

A kernel-free boundary integral method for elliptic PDEs on a doubly-connected domain

Yue Cao · Yaning Xie · Mahesh
Krishnamurthy · Shuwang Li · Wenjun
Ying

Received: date / Accepted: date

Abstract We present a kernel-free boundary integral method (KFBIM) for solving variable coefficients partial differential equations (PDEs) in a doubly-connected domain. We focus our study on boundary value problems (BVP) and interface problems. A unique feature of the KFBIM is that the method does not require an analytical form of the Green's function for designing quadratures, but rather computes boundary or volume integrals by solving an equivalent interface problem on Cartesian mesh. We first decompose the problem defined in a doubly-connected into two separate interface problems. The system of boundary integral equations is solved using a Krylov method. The method is second-order accurate in space, and its complexity is linearly proportional to the number of mesh points. Numerical examples demonstrate that the method is robust for variable coefficients PDEs, even for cases with large diffusion coefficients ratio and complex geometries where two interfaces are close.

Keywords Elliptic partial differential equation · Variable coefficients · Kernel-free boundary integral method · Doubly-connect domain

Y. Cao and S. Li

Department of Applied Mathematics, Illinois Institute of Technology, Chicago, United States

E-mail: ycao33@hawk.iit.edu, sli15@iit.edu

Y. Xie

College of Science, Zhejiang University of Technology, Xihu, Hangzhou 310023, P. R. China

E-mail: xieyaning2015@sjtu.edu.cn

M. Krishnamurthy

Department of Electrical and Computer Engineering, Illinois Institute of Technology, Chicago, United States

E-mail: kmahesh@iit.edu

W. Ying *Corresponding author*

School of Mathematical Sciences, MOE-LSC and Institute of Natural Science, Shanghai Jiao Tong University, Shanghai, China.

E-mail: wying@sjtu.edu.cn

1 Introduction

There are many computational methods that have been developed for solving boundary value or interface problems, e.g. [1,2,3,4,5]. Among these methods, the boundary integral method is able to treat the boundary or interface conditions exactly and usually considered to be the most accurate method, provided well-developed accurate and stable quadratures of boundary integrals are available. Moreover, after integral formulations, the dimensionality of the PDE problem is reduced by one. Namely, a two-dimensional domain problem can be solved via the integration on a one-dimensional curve, thus computational cost is reduced dramatically. Though relatively few, there are stability and convergence proofs of boundary integral methods for interface problems [6,7,8]. Numerically, the discretization of integrals usually leads to full dense matrix that can be solved using an iterative method (e.g. GMRES). In GMRES, a direct summation method to compute the matrix-vector multiplication requires $O(N^2)$ operations, where N is the number of computational points on the boundary. In practice, a fast summation method is used to reduce this computation cost from $O(N^2)$ to $O(N)$ or $O(N\log N)$ [9,10,11,12,13,14].

The KFBIM in [17,18] is a generalization of the traditional boundary integral methods and can be interpreted as the grid-based integral methods by Mayo [19,20]. A unique feature of KFBIM scheme is that we do not require the explicit form of the Green's function or special quadratures to directly evaluate integrals, especially nearly singular, singular or hyper-singular boundary integrals. Instead, the idea behind KFBIM is to reinterpret boundary integrals as solutions to equivalent simple interface problems posed in a rectangle box, which can be solved efficiently by a finite difference method coupled with numerical corrections (to gain accuracy), FFT based solution and interpolations (to gain efficiency). The KFBIM has been developed to be a general method for elliptic PDEs in two and three dimensions [21,22,23,24,25,26].

A natural question is why the boundary integrals are introduced, as the original elliptic PDE can be directly solved by finite differences, e.g. the Shortley-Weller method [15] and Gibou's finite difference method [16]. The resulting discrete system of equations by direct finite difference methods are typically ill-conditioned, requiring much more computational work or making the solution sensitive to computer round-off or other errors. It is also not easy (as least not obvious) to apply fast elliptic solvers to solve those finite difference systems. The idea of reformulating the elliptic PDEs as a boundary integral equation, or a system of integral equations (usually the Fredholm integral equations of the second kind) by the KFBIM method, helps remove the ill-conditioning property compared with the traditional finite difference approaches. The resulting system by the KFBIM scheme is well-conditioned, requiring only a fixed number of iterations when an iterative method is applied, which is an excellent property that the previous methods do not have, and even can be solved by fast elliptic solvers. In addition to the solution of the PDE, the KFBIM also produces the results of layer densities at the boundary or interface, which is a useful quantity to consider in practice.

In this paper, we extend the KFBIM for solving variable coefficients elliptic PDEs in doubly-connected or annulus domains (Fig. 1). Typical applications in such domains include Hele-Shaw flows with coupled interface [27, 28, 29, 30], design of electric machines [31], and simulation of tumor growth [32, 33]. Depending on the modeling purpose, one may specify a variety of conditions on the boundary, e.g. Dirichlet and Neumann conditions for BVPs, or jump conditions for interface problems. Numerical challenges arise when the diffusion coefficients ratio for connected domains gets large or the two boundary or interfaces get close.

We first decompose the given PDE problems into two interface problems and then rewrite them into a linear system using well defined single or double layer integrals. We solve these integrals on a Cartesian grid based method, e.g. a finite difference scheme. To improve the accuracy of the integral evaluation, we need corrections only for points near the interface. The resulting linear system is then solved iteratively using a Krylov subspace method (GMRES) [34, 35]. During each iteration, a geometric multigrid preconditioned conjugate gradient method (GMG-PCG) is used to solve the equivalent interface problem. Numerical experiments show that iteration numbers of GMRES or GMG-PCG are independent of mesh resolution. The computation CUP time is linearly proportional to mesh node number. The second order accuracy in space is demonstrated for constant coefficients and variable coefficients PDEs. Numerical results show that the KFBIM is efficient in handling cases with closely packed interfaces.

The organization of this paper is as follow. In Section 2, we present variable coefficients partial differential equations (PDEs) in a doubly-connected domain. In Section 3, we define boundary integrals. In Section 4, we present the formulation of integral equations. In Section 5, we present details on computing boundary and volume integral on a Cartesian mesh. In Section 6, we demonstrate accuracy and efficiency with numerical examples. In last section, we conclude and discuss potential applications of the method.

2 Elliptic PDEs in a doubly-connected domain

2.1 Boundary value problem

Let $\Omega_1, \Omega_2 \subset \mathbb{R}^d$ ($d = 2$ or 3) be domains enclosed respectively by smooth boundary Γ_1 and Γ_2 . $\Omega_i = \overline{\Omega}_1^c \cap \Omega_2$ is a doubly-connected domain with smooth boundaries Γ_1 and Γ_2 (see Fig. 1a). We consider the following boundary value problem,

$$\mathcal{A}_i u \equiv \nabla \cdot (\sigma_i(\mathbf{p}) \nabla u) - \kappa_i(\mathbf{p}) u = f_i(\mathbf{p}) \quad \text{in } \Omega_i, \quad (2.1)$$

$$\sigma_i \partial_n u = g^N \quad \text{on } \Gamma_1, \quad (2.2)$$

$$u = g^D \quad \text{on } \Gamma_2, \quad (2.3)$$

where variable coefficients $\sigma_i(\mathbf{p}) > 0$ and $\kappa_i(\mathbf{p}) \geq 0$ are smooth spatially. $f_i(\mathbf{p})$ is a smooth source function in Ω_i , and u is the unknown function to solve.

$\mathcal{A}_i$ is the differential operator of the PDE. g^D and g^N are the Dirichlet and Neumann boundary conditions. $\mathbf{n}$ denotes the unit outward normal vector on each boundary. $\partial_n u$ is the normal derivative of u .

2.2 Interface problem

Let $\mathcal{B} \subset \mathbb{R}^d$ ($d = 2$ or 3) be a rectangle box. Γ_1 and Γ_2 are smooth interfaces in $\mathcal{B}$ and partition the box into three subdomains, Ω_1 , Ω_2 and Ω_3 . They are three layers from most inside to most outside. The notations are different from section 2.1. $\partial\overline{\Omega}_1 \cap \partial\overline{\Omega}_2 = \Gamma_1$, $\partial\overline{\Omega}_2 \cap \partial\overline{\Omega}_3 = \Gamma_2$ (See Fig. 1b). We consider the following interface problem,

$$\mathcal{A}_1 u_1 \equiv \nabla \cdot (\sigma_1(\mathbf{p}) \nabla u_1) - \kappa_1(\mathbf{p}) u_1 = f_1(\mathbf{p}) \quad \text{in } \Omega_1, \quad (2.4)$$

$$\mathcal{A}_2 u_2 \equiv \nabla \cdot (\sigma_2(\mathbf{p}) \nabla u_2) - \kappa_2(\mathbf{p}) u_2 = f_2(\mathbf{p}) \quad \text{in } \Omega_2, \quad (2.5)$$

$$\mathcal{A}_3 u_3 \equiv \nabla \cdot (\sigma_3(\mathbf{p}) \nabla u_3) - \kappa_3(\mathbf{p}) u_3 = f_3(\mathbf{p}) \quad \text{in } \Omega_3, \quad (2.6)$$

$$u_1 - u_2 = g_1 \quad \text{and} \quad \sigma_1 \partial_n u_1 - \sigma_2 \partial_n u_2 = j_1 \quad \text{on } \Gamma_1, \quad (2.7)$$

$$u_2 - u_3 = g_2 \quad \text{and} \quad \sigma_2 \partial_n u_2 - \sigma_3 \partial_n u_3 = j_2 \quad \text{on } \Gamma_2, \quad (2.8)$$

$$u_3 = 0 \quad \text{on } \partial\mathcal{B}, \quad (2.9)$$

where variable coefficients $\sigma_1, \sigma_2, \sigma_3 > 0$ and $\kappa_1, \kappa_2, \kappa_3 \geq 0$ are smooth spatially. We assume $\sigma_1, \sigma_2, \sigma_3$ are continuous differentiable in $\mathcal{B}$ and $\kappa_1, \kappa_2, \kappa_3$ are continuous in $\mathcal{B}$. f_1, f_2, f_3 are smooth source functions in $\mathcal{B}$. u_1, u_2, u_3 are the unknown functions to solve. $\mathcal{A}_i$ is the differential operator of PDE in Ω_i ($i = 1, 2, 3$). $\mathbf{n}$ denotes the unit outward normal vector on each interface.

