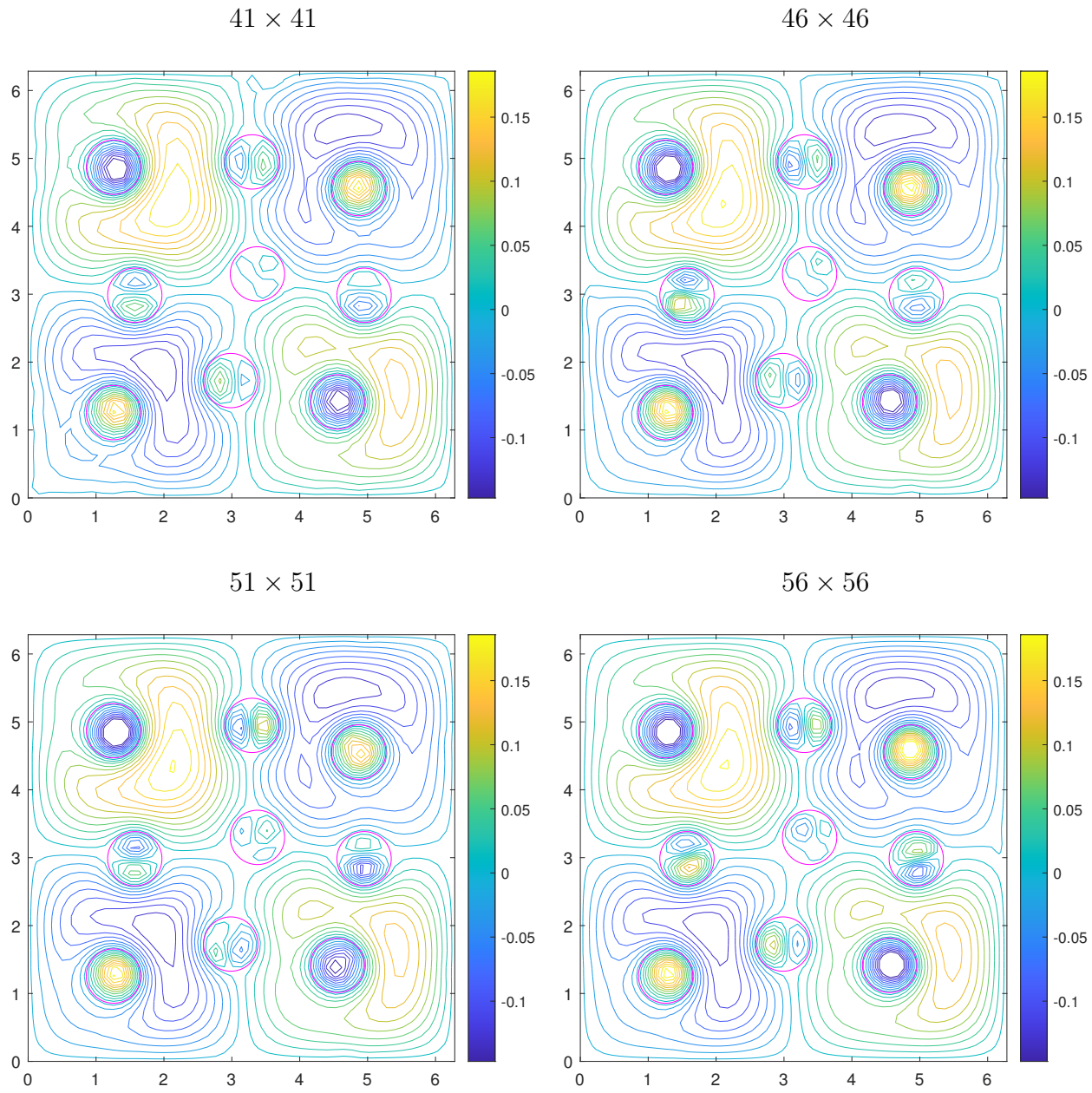


Figure 23: Example 4, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4, global C^4 regularity, fourth-order PDE in the extension domain: contour plots of the solution in the computational domain with different grids.