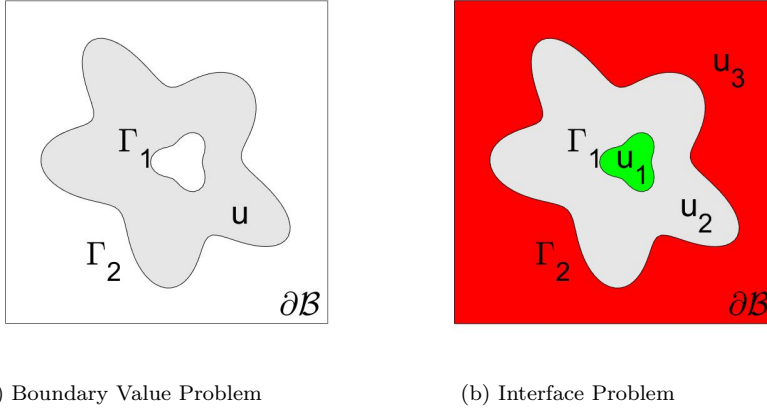


Fig. 1: Schematic diagrams of a doubly-connected domain for boundary value problem (a) and interface problem (b).

3 Boundary integrals and volume integrals

3.1 Boundary value problem

Let $G_i(\mathbf{p}, \mathbf{q})$ be the corresponding Green's function to the PDEs in Eqs. (2.1) to (2.3). $\nabla_{\mathbf{p}}$ is the gradient operator with respect to spatial variable $\mathbf{p} \in \mathbb{R}^d$. $\delta(\mathbf{p} - \mathbf{q})$ is the Dirac delta function. The Green function satisfies

$$\begin{aligned} \mathcal{A}_i G_i(\mathbf{p}, \mathbf{q}) &= \nabla_{\mathbf{p}} \cdot (\sigma_i(\mathbf{p}) \nabla_{\mathbf{p}} G_i(\mathbf{p}, \mathbf{q})) - \kappa_i(\mathbf{p}) G_i(\mathbf{p}, \mathbf{q}) = \delta(\mathbf{p} - \mathbf{q}) && \text{in } \mathcal{B}, \\ G_i(\mathbf{p}, \mathbf{q}) &= 0 && \text{on } \partial\mathcal{B}. \end{aligned}$$

$\mathbf{n}_{\mathbf{p}}$ is the unit outward normal vector at $\mathbf{p} \in \Gamma_1$. Using a density function ψ_1 , we define the following single layer boundary integral

$$\mathcal{L}_i \psi_1(\mathbf{p}) \equiv \int_{\Gamma_1} G_i(\mathbf{p}, \mathbf{q}) \psi_1(\mathbf{q}) ds_{\mathbf{q}} \quad (3.1)$$

and a adjoint double layer boundary integral

$$\mathcal{M}_i^* \psi_1(\mathbf{p}) = \sigma_i \partial_n (\mathcal{L}_i \psi_1) \equiv \int_{\Gamma_1} \sigma_i(\mathbf{p}) \frac{\partial G_i(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{p}}} \psi_1(\mathbf{q}) ds_{\mathbf{q}}. \quad (3.2)$$

Similarly, we use a density function φ_2 to define a double layer boundary integral

$$\mathcal{M}_i \varphi_2(\mathbf{p}) \equiv \int_{\Gamma_2} \sigma_i(\mathbf{q}) \frac{\partial G_i(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{q}}} \varphi_2(\mathbf{q}) ds_{\mathbf{q}} \quad (3.3)$$

and a hyper singular boundary integral

$$\mathcal{N}_i \varphi_2(\mathbf{p}) = \sigma_i \partial_n (\mathcal{M}_i \varphi_2) \equiv \int_{\Gamma_2} \sigma_i(\mathbf{p}) \sigma_i(\mathbf{q}) \frac{\partial^2 G_i(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{q}} \partial \mathbf{n}_{\mathbf{p}}} \varphi_2(\mathbf{q}) ds_{\mathbf{q}}. \quad (3.4)$$

The volume integral is defined as

$$\mathcal{G}_i f_i(\mathbf{p}) \equiv \int_{\Omega_i} G_i(\mathbf{p}, \mathbf{q}) f_i(\mathbf{q}) d\mathbf{q}. \quad (3.5)$$

3.2 Interface problem

For the interface problem Eqs. (2.4)-(2.9), the Green's functions corresponding to the three PDEs are generally different. We consider $G_1(\mathbf{p}, \mathbf{q})$ is the Green function associated with PDE (2.4) in Ω_1 and satisfies

$$\begin{aligned} \mathcal{A}_1 G_1(\mathbf{p}, \mathbf{q}) &= \delta(\mathbf{p} - \mathbf{q}) && \text{in } \mathcal{B}, \\ G_1(\mathbf{p}, \mathbf{q}) &= 0 && \text{on } \partial\mathcal{B}; \end{aligned}$$

$G_2(\mathbf{p}, \mathbf{q})$ is the Green function associated with PDE (2.5) in Ω_2 and satisfies

$$\begin{aligned} \mathcal{A}_2 G_2(\mathbf{p}, \mathbf{q}) &= \delta(\mathbf{p} - \mathbf{q}) && \text{in } \mathcal{B}, \\ G_2(\mathbf{p}, \mathbf{q}) &= 0 && \text{on } \partial\mathcal{B}; \end{aligned}$$

$G_3(\mathbf{p}, \mathbf{q})$ is the Green function associated with PDE (2.6) in Ω_3 and satisfies

$$\begin{aligned} \mathcal{A}_3 G_3(\mathbf{p}, \mathbf{q}) &= \delta(\mathbf{p} - \mathbf{q}) && \text{in } \mathcal{B}, \\ G_3(\mathbf{p}, \mathbf{q}) &= 0 && \text{on } \partial\mathcal{B}. \end{aligned}$$

Similarly, using density functions φ_1, φ_2 , we define double layer boundary integrals

$$\mathcal{M}_1 \varphi_1(\mathbf{p}) \equiv \int_{\Gamma_1} \sigma_1(\mathbf{q}) \frac{\partial G_1(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{q}}} \varphi_1(\mathbf{q}) ds_{\mathbf{q}}, \quad (3.6)$$

$$\mathcal{M}_2 \varphi_1(\mathbf{p}) \equiv \int_{\Gamma_1} \sigma_2(\mathbf{q}) \frac{\partial G_2(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{q}}} \varphi_1(\mathbf{q}) ds_{\mathbf{q}}, \quad (3.7)$$

$$\mathcal{M}_2 \varphi_2(\mathbf{p}) \equiv \int_{\Gamma_2} \sigma_2(\mathbf{q}) \frac{\partial G_2(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{q}}} \varphi_2(\mathbf{q}) ds_{\mathbf{q}}, \quad (3.8)$$

$$\mathcal{M}_3 \varphi_2(\mathbf{p}) \equiv \int_{\Gamma_2} \sigma_3(\mathbf{q}) \frac{\partial G_3(\mathbf{p}, \mathbf{q})}{\partial \mathbf{n}_{\mathbf{q}}} \varphi_2(\mathbf{q}) ds_{\mathbf{q}}. \quad (3.9)$$

Using density functions ψ_1 and ψ_2 , we define single layer boundary integrals

$$\mathcal{L}_1 \psi_1(\mathbf{p}) \equiv \int_{\Gamma_1} G_1(\mathbf{p}, \mathbf{q}) \psi_1(\mathbf{q}) ds_{\mathbf{q}}, \quad (3.10)$$

$$\mathcal{L}_2 \psi_1(\mathbf{p}) \equiv \int_{\Gamma_1} G_2(\mathbf{p}, \mathbf{q}) \psi_1(\mathbf{q}) ds_{\mathbf{q}}, \quad (3.11)$$

$$\mathcal{L}_2 \psi_2(\mathbf{p}) \equiv \int_{\Gamma_2} G_2(\mathbf{p}, \mathbf{q}) \psi_2(\mathbf{q}) ds_{\mathbf{q}}, \quad (3.12)$$

$$\mathcal{L}_3 \psi_2(\mathbf{p}) \equiv \int_{\Gamma_2} G_3(\mathbf{p}, \mathbf{q}) \psi_2(\mathbf{q}) ds_{\mathbf{q}}. \quad (3.13)$$

The volume integrals are defined as

$$\mathcal{G}_1 f_1(\mathbf{p}) \equiv \int_{\Omega_1} G_1(\mathbf{p}, \mathbf{q}) f_1(\mathbf{q}) d\mathbf{q}, \quad (3.14)$$

$$\mathcal{G}_2 f_2(\mathbf{p}) \equiv \int_{\Omega_1 \cup \Omega_2} G_2(\mathbf{p}, \mathbf{q}) f_2(\mathbf{q}) d\mathbf{q}, \quad (3.15)$$

$$\mathcal{G}_3 f_3(\mathbf{p}) \equiv \int_{\Omega_3} G_3(\mathbf{p}, \mathbf{q}) f_3(\mathbf{q}) d\mathbf{q}. \quad (3.16)$$

Adjoint double layer and hyper-singular boundary integrals operator, denoted as $\mathcal{N}_1, \mathcal{N}_2, \mathcal{M}_1^*$ and $\mathcal{M}_2^*$, are interpreted similarly to Eqs. (3.2) and (3.4) in the previous subsection.