FE solution IRBF solution

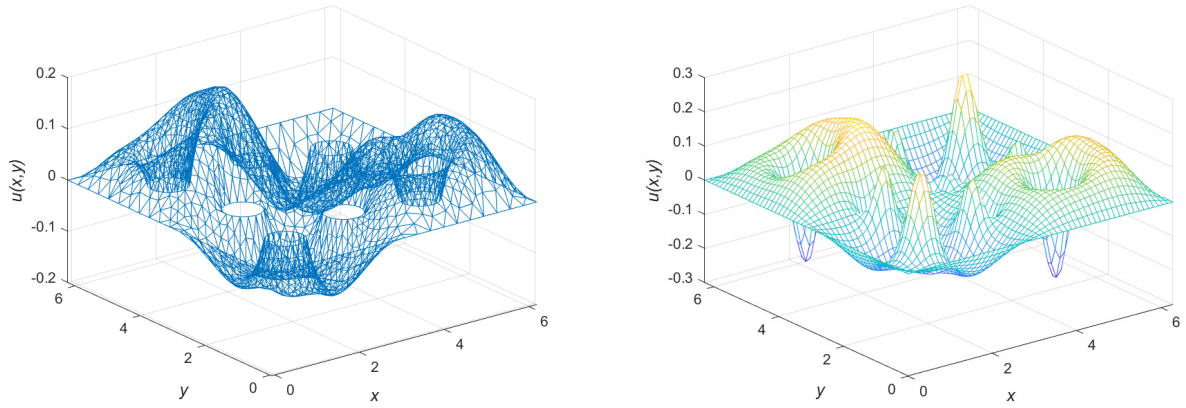
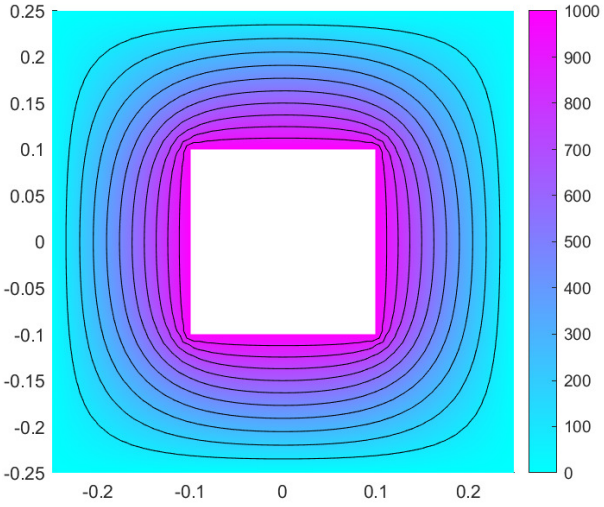
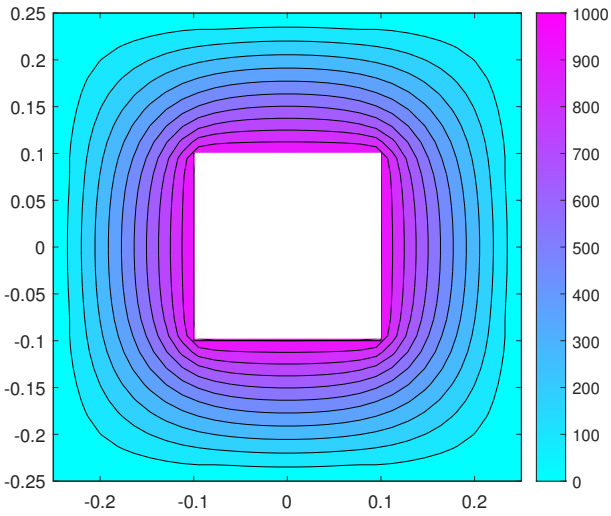


Figure 24: Example 4, $0 \leq x, y \leq 2\pi$, 9 holes of radius 0.4, global C^4 regularity, fourth-order PDE in the extension domain: FE and IRBF solutions. The solution is smooth in the entire computational domain for the latter but not defined in the holes for the former.

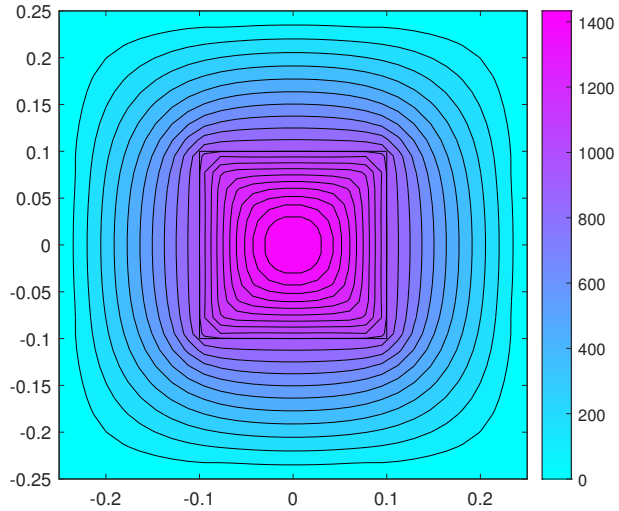
FEM solution



IRBF solution in Ω



IRBF solution in C



IRBF solution in C

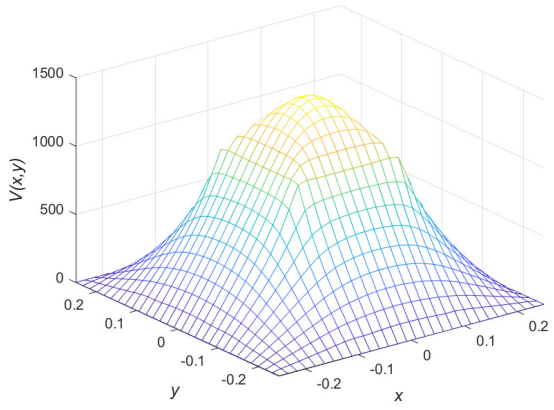


Figure 25: Example 5, $-0.25 \leq x, y \leq 0.25$, square hole of $-0.1 \leq x, y \leq 0.1$, 30×30, global C^2 regularity, fourth-order PDE in the extension domain, non-smooth immersed boundary: FE and IRBF solutions. The solution is smooth in the entire computational domain for the latter but not defined in the square hole for the former.