4 Formulation of boundary integral equations

4.1 Boundary value problem

In this section, we rewrite the boundary value problem into its integral form. Let $\Omega_1 \subset R^2$ be a bounded domain with smooth boundary Γ_1 (Fig. 2a). The space outside Γ_1 is denoted as $\overline{\Omega}_1^c$. $u_e(\mathbf{x})$, an unknown function, is the solution to an exterior Neumann BVP,

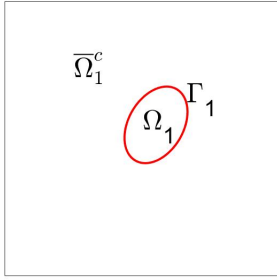
$$\mathcal{A}_i u_e \equiv \nabla \cdot (\sigma_i \nabla u_e) - \kappa_i u_e = f_i \quad \text{in } \overline{\Omega}_1^c, \quad (4.1)$$

$$\sigma_i \partial_n u_e = h_N \quad \text{on } \Gamma_1. \quad (4.2)$$

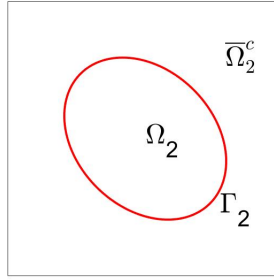
Let $\Omega_2 \subset R^2$ be a bounded domain with smooth boundary Γ_2 (Fig. 2b). The space outside Γ_2 is denoted as $\overline{\Omega}_2^c$. $u_i(\mathbf{x})$, an unknown function, is the solution to an interior Dirichlet BVP,

$$\mathcal{A}_i u_i \equiv \nabla \cdot (\sigma_i \nabla u_i) - \kappa_i u_i = 0 \quad \text{in } \Omega_2, \quad (4.3)$$

$$u_i = h_D \quad \text{on } \Gamma_2. \quad (4.4)$$



(a) Exterior Problem



(b) Interior Problem

Fig. 2: Illustration of interior and exterior boundary value problems.

Here boundaries Γ_1 , Γ_2 , PDE operator $\mathcal{A}_i$, and source term f_i are the same to Eqs. (2.1) - (2.3). However, boundary conditions h_N and h_D are different from g^D in Eq. (2.2) and g^N in Eq. (2.3). Actually, we do not need the explicit form of h_N and h_D in the integral formulations. The solution to the exterior problem is

$$u_e(\mathbf{p}) = -\mathcal{L}_i \psi_1(\mathbf{p}) + \mathcal{G}_i f_i(\mathbf{p}) \quad \text{in } \overline{\Omega}_1^c;$$

and the solution to the interior problem is

$$u_i(\mathbf{p}) = \mathcal{M}_i \varphi_2(\mathbf{p}) \quad \text{in } \Omega_2.$$

Using boundary conditions (2.2) and (2.3), we obtain the following boundary integral equations

$$\mathcal{N}_i \varphi_2 - \mathcal{M}_i^* \psi_1 - \frac{\psi_1}{2} = g^N - \mathbf{n}_p \cdot \sigma_i \nabla_p \mathcal{G}_i f_i \quad \text{on } \Gamma_1, \quad (4.5)$$

$$\mathcal{M}_i \varphi_2 + \frac{\varphi_2}{2} - \mathcal{L}_i \psi_1 = g^D - \mathcal{G}_i f_i \quad \text{on } \Gamma_2. \quad (4.6)$$

In matrix form, the exterior and interior problems are grouped into the following system

$$\begin{pmatrix} \mathcal{N}_i & -\mathcal{M}_i^* - \frac{\mathcal{I}}{2} \\ \mathcal{M}_i + \frac{\mathcal{I}}{2} & -\mathcal{L}_i \end{pmatrix} \begin{pmatrix} \varphi_2 \\ \psi_1 \end{pmatrix} = \begin{pmatrix} g^N - \mathbf{n}_p \cdot \sigma_i \nabla_p \mathcal{G}_i f_i \\ g^D - \mathcal{G}_i f_i \end{pmatrix}, \quad (4.7)$$

where $\mathcal{I}$ is the identity operator. When PDE operator is Laplace operator, i.e. $\mathcal{A}_i = \Delta$, and f_i is equal to zero, we need one more condition in the system, $\int_{\Gamma_1} \psi_1(\mathbf{p}) d\mathbf{s}_p = 0$, and one more term $c \ln |\mathbf{p}|$ in the solution, see [36] for details. After discretization of density functions and boundary conditions, we solve the system by a Krylov subspace method, the generalized minimum residual method (GMRES) [34]. The final solution to the Eqs. (2.1) - (2.3) is given by

$$u(\mathbf{p}) = \mathcal{M}_i \varphi_2(\mathbf{p}) - \mathcal{L}_i \psi_1(\mathbf{p}) + \mathcal{G}_i f_i(\mathbf{p}) \quad \text{in } \overline{\Omega}_1^c \cap \Omega_2.$$

4.2 Interface problem

Let $\mathcal{B} \subset \mathbb{R}^2$ be a rectangle box. $\Omega_1 \subset \mathbb{R}^2$ is a bounded domain with smooth boundary Γ_1 (see Fig. 3a). The domain outside Γ_1 is denoted as $\overline{\Omega}_1^c$. $u_1(\mathbf{x})$ and $u_2(\mathbf{x})$ are unknown functions, as the solutions to an interface problem

$$\mathcal{A}_1 u_1 \equiv \nabla \cdot (\sigma_1(\mathbf{p}) \nabla u_1) - \kappa_1(\mathbf{p}) u_1 = f_1(\mathbf{p}) \quad \text{in } \Omega_1, \quad (4.8)$$

$$\mathcal{A}_2 u_2 \equiv \nabla \cdot (\sigma_2(\mathbf{p}) \nabla u_2) - \kappa_2(\mathbf{p}) u_2 = 0 \quad \text{in } \overline{\Omega}_1^c, \quad (4.9)$$

$$u_1 - u_2 = g'_1 \quad \text{and} \quad \sigma_1 \partial_n u_1 - \sigma_2 \partial_n u_2 = j'_1 \quad \text{on } \Gamma_1, \quad (4.10)$$

$$u_2 = 0 \quad \text{on } \partial \mathcal{B}. \quad (4.11)$$

Let $\mathcal{B} \subset \mathbb{R}^2$ be a rectangle box. $\Omega'_2 \subset \mathbb{R}^2$ is a bounded domain with smooth boundary Γ_2 (see Fig. 3b). The domain outside Γ_2 is denoted as Ω_3 . $u'_2(\mathbf{x})$ and $u_3(\mathbf{x})$ are unknown functions, as the solutions to another interface problem

$$\mathcal{A}_2 u'_2 \equiv \nabla \cdot (\sigma_2(\mathbf{p}) \nabla u'_2) - \kappa_2(\mathbf{p}) u'_2 = f_2(\mathbf{p}) \quad \text{in } \Omega'_2, \quad (4.12)$$

$$\mathcal{A}_3 u_3 \equiv \nabla \cdot (\sigma_3(\mathbf{p}) \nabla u_3) - \kappa_3(\mathbf{p}) u_3 = f_3(\mathbf{p}) \quad \text{in } \Omega_3, \quad (4.13)$$

$$u'_2 - u_3 = g'_2 \quad \text{and} \quad \sigma_2 \partial_n u'_2 - \sigma_3 \partial_n u_3 = j'_2 \quad \text{on } \Gamma_2, \quad (4.14)$$

$$u_3 = 0 \quad \text{on } \partial \mathcal{B}, \quad (4.15)$$

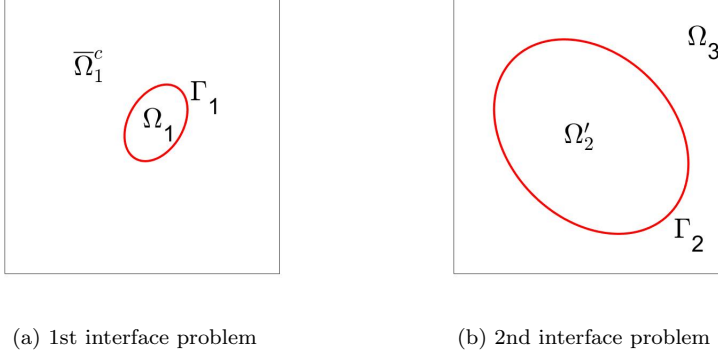


Fig. 3: Illustration of the two interface problems.

Here both interfaces Γ_1, Γ_2 , PDE operators $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$, and source term f_1, f_2, f_3 are the same to the interface problem Eqs. (2.4)-(2.9). $\bar{\Omega}_1^c = \Omega_2 \cup \Omega_3 \cup \Gamma_2$. $\Omega'_2 = \Omega_1 \cup \Omega_2 \cup \Gamma_1$. But interface conditions g'_1, j'_1, g'_2 and j'_2 are different from the interface conditions in Eqs. (2.7) and (2.8), and are not necessarily required for the KFBIM. The solution to the 1st interface problem is

$$\begin{aligned} u_1(\mathbf{p}) &= \mathcal{M}_1 \varphi_1(\mathbf{p}) - \mathcal{L}_1 \psi_1(\mathbf{p}) + \mathcal{G}_1 f_1 & \text{in } \Omega_1, \\ u_2(\mathbf{p}) &= \mathcal{M}_2 \varphi_1(\mathbf{p}) - \mathcal{L}_2 \psi_1(\mathbf{p}) & \text{in } \bar{\Omega}_1^c; \end{aligned}$$

and the solution to the 2nd interface problem is

$$\begin{aligned} u'_2(\mathbf{p}) &= \mathcal{M}_2 \varphi_2(\mathbf{p}) - \mathcal{L}_2 \psi_2(\mathbf{p}) + \mathcal{G}_2 f_2 & \text{in } \Omega'_2, \\ u_3(\mathbf{p}) &= \mathcal{M}_3 \varphi_2(\mathbf{p}) - \mathcal{L}_3 \psi_2(\mathbf{p}) + \mathcal{G}_3 f_3 & \text{in } \Omega_3. \end{aligned}$$

We have four boundary integral equations on Γ_1 and Γ_2 by interface conditions Eqs. (2.7) and (2.8),

$$\begin{aligned} (\mathcal{M}_1 - \mathcal{M}_2 + \mathcal{I})\varphi_1 + (\mathcal{L}_2 - \mathcal{L}_1)\psi_1 - \mathcal{M}_2\varphi_2 + \mathcal{L}_2\psi_2 \\ = g_1 - \mathcal{G}_1 f_1 + \mathcal{G}_2 f_2 \end{aligned} \quad \text{on } \Gamma_1, \quad (4.16)$$

$$\begin{aligned} (\mathcal{N}_1 - \mathcal{N}_2)\varphi_1 + (\mathcal{M}_2^* - \mathcal{M}_1^* + \mathcal{I})\psi_1 + \mathcal{M}_2^*\psi_2 - \mathcal{N}_2\varphi_2 \\ = j_1 + \mathbf{n}_p \cdot \sigma_2 \nabla_p \mathcal{G}_2 f_2 - \mathbf{n}_p \cdot \sigma_1 \nabla_p \mathcal{G}_1 f_1 \end{aligned} \quad \text{on } \Gamma_1, \quad (4.17)$$

$$\begin{aligned} \mathcal{M}_2\varphi_1 - \mathcal{L}_2\psi_1 + (\mathcal{M}_2 - \mathcal{M}_3 + \mathcal{I})\varphi_2 + (\mathcal{L}_3 - \mathcal{L}_2)\psi_2 \\ = g_2 - \mathcal{G}_2 f_2 + \mathcal{G}_3 f_3 \end{aligned} \quad \text{on } \Gamma_2, \quad (4.18)$$

$$\begin{aligned} \mathcal{N}_2\varphi_1 - \mathcal{M}_2^*\psi_1 + (\mathcal{N}_2 - \mathcal{N}_3)\varphi_2 + (\mathcal{M}_3^* - \mathcal{M}_2^* + \mathcal{I})\psi_2 \\ = j_2 + \mathbf{n}_p \cdot \sigma_3 \nabla_p \mathcal{G}_3 f_3 - \mathbf{n}_p \cdot \sigma_2 \nabla_p \mathcal{G}_2 f_2 \end{aligned} \quad \text{on } \Gamma_2. \quad (4.19)$$

In matrix form, the above integral equations can be written as

$$\begin{pmatrix} \mathcal{M}_1 - \mathcal{M}_2 + \mathcal{I} & \mathcal{L}_2 - \mathcal{L}_1 & -\mathcal{M}_2 & \mathcal{L}_2 \\ \mathcal{N}_1 - \mathcal{N}_2 & \mathcal{M}_2^* - \mathcal{M}_1^* + \mathcal{I} & -\mathcal{N}_2 & \mathcal{M}_2^* \\ \mathcal{M}_2 & -\mathcal{L}_2 & \mathcal{M}_2 - \mathcal{M}_3 + \mathcal{I} & \mathcal{L}_3 - \mathcal{L}_2 \\ \mathcal{N}_2 & -\mathcal{M}_2^* & \mathcal{N}_2 - \mathcal{N}_3 & \mathcal{M}_3^* - \mathcal{M}_2^* + \mathcal{I} \end{pmatrix} \begin{pmatrix} \varphi_1 \\ \psi_1 \\ \varphi_2 \\ \psi_2 \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix}$$

where

$$\begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix} = \begin{pmatrix} g_1 - \mathcal{G}_1 f_1 + \mathcal{G}_2 f_2 \\ j_1 - \sigma_1 \partial_{\mathbf{n}}(\mathcal{G}_1 f_1) + \sigma_2 \partial_{\mathbf{n}}(\mathcal{G}_2 f_2) \\ g_2 - \mathcal{G}_2 f_2 + \mathcal{G}_3 f_3 \\ j_2 - \sigma_2 \partial_{\mathbf{n}}(\mathcal{G}_2 f_2) + \sigma_3 \partial_{\mathbf{n}}(\mathcal{G}_3 f_3) \end{pmatrix}. \quad (4.20)$$

Here notation $\partial_{\mathbf{n}} = \mathbf{n}_{\mathbf{p}} \cdot \nabla_{\mathbf{p}}$. After solving the above linear system by GMRES, the final solution u to the interface problem is given by

$$\begin{aligned} u(\mathbf{p}) &= u_1(\mathbf{p}) && \text{in } \Omega_1, \\ u(\mathbf{p}) &= u_2(\mathbf{p}) + u'_2(\mathbf{p}) && \text{in } \Omega_2, \\ u(\mathbf{p}) &= u_3(\mathbf{p}) && \text{in } \Omega_3. \end{aligned}$$

5 Evaluation of boundary or volume integral

In general, the Green's function of boundary integrals is not available for elliptic PDEs with variable coefficients. The KFBIM circumvents the direct evaluation of boundary integrals, and compute boundary or volume integrals value as an interpolation of Cartesian grid value from a discretized equivalent interface problem. For completeness, we give technical details in this section, following [17, 18, 21].

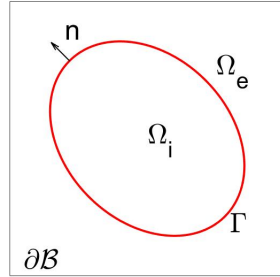


Fig. 4: Irregular Domain with exterior domain Ω_e and interior domain Ω_i .

5.1 Integrals reinterpretation

In a rectangle box $\mathcal{B}$, an irregular interface Γ separate it into two domains Ω_i and Ω_e (See Fig. 4 for an illustration). A piecewise smooth function $w(\mathbf{p})$ defined in $\mathcal{B}$ has jumps across Γ . We denote $w^+(\mathbf{p})$ and $w^-(\mathbf{p})$ respectively as

the restriction of $w(\mathbf{p})$ in Ω_i and Ω_e . When $\mathbf{p} \in \Gamma$, $w^+(\mathbf{p})$ and $w^-(\mathbf{p})$ means limits from corresponding sides. We define jumps of $w(\mathbf{p})$ across Γ as follow

$$\begin{aligned} [w(\mathbf{p})] &= w^+(\mathbf{p}) - w^-(\mathbf{p}) && \text{on } \Gamma, \\ [\partial_{\mathbf{n}} w(\mathbf{p})] &= \partial_{\mathbf{n}} w^+(\mathbf{p}) - \partial_{\mathbf{n}} w^-(\mathbf{p}) && \text{on } \Gamma. \end{aligned}$$

Using the jumps defined above, the double layer boundary integral $v(\mathbf{p}) = \mathcal{M}_i \varphi(\mathbf{p})$ satisfies an interface problem

$$\begin{aligned} \mathcal{A}_i v &\equiv \nabla \cdot (\sigma_i(\mathbf{p}) \nabla v) - \kappa_i(\mathbf{p}) v = 0 && \text{in } \mathcal{B} \setminus \Gamma, \\ [v] &= \varphi && \text{on } \Gamma, \\ \sigma_i [\partial_{\mathbf{n}} v] &= 0 && \text{on } \Gamma, \\ v &= 0 && \text{on } \partial \mathcal{B}; \end{aligned}$$

the single layer boundary integral $v(\mathbf{p}) = -\mathcal{L}_i \psi(\mathbf{p})$ satisfies an interface problem

$$\begin{aligned} \mathcal{A}_i v &\equiv \nabla \cdot (\sigma_i(\mathbf{p}) \nabla v) - \kappa_i(\mathbf{p}) v = 0 && \text{in } \mathcal{B} \setminus \Gamma, \\ [v] &= 0 && \text{on } \Gamma, \\ \sigma_i [\partial_{\mathbf{n}} v] &= \psi && \text{on } \Gamma, \\ v &= 0 && \text{on } \partial \mathcal{B}. \end{aligned}$$

Finally, the volume integral $v(\mathbf{p}) = \mathcal{G}_i f_i(\mathbf{p})$ satisfies the following interface problem

$$\begin{aligned} \mathcal{A}_i v &= f_i && \text{in } \Omega_i, \\ \mathcal{A}_i v &= 0 && \text{in } \Omega_e, \\ [v] &= 0 && \text{on } \Gamma, \\ \sigma_i [\partial_{\mathbf{n}} v] &= 0 && \text{on } \Gamma, \\ v &= 0 && \text{on } \partial \mathcal{B}. \end{aligned}$$

Here boundary or volume integrals are defined the same as those used in section 3. The equivalence between the interface problem and boundary or volume integrals can be found in [21].

5.2 PDE discretization

Let box $\mathcal{B} = [a, b] \times [c, d]$. N is mesh size on each axis direction and $h_x = \frac{b-a}{N}$, $h_y = \frac{c-d}{N}$. We assume $h_x = h_y = h$ for simplicity, $x_i = a + ih$, $y_j = c + jh$, $\mathbf{p}_{i,j} = (x_i, y_j)$, $i, j = 0 \dots N$. We discretize PDE

$$\mathcal{A} v \equiv \nabla \cdot (\sigma(\mathbf{p}) \nabla v) - \kappa(\mathbf{p}) v = f \quad \text{in } \mathcal{B} \setminus \Gamma$$

on the Cartesian grid with a modified finite difference scheme

$$\mathcal{A}_h v_{i,j} \equiv \frac{s_{i,j} - 4\bar{\sigma}_{i,j}v_{i,j}}{h^2} - \kappa_{i,j}v_{i,j} = f_{i,j}, \quad (5.1)$$

$$s_{i,j} = \sigma_{i+\frac{1}{2},j}v_{i+1,j} + \sigma_{i-\frac{1}{2},j}v_{i-1,j} + \sigma_{i,j+\frac{1}{2}}v_{i,j+1} + \sigma_{i,j-\frac{1}{2}}v_{i,j-1}, \quad (5.2)$$

$$\bar{\sigma}_{i,j} = \frac{\sigma_{i+\frac{1}{2},j} + \sigma_{i-\frac{1}{2},j} + \sigma_{i,j+\frac{1}{2}} + \sigma_{i,j-\frac{1}{2}}}{4}, \quad (5.3)$$

where $v_{i,j}$ is a finite difference approximation of $v(p_{i,j})$, $\sigma_{i+\frac{1}{2},j} = \sigma(x_i + \frac{h}{2}, y_j)$, $\sigma_{i-\frac{1}{2},j} = \sigma(x_i - \frac{h}{2}, y_j)$, $\sigma_{i,j+\frac{1}{2}} = \sigma(x_i, y_j + \frac{h}{2})$, $\sigma_{i,j-\frac{1}{2}} = \sigma(x_i, y_j - \frac{h}{2})$, $\kappa_{i,j} = \kappa(x_i, y_j)$, and $f_{i,j} = f(x_i, y_j)$. The finite difference scheme here is based on a five point stencil. The equation (5.1) has a second order accuracy if without discontinuities across Γ .

5.3 Correction in discrete system

If a node $\mathbf{p}_{i,j}$ and any of its neighbors $\mathbf{p}_{i+1,j}$, $\mathbf{p}_{i-1,j}$, $\mathbf{p}_{i,j+1}$, $\mathbf{p}_{i,j-1}$ are on two different sides of Γ , we call $\mathbf{p}_{i,j}$ as an irregular node. Otherwise $\mathbf{p}_{i,j}$ is a regular node. The truncation error of the finite difference method is large at those irregular nodes,

$$\mathcal{A}_h v_{i,j} - f_{i,j} = \begin{cases} O(h^2) & \text{if } (x_i, y_j) \text{ is a regular point,} \\ O(h^{-2}) & \text{if } (x_i, y_j) \text{ is an irregular point.} \end{cases}$$

We need to correct the scheme at irregular points to keep the second order accuracy on the whole mesh. We demonstrate the correction procedure using a simple case shown in Fig. 5 where Γ intersects a horizontal grid line at (z_i, y_j) , $x_i \leq z_i < x_{i+1}$.

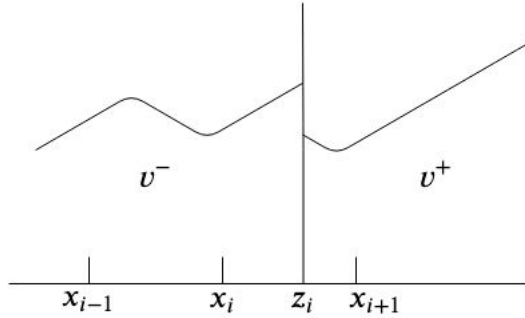


Fig. 5: Irregular Node

The local truncation error at the i -th node is

$$E_{h,x}(x_i, y_j) \equiv \frac{\sigma_{i+\frac{1}{2},j}(v^+(x_{i+1}, y_j) - v^-(x_i, y_j))}{h^2} - \frac{\sigma_{i-\frac{1}{2},j}(v^-(x_i, y_j) - v^-(x_{i-1}, y_j))}{h^2} - \frac{\partial}{\partial x}(\sigma \frac{\partial}{\partial x} v^-(x_i, y_j)). \quad (5.4)$$

We next expand $v^\pm(x_{i+1}, y_j)$ at (z_i, y_j) with a Taylor expansion at the z_i point,

$$v^\pm(x_{i+1}, y_j) = v^\pm(z_i, y_j) + \partial_x v^\pm(z_i, y_j)(x_{i+1} - z_i) + \frac{1}{2} \partial_{xx} v^\pm(z_i, y_j)(x_{i+1} - z_i)^2 + \frac{1}{6} \partial_{xxx} v^\pm(z_i, y_j)(x_{i+1} - z_i)^3 + O(h^4). \quad (5.5)$$

The truncation error at (x_i, y_j) is

$$\begin{aligned} E_{h,x}(x_i, y_j) &\equiv \frac{\sigma_{i+\frac{1}{2},j}(v^+(x_{i+1}, y_j) - v^-(x_{i+1}, y_j))}{h^2} \\ &= \frac{\sigma_{i+\frac{1}{2},j}}{h^2} \left\{ [v] + [v_x](x_{i+1} - z_i) + \frac{1}{2} [v_{xx}](x_{i+1} - z_i)^2 + \frac{1}{6} [v_{xxx}](x_{i+1} - z_i)^3 \right\} + O(h^2), \end{aligned} \quad (5.6)$$

where $[v] = v^+(z_i, y_j) - v^-(z_i, y_j)$, $[v_x] = \partial_x v^+(z_i, y_j) - \partial_x v^-(z_i, y_j)$, $[v_{xx}] = \partial_{xx} v^+(z_i, y_j) - \partial_{xx} v^-(z_i, y_j)$, and $[v_{xxx}] = \partial_{xxx} v^+(z_i, y_j) - \partial_{xxx} v^-(z_i, y_j)$ are jumps of v value and its partial derivatives across the interface Γ . Next we add a correction term

$$C_{h,x}^+(x_i, y_j) = \frac{\sigma_{i+\frac{1}{2},j}}{h^2} \left\{ [v] + [v_x](x_{i+1} - z_i) + \frac{1}{2} [v_{xx}](x_{i+1} - z_i)^2 \right\} \quad (5.7)$$

on the right hand side of the finite difference equation (5.1). The computation of jumps $[v]$, $[v_x]$, $[v_{xx}]$ are given in section 5.6. Similarly if a horizontal grid line intersects Γ between x_{i-1} and x_i , We also need add a correction term

$$C_{h,x}^-(x_i, y_j) = -\frac{\sigma_{i-\frac{1}{2},j}}{h^2} \left\{ [v] + [v_x](x_{i-1} - z_i) + \frac{1}{2} [v_{xx}](x_{i-1} - z_i)^2 \right\}. \quad (5.8)$$

Γ may intersect grid lines horizontally and vertically multiple times. For every intersection point, we add necessary correction terms among $C_{h,x}^+$, $C_{h,x}^-$, $C_{h,y}^+$, $C_{h,y}^-$ on the right hand side of (5.1). We denote $C_h(x_i, y_j)$ as a summation of all corrections at (x_i, y_j) and summarize a modified finite difference equation

$$\mathcal{A}_h v_{i,j} = \begin{cases} f_{i,j} & \text{if } (x_i, y_j) \text{ is a regular point,} \\ f_{i,j} + C_h(x_i, y_j) & \text{if } (x_i, y_j) \text{ is an irregular point.} \end{cases}$$

The finite difference method here is of second order accurate except at irregular points, where it has first order accuracy [17, 37].

5.4 Solution of the discrete finite difference equations

The coefficients matrix created by the finite difference method is a symmetric positive definite matrix because of its diagonal dominance. For a constant coefficient PDE, i.e. Laplace or modified Helmholtz operator, a fast Fourier transform based elliptic PDE solver can be used to solve (5.1). We implement a geometric multigrid preconditioned conjugate gradient iterative method (GMG-PCG) for a variable coefficients PDE [21]. The whole process is a full multigrid process where a preconditioned conjugate gradient iterative method is coupled with every single V-cycle, while the preconditioning is a single multigrid V cycle process. The pre-smoothing and post-smoothing are one forward and one backward Gauss-Seidel iteration. The prolongation is calculated by a bilinear interpolation and the restriction is the adjoint of the prolongation. The coarsest grid has only one point inside box $\mathcal{B}$.

5.5 Interpolation of integrals on the interface

Assume v_h is a piecewise smooth function which satisfies PDE operator $\mathcal{A}$. It has discontinuities of function value and partial derivatives across Γ . Let $v_h(\mathbf{p}_{i,j}) = v_{i,j}$ where $v_{i,j}$ is the numerical solution given by the finite difference method. v_h^+ and v_h^- are restrictions of v_h in Ω_i and Ω_e . We expand $v_h(\mathbf{p})$ on one point $\mathbf{q} \in \Gamma$ (See Fig. 6),

$$\begin{aligned} v_h(\mathbf{p}) = & v_h^+(\mathbf{q}) + \frac{\partial v_h^+(\mathbf{q})}{\partial x} \xi + \frac{\partial v_h^+(\mathbf{q})}{\partial y} \eta + \frac{1}{2} \frac{\partial^2 v_h^+(\mathbf{q})}{\partial x^2} \xi^2 \\ & + \frac{1}{2} \frac{\partial^2 v_h^+(\mathbf{q})}{\partial x \partial y} \xi \eta + \frac{1}{2} \frac{\partial^2 v_h^+(\mathbf{q})}{\partial y^2} \eta^2 + O(|\mathbf{p} - \mathbf{q}|^3) \quad \text{if } \mathbf{p} \in \Omega_i, \end{aligned} \quad (5.9)$$

and

$$\begin{aligned} v_h(\mathbf{p}) = & v_h^-(\mathbf{q}) + \frac{\partial v_h^-(\mathbf{q})}{\partial x} \xi + \frac{\partial v_h^-(\mathbf{q})}{\partial y} \eta + \frac{1}{2} \frac{\partial^2 v_h^-(\mathbf{q})}{\partial x^2} \xi^2 \\ & + \frac{1}{2} \frac{\partial^2 v_h^-(\mathbf{q})}{\partial x \partial y} \xi \eta + \frac{1}{2} \frac{\partial^2 v_h^-(\mathbf{q})}{\partial y^2} \eta^2 + O(|\mathbf{p} - \mathbf{q}|^3) \quad \text{if } \mathbf{p} \in \Omega_e. \end{aligned} \quad (5.10)$$

Here $(\xi, \eta) = \mathbf{p} - \mathbf{q}$. The way to choose six grid nodes around $\mathbf{q}$ is as follow. Firstly, we pick the closest mesh point $\mathbf{p}_0$ to $\mathbf{q}$. Then 4 neighbors of $\mathbf{p}_0$ are also chosen. $\mathbf{p}_5$ is the last one to construct a small rectangle which contains $\mathbf{q}$.

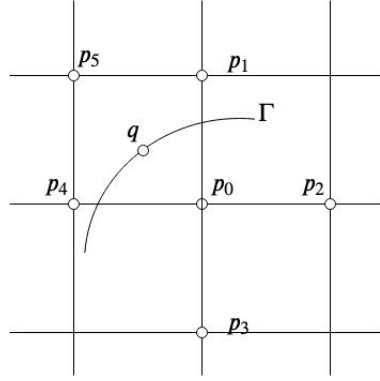


Fig. 6: Interpolation Points Selection

We denote the restriction of v_h and its partial derivative as

$$V^\pm \equiv v_h^\pm(\mathbf{q}), \quad V_x^\pm \equiv \frac{\partial v_h^\pm(\mathbf{q})}{\partial x}, \quad V_y^\pm \equiv \frac{\partial v_h^\pm(\mathbf{q})}{\partial y}, \quad (5.11)$$

$$V_{xx}^\pm \equiv \frac{\partial^2 v_h^\pm(\mathbf{q})}{\partial x^2}, \quad V_{xy}^\pm \equiv \frac{\partial^2 v_h^\pm(\mathbf{q})}{\partial x \partial y}, \quad V_{yy}^\pm \equiv \frac{\partial^2 v_h^\pm(\mathbf{q})}{\partial y^2}. \quad (5.12)$$

They are limit values from the corresponding side, i.e. $V^+ = \lim_{\mathbf{p}' \rightarrow \mathbf{q}} v_h(\mathbf{p}')$, $\mathbf{p}' \in \Omega_i$. Equations (5.9) and (5.10) are represented as

$$V^+ + V_x^+ \xi_j + V_y^+ \eta_j + \frac{1}{2} \xi_j^2 V_{xx}^+ + V_{xy}^+ \xi_j \eta_j + \frac{1}{2} V_{yy}^+ \eta_j^2 = V_j \quad \text{if } \mathbf{p}_j \in \Omega_i, \quad (5.13)$$

$$V^- + V_x^- \xi_j + V_y^- \eta_j + \frac{1}{2} \xi_j^2 V_{xx}^- + V_{xy}^- \xi_j \eta_j + \frac{1}{2} V_{yy}^- \eta_j^2 = V_j \quad \text{if } \mathbf{p}_j \in \Omega_e, \quad (5.14)$$

for $j = 0, 1, \dots, 5$. Here $(\xi_j, \eta_j) = \mathbf{p}_j - \mathbf{q}$ and $V_j \equiv v_h(\mathbf{p}_j)$. We denote

$$J_j = [V] + [V_x] \xi_j + [V_y] \eta_j + \frac{1}{2} [V_{xx}] \xi_j^2 + [V_{xy}] \xi_j \eta_j + \frac{1}{2} [V_{yy}] \eta_j^2 \quad (5.15)$$

and reorganize (5.14) into

$$V^+ + V_x^+ \xi_j + V_y^+ \eta_j + \frac{1}{2} \xi_j^2 V_{xx}^+ + V_{xy}^+ \xi_j \eta_j + \frac{1}{2} V_{yy}^+ \eta_j^2 = V_j + J_j \quad \text{if } \mathbf{p}_j \in \Omega_e. \quad (5.16)$$

We have six unknowns V^+ , V_x^+ , V_y^+ , V_{xx}^+ , V_{xy}^+ and V_{yy}^+ for each boundary node $\mathbf{q}$. Also there are six equations from (5.13) and (5.16). After jumps are computed (details given in the next subsection), it is straightforward to solve a 6×6 linear system.

5.6 Compute jumps of partial derivatives

Assume $v(\mathbf{p})$ satisfy the interface problem

$$\mathcal{A}v \equiv \nabla \cdot (\sigma \nabla v) - \kappa v = f \quad \text{in } \mathcal{B} \setminus \Gamma, \quad (5.17)$$

$$[v] = \varphi \quad \text{on } \Gamma, \quad (5.18)$$

$$\sigma[\partial_{\mathbf{n}}v] = \psi \quad \text{on } \Gamma. \quad (5.19)$$

Denote $\boldsymbol{\tau}$ as a tangent vector of a point on Γ . Let the parametric description of Γ as follows,

$$x = x(\theta) \quad \text{and} \quad y = y(\theta),$$

and

$$\boldsymbol{\tau} = (x_\theta, y_\theta) \quad \text{and} \quad \mathbf{n} = \left(\frac{-y_\theta}{\sqrt{x_\theta^2 + y_\theta^2}}, \frac{x_\theta}{\sqrt{x_\theta^2 + y_\theta^2}} \right),$$

where $x_\theta = \frac{\partial x}{\partial \theta}$ and $y_\theta = \frac{\partial y}{\partial \theta}$. We denote $\partial_{\boldsymbol{\tau}} = \tau_1 \frac{\partial}{\partial x} + \tau_2 \frac{\partial}{\partial y}$ and $\partial_{\mathbf{n}} = n_1 \frac{\partial}{\partial x} + n_2 \frac{\partial}{\partial y}$. Taking tangential derivative of Eq. (5.18) gives

$$\partial_{\boldsymbol{\tau}}[v] = \partial_{\boldsymbol{\tau}}\varphi \quad \text{on } \Gamma, \quad (5.20)$$

and expand the right hand side,

$$\partial_{\boldsymbol{\tau}}\varphi = x_\theta\varphi_x + y_\theta\varphi_y = \varphi_\theta.$$

Equations (5.19) and (5.20) are equivalent to

$$n_1[v_x] + n_2[v_y] = \frac{\psi}{\sigma}, \quad (5.21)$$

$$x_\theta[v_x] + y_\theta[v_y] = \varphi_\theta. \quad (5.22)$$

After solving those two equations, we have $[v_x]$ and $[v_y]$, and plug jumps into Eq. (5.17),

$$\nabla \cdot (\sigma \nabla[v]) - \kappa[v] = [f] \quad \text{on } \Gamma, \quad (5.23)$$

and expand it,

$$\sigma_x[v_x] + \sigma[v_{xx}] + \sigma_y[v_y] + \sigma[v_{yy}] = [f] + \kappa[v].$$

Take double tangential derivative of Eq. (5.18),

$$\partial_{\boldsymbol{\tau}\boldsymbol{\tau}}[v] = \partial_{\boldsymbol{\tau}\boldsymbol{\tau}}\varphi \quad \text{on } \Gamma, \quad (5.24)$$

and we have the following equation by taking tangential derivative of Eq. (5.22),

$$\begin{aligned} & x_\theta^2[v_{xx}] + y_\theta^2[v_{yy}] + 2x_\theta y_\theta[v_{xy}] \\ & + (x_\theta \frac{\partial x_\theta}{\partial x} + y_\theta \frac{\partial x_\theta}{\partial y})[v_x] + (x_\theta \frac{\partial y_\theta}{\partial x} + y_\theta \frac{\partial y_\theta}{\partial y})[v_y] = \varphi_{\theta\theta}. \end{aligned}$$

Here $\varphi_{\theta\theta} = \frac{\partial^2 \varphi}{\partial \theta^2}$, $x_{\theta\theta} = \frac{\partial^2 x}{\partial \theta^2}$, $y_{\theta\theta} = \frac{\partial^2 y}{\partial \theta^2}$. Take tangential derivative of Eq. (5.19),

$$\partial_{\tau}\{\sigma[\partial_{\mathbf{n}}v]\} = \partial_{\tau}\psi \quad \text{on } \Gamma, \quad (5.25)$$

and expand it,

$$\begin{aligned} & n_1[v_x](\sigma_x x_{\theta} + \sigma_y y_{\theta}) + n_2[v_y](\sigma_x x_{\theta} + \sigma_y y_{\theta}) \\ & + \sigma[v_x](x_{\theta} \frac{\partial n_1}{\partial x} + y_{\theta} \frac{\partial n_1}{\partial y}) + \sigma[v_y](x_{\theta} \frac{\partial n_2}{\partial x} + y_{\theta} \frac{\partial n_2}{\partial y}) \\ & + \sigma n_1 x_{\theta}[v_{xx}] + \sigma n_2 y_{\theta}[v_{yy}] + \sigma[v_{xy}](n_1 y_{\theta} + n_2 x_{\theta}) = \psi_{\theta} \end{aligned}$$

Equations (5.23)-(5.25) are equivalent to

$$[v_{xx}] + [v_{yy}] = r_1 \quad (5.26)$$

$$x_{\theta}^2[v_{xx}] + y_{\theta}^2[v_{yy}] + 2x_{\theta}y_{\theta}[v_{xy}] = r_2, \quad (5.27)$$

$$n_1 x_{\theta}[v_{xx}] + n_2 y_{\theta}[v_{yy}] + (n_1 y_{\theta} + n_2 x_{\theta})[v_{xy}] = r_3, \quad (5.28)$$

where the right hand side

$$r_1 = \{\kappa[v] - \sigma_x[v_x] - \sigma_y[v_y] + [f]\}/\sigma, \quad (5.29)$$

$$r_2 = \varphi_{\theta\theta} - x_{\theta\theta}[v_x] - y_{\theta\theta}[v_y], \quad (5.30)$$

$$r_3 = \frac{\psi_{\theta}}{\sigma} - \frac{\sigma_{\theta}}{\sigma} \{n_1[v_x] + n_2[v_y]\} - \frac{\partial n_1}{\partial \theta}[v_x] - \frac{\partial n_2}{\partial \theta}[v_y]. \quad (5.31)$$

We obtain $[v_{xx}]$, $[v_{xy}]$ and $[v_{yy}]$ after solving three equations.

In summary, we have the following algorithm for solving the boundary value problem (2.1)-(2.3). The algorithm is similar for solving the interface problem (2.4)-(2.9). The overall algorithm is given below.

Data: Discrete boundary conditions

Result: Evaluate unknown u in doubly-connected domain

Initialization of mesh grid;

Compute volume integral $\mathcal{G}_i f_i$;

GMRES iteration is as follows:

while ψ_1 and φ_2 do not meet the tolerance **do**

 Given ψ_1 and φ_2 , compute jumps on intersection points;

 Correction finite difference equations;

 Solve equations with GMG-PCG;

 Compute jumps on boundary nodes;

 Interpolate boundary integrals on Γ_1 and Γ_2 ;

if Interpolation is close enough to boundary conditions **then**

 Quit the loop;

else

 Get ψ_1 and φ_2 for next step;

end

end

6 Numerical Examples

In this section, we provide a few preliminary numerical examples to illustrate the accuracy and capability of the method. In example 1, we consider a variable coefficient boundary value problem. In example 2, we consider a variable coefficient two-interface problem. We would also like to illustrate the good behavior of the method by considering example 3, a two-interface problem whose diffusion coefficients differs significantly between computational domains as shown in Fig. 1. In example 4, we consider an interface problem with closely-spaced interfaces. Unless specified, we use uniform grid on the computational domain $\mathcal{B} = [-1, 1] \times [-1, 1]$.

In all calculations, boundaries and interfaces are explicitly represented with parametric curve. The discretized boundary integral equations are solved iteratively by GMRES. In boundary value problem (2.1)-(2.3), initial value of density functions φ_2 and ψ_1 are setup the same as the right hand side of Eq. (4.7). In interface problem (2.4)-(2.9), initial value of density functions φ_1 , ψ_1 , φ_2 and ψ_2 are setup the same as the right hand side of Eq. (4.20), respectively. The GMRES iteration of the integral equation system stops when l^2 -norm of the residual relative to l^2 -norm of initial residual is less than the tolerance $e_{gmres} = 1\text{E-}10$.

In examples 1, 2 and 4, the modified finite difference scheme for evaluating integrals is solved by a geometric multigrid preconditioned conjugate gradient iteration method (GMG-PCG) [21]. In example 3, we use a fast Fourier transform based Poisson solver to gain more efficiency [21]. GMG-PCG stops when l^2 -norm of residual relative to l^2 -norm of initial residual is less than $e_{mg} = 1\text{E-}10$. All numerical experiments are performed on a desktop with Intel(R) Core i7-7700K 4.2GHz CPU, 16 GB 2133 MHz DDR3 ram.

Numerical results are listed in Table 1. The first column shows the resolution of the mesh grid. The second column shows the iteration number of GMRES. The third column indicates the l^2 -norm of all errors on mesh in the domain, $\|e_h\|_{l^2} = \sqrt{\frac{\sum e_i^2}{N}}$, where N is the total number of mesh grids in the domain. The fourth column indicates the infinity norm of all errors, $\|e_h\|_{\infty} = \max |e_i|$. Error e_i measures the difference between exact value given by the preset function and numerical value at the i -th mesh node. The last column shows the CPU time for each resolution.

Example 1 (Boundary value problem). The test function is $u(x, y) = e^{-x} \cos(y) + e^{-y} \sin(x)$ with coefficients $\sigma(x, y) = 1.5 + 0.5(\sin(x) + \cos(y))$, $\kappa(x, y) = e^{0.6x+0.8y}$, the source term $f(x, y) = \frac{\sin^2(y) - \cos(x) \cos(y)}{2} e^{-x} + \frac{\cos^2(x) + \sin(x) \sin(y)}{2} e^{-y} - e^{-0.4x+0.8y} \cos(y) - e^{0.6x-0.2y} \sin(x)$, and boundary conditions $g^N(x, y) = \left(1.5 + 0.5(\sin(x) + \cos(y))\right) \partial_{\mathbf{n}}(e^{-x} \cos(y) + e^{-y} \sin(x))$ on Γ_1 and $g^D(x, y) = e^{-x} \cos(y) + e^{-y} \sin(x)$ on Γ_2 . The outer boundary is a 5-fold curve centered at $(0, 0)$. The inner boundary is 3-fold curve centered at $(0.1, 0.1)$. A parametric description

of the two interfaces is given by

$$\begin{aligned} x(\theta) &= a(\alpha + \beta \cos(k\theta)) \cos(\theta), \\ y(\theta) &= b(\alpha + \beta \cos(k\theta)) \sin(\theta), \end{aligned}$$

where $\theta \in [0, 2\pi)$. For the outer boundary, we set $a = 0.7, b = 0.9, k = 5, \beta = 0.2$, and $\alpha = 0.8$. For the inner boundary, we set $a = 0.2, b = 0.2, k = 3, \beta = 0.2$, and $\alpha = 0.8$. The outer star curve is rotated counter-clockwisely by $\frac{\pi}{4}$. The inner star curve is rotated counter-clockwisely by $\frac{\pi}{3}$.

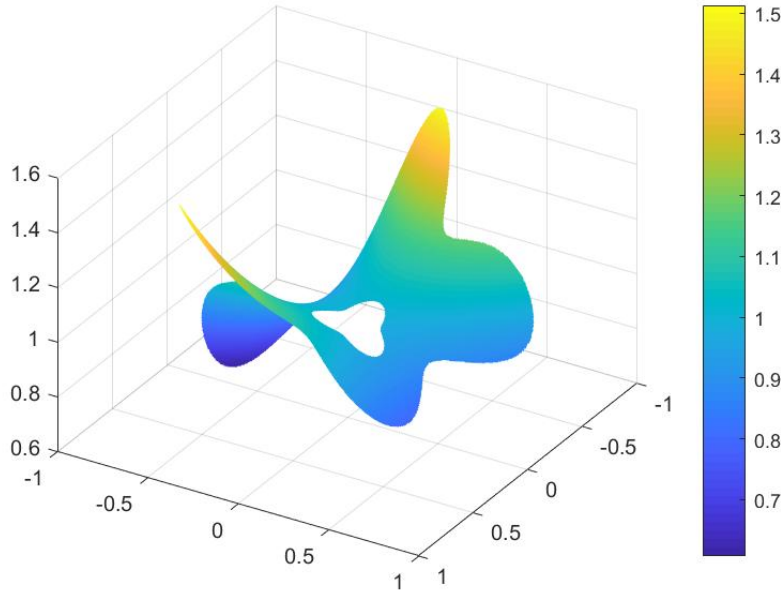


Fig. 7: Numerical solution of example 1 with 512×512 resolution.

Table 1: Numerical errors for example 1.

Resolution	#GMRES	$\ e_h\ _{l^2}$	$\ e_h\ _\infty$	Time(sec)
64×64	35	5.86E-5	5.79E-4	0.42
128×128	35	5.44E-6	1.40E-4	1.76
256×256	33	6.77E-7	2.25E-5	7.54
512×512	33	1.03E-7	3.40E-6	34.35
1024×1024	33	2.42E-8	6.12E-7	143.65

Example 2 (Interface problem). The test functions are $u_1(x, y) = e^{-x} \cos(y) + e^{-y} \sin(x)$, $u_2(x, y) = e^{0.6x+0.8y}$ and $u_3(x, y) = \sin(\frac{\pi}{2}(x+3)) \sin(\frac{\pi}{2}(y+1))$ with coefficients, source terms, and interface conditions given by

$$\begin{aligned}
\sigma_1(x, y) &= 1.5 + 0.5(\sin(x) + \cos(y)), \\
\kappa_1(x, y) &= e^{0.6x+0.8y}, \\
f_1(x, y) &= \frac{\sin^2(y) - \cos(x) \cos(y)}{2} e^{-x} + \frac{\cos^2(x) + \sin(x) \sin(y)}{2} e^{-y} \\
&\quad - e^{-0.4x+0.8y} \cos(y) - e^{0.6x-0.2y} \sin(x), \\
\sigma_2(x, y) &= 1 + x^2 + y^2, \\
\kappa_2(x, y) &= e^{\sin(\pi(x+y))+1}, \\
f_2(x, y) &= e^{0.6x+0.8y} (1 + 1.2x + 1.6y + x^2 + y^2 - e^{\sin(\pi(x+y))+1}), \\
\sigma_3(x, y) &= \cos(\pi(x+y)) + 2, \\
\kappa_3(x, y) &= x^2 + y^2, \\
f_3(x, y) &= -\frac{\pi^2}{2} \left(\sin(\pi(x+y)) \cos(\frac{\pi}{2}(x+3)) \sin(\frac{\pi}{2}(y+1)) \right. \\
&\quad \left. + \sin(\pi(x+y)) \sin(\frac{\pi}{2}(x+3)) \cos(\frac{\pi}{2}(y+1)) \right) \\
&\quad + \sin(\frac{\pi}{2}(x+3)) \sin(\frac{\pi}{2}(y+1)) \left(x^2 + y^2 - \frac{\pi^2}{2} (\cos(\pi(x+y)) + 2) \right), \\
g_1(x, y) &= e^{-x} \cos(y) + e^{-y} \sin(x) - e^{0.6x+0.8y} \quad \text{on } \Gamma_1, \\
j_1(x, y) &= \left(1.5 + 0.5(\sin(x) + \cos(y)) \right) \partial_{\mathbf{n}}(e^{-x} \cos(y) + e^{-y} \sin(x)) \\
&\quad - \left(1 + x^2 + y^2 \right) \partial_{\mathbf{n}}(e^{0.6x+0.8y}) \quad \text{on } \Gamma_1, \\
g_2(x, y) &= e^{0.6x+0.8y} - \sin(\frac{\pi}{2}(x+3)) \sin(\frac{\pi}{2}(y+1)) \quad \text{on } \Gamma_2, \\
j_2(x, y) &= \left(1 + x^2 + y^2 \right) \partial_{\mathbf{n}}(e^{0.6x+0.8y}) \\
&\quad - \left(\cos(\pi(x+y)) + 2 \right) \partial_{\mathbf{n}}(\sin(\frac{\pi}{2}(x+3)) \sin(\frac{\pi}{2}(y+1))) \quad \text{on } \Gamma_2.
\end{aligned}$$

The inner and outer interfaces are setup the same as Γ_1 and Γ_2 in example 1.

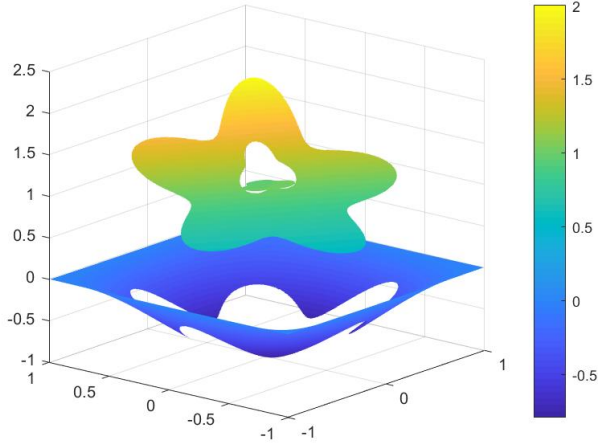
Fig. 8: Numerical solution of example 2 with 512×512 resolution

Table 2: Numerical errors for example 2.

Resolution	#GMRES	$\ e_h\ _{l^2}$	$\ e_h\ _\infty$	Time(sec)
64×64	16	1.45E-3	4.23E-3	0.46
128×128	16	3.58E-4	8.88E-4	1.95
256×256	16	9.04E-5	2.26E-4	8.75
512×512	16	2.36E-5	5.73E-5	39.46
1024×1024	16	5.63E-6	1.44E-5	174.54

Example 3 (Interface problem with widely-varied diffusion coefficients). The interface conditions on Γ_1 and Γ_2 are given such that the solution functions to the interface problem are $u_1(x, y) = e^y \cos(x) + e^x \sin(y)$, $u_2(x, y) = x^2 - y^2$, $u_3(x, y) = \frac{x}{x^2 + y^2}$. Here PDE operators $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$ in the interface problem Eqs. (2.4)-(2.9) are Laplace operator Δ . The source terms f_1 , f_2 , f_3 and coefficients κ_1 , κ_2 , κ_3 are set to be zero. We only consider constant diffusion coefficients σ_1 , σ_2 and σ_3 . The inner and outer interfaces are the same as Γ_1 and Γ_2 used in example 2.

Table 3: Numerical results of example 3 using diffusion coefficients $\sigma_1 = 1E - 6, \sigma_2 = 1, \sigma_3 = 1E6$.

Resolution	#GMRES	$\ e_h\ _{l^2}$	$\ e_h\ _{\infty}$	Time(sec)
128×128	21	3.94E-4	1.60E-3	0.29
256×256	20	7.09E-5	4.90E-4	1.09
512×512	20	1.53E-5	9.88E-5	4.43
1024×1024	20	1.05E-6	5.83E-6	18.06
2048×2048	20	1.91E-7	8.56E-7	75.79

Table 4: Numerical results of example 3 using $\sigma_1 = 50, \sigma_2 = 1, \sigma_3 = 2E - 2$.

Resolution	#GMRES	$\ e_h\ _{l^2}$	$\ e_h\ _{\infty}$	Time(sec)
128×128	23	0.12	0.17	0.32
256×256	23	7.58E-3	1.14E-2	1.25
512×512	23	5.90E-4	9.02E-4	5.01
1024×1024	23	1.59E-4	2.39E-4	20.42
2048×2048	23	1.57E-5	2.36E-5	84.92

Example 4 (Interface problem with two closely-spaced interfaces).

We consider a case with two interfaces closely spaced, as shown in [9](#). The inner interface Γ_1 is taken to have the same morphology as the outer interface Γ_2 . The maximum distance between points on Γ_1 and Γ_2 with the same angular θ is about $5h$, where h is mesh size. Test functions u_i , coefficients σ_i, κ_i , and f_i are the same as those used in example 2.

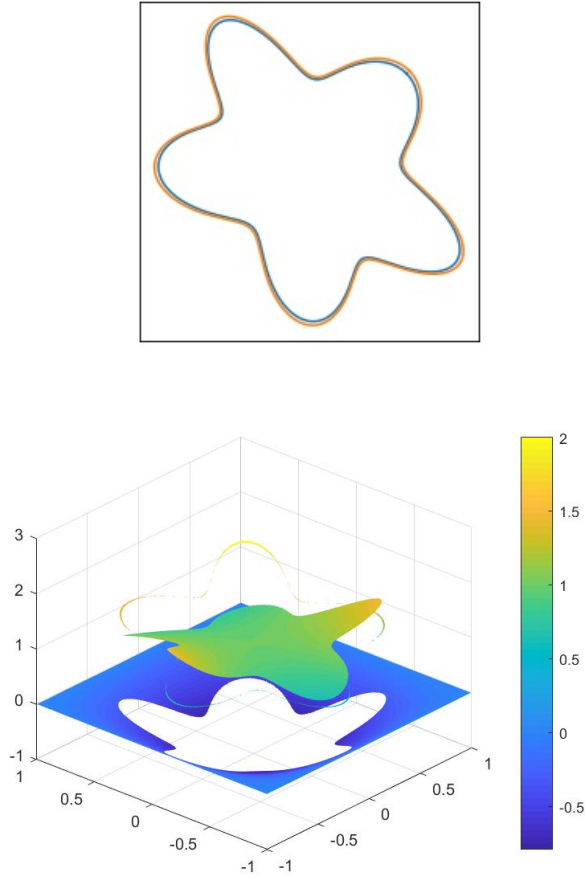


Fig. 9: (a) Closely-spaced interfaces used in example 4; (b) numerical solution using 512×512 mesh.

Table 5: Numerical results of example 4 using two closely-spaced interfaces.

Resolution	#GMRES	$\ e_h\ _{l^2}$	$\ e_h\ _{\infty}$	Time(sec)
64×64	21	1.40E-3	3.74E-3	0.62
128×128	23	3.50E-4	8.87E-4	2.85
256×256	25	8.83E-5	2.26E-4	13.67
512×512	26	2.32E-5	5.72E-5	63.95
1024×1024	27	5.55E-6	1.44E-5	283.25

For all interface problems, we also check the conditioning of the linear system Eq. (4.20). For resolutions from 64×64 to 512×512 , we find condition numbers (measured in l^2 -norm) range from $8.48E2$ to $7.0E4$, respectively, indicating the linear system is not ill-conditioned.

7 Concluding remarks

We have developed a kernel free boundary integral method for solving elliptical PDEs with variable coefficients. We focus our study on problems defined on a doubly-connected domain with multiple boundaries and interfaces. Unlike the traditional boundary integral methods, our methods do not require the explicit form of the Green's function (and special quadratures for evaluating the integrals). Instead, we compute boundary or volume integrals by solving an equivalent interface problem, using finite difference schemes on Cartesian mesh. Though the accuracy is limited primarily by the order of the finite difference scheme implemented, the kernel free methods offer a convenient way to evaluate integrals, especially PDEs with variable coefficients on a complex domain. Numerical example show that the computational complexity is linearly proportional to mesh size and the resulting linear system is well-conditioned. The iteration number of GMRES is independent of the mesh size. In future work, we will investigate *moving* boundary or interface problems, especially problems related to real physical applications and defined on complex geometry with non-uniform material properties.

8 Acknowledgement

S. L. and M. K acknowledge the support from the National Science Foundation, Division of Electrical, Communications and Cyber Systems grant ECCS-1927432. S. L. also acknowledges the partial support from the National Science Foundation, Division of Mathematical Sciences grant DMS-1720420.

References

1. C. Pozrikidis. *Boundary Integral and Singularity Methods for Linearized Viscous Flow*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 1992.
2. J. Sethian. *Level Set Methods*. Cambridge University Press, 1996.
3. C.W Hirt and B.D Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981.
4. R.J. LeVeque and Z. L. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 32:1019–1044, 1995.
5. C.S. Peskin. The immersed boundary method. *Acta Numer.*, pages 1–39, 2002.
6. David M. Ambrose, Michael Siegel, and Keyang Zhang. Convergence of the boundary integral method for interfacial stokes flow. *arXiv:2105.07056*, 2021.
7. Thomas Y. Hou, John Lowengrub, and Robert Krasny. Convergence of a point vortex method for vortex sheets. *SIAM Journal on Numerical Analysis*, 28(2):308–320, 1991.

8. Wenrui Hao, Bei Hu, Shuwang Li, and Lingyu Song. Convergence of boundary integral method for a free boundary system. *Journal of Computational and Applied Mathematics*, 334:128–157, 2018.
9. L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73:325–348, 1987.
10. H.Cheng, L. Greengard, and V. Rokhlin. A fast adaptive multipole algorithm in three dimensions. *J. Comput. Phys.*, 155:468–498, 1999.
11. Z. Liang, Z. Gimbutas, L. Greengard, J. Huang, and S. Jiang. A fast multipole method for the rotne-prager-yamakawa tensor and its applications. *J. Comput. Phys.*, 234:133–139, 2013.
12. J. Barnes and P. Hut. A hierarchical $O(N \log N)$ force-calculation algorithm. *Nature*, 324:446–449, 1986.
13. K. Lindsay and R. Krasny. A particle method and adaptive treecode for vortex sheet motion in three-dimensional flow. *J. Comput. Phys.*, 172:879–907, 2001.
14. H. Feng, A. Barua, S. Li, and X. Li. A parallel adaptive treecode algorithm for evolution of elastically stressed solids. *Commun. Comput. Phys.*, 15:365–387, 2014.
15. G. H. Shortley and R. Weller. Numerical solution of laplace’s equation. *Journal of Applied Physics*, 9:334–348, 1938.
16. L.-T. Cheng, F. Gibou, F. Fedkiw, and M. Kang. A second-order-accurate symmetric discretization of the poisson equation on irregular domains. *J. Computational Physics*, 176:205–227, 2002.
17. W.J. Ying and C. S. Henriquez. A kernel-free boundary integral method for elliptic boundary value problems. *J. Comp. Phys*, 227:1046–1074, 2007.
18. W.J. Ying and W. C. Wang. A kernel-free boundary integral method for implicitly defined surfaces. *J. Comp. Phys*, 252:606–624, 2013.
19. A. Mayo. The fast solution of poisson’s and the biharmonic equations on irregular regions. *SIAM J. Numer. Anal.*, 21:285–299, 1984.
20. A. Mayo. Fast high order accurate solution of laplace’s equation on irregular regions. *SIAM J. Sci. Statist. Comput.*, 6:144–157, 1985.
21. W.J. Ying and W.C. Wang. A kernel-free boundary integral method for variable coefficients elliptic pdes. *Communications in Computational Physics*, 15:1108–1140, 2014.
22. Wenjun Ying. A cartesian grid-based boundary integral method for an elliptic interface problem on closely packed cells. *Communications in Computational Physics*, 2018.
23. Yaning Xie and Wenjun Ying