FEM solution IRBF solution

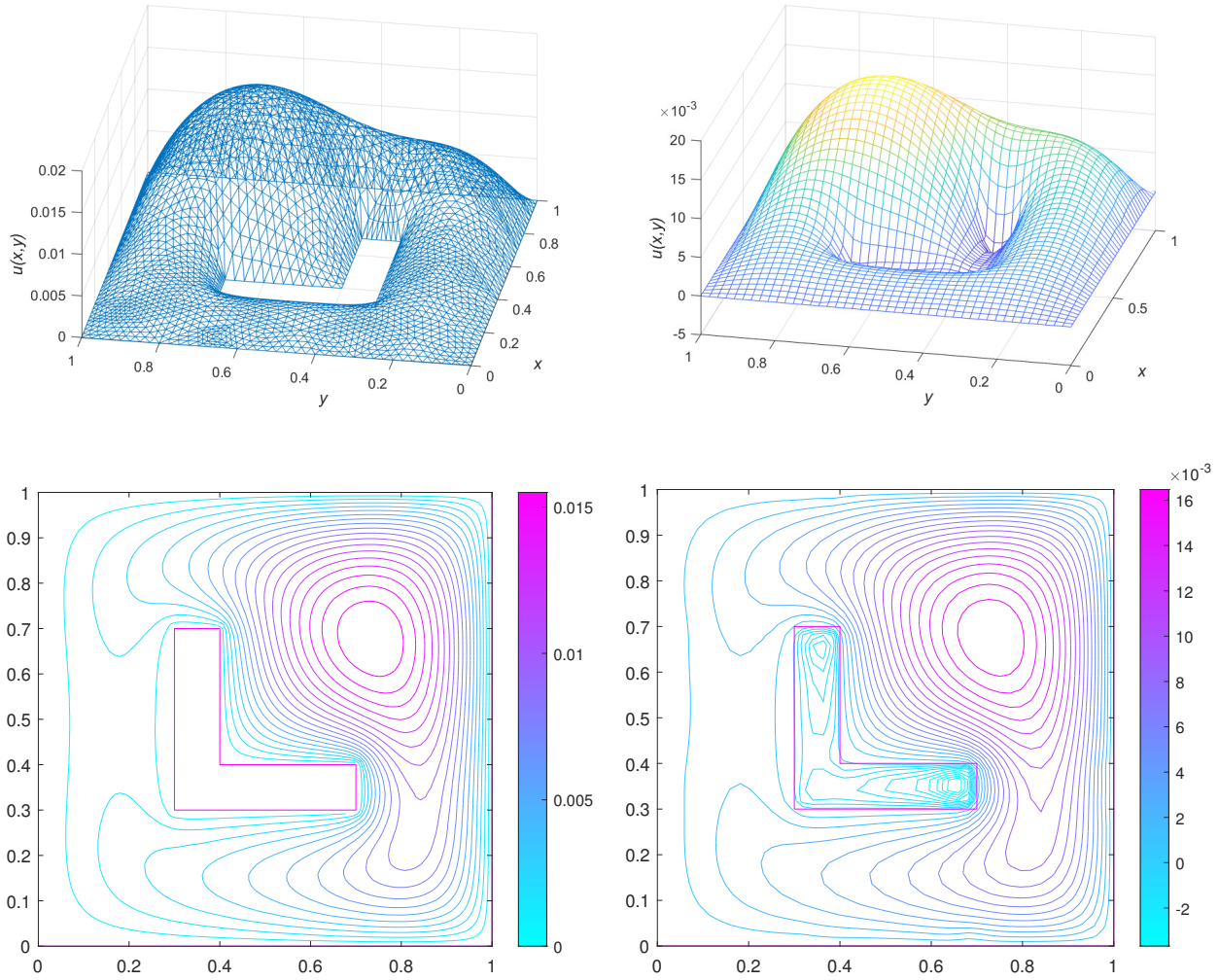


Figure 26: Example 6, $0 \leq x, y \leq 1$, L-shaped hole, 45×45, global C^1 regularity, fourth-order PDE in the extension domain, non-smooth immersed boundary: FE (left) and IRBF (right) solutions. The solution is smooth in the entire computational domain for the latter but not defined in the hole for the former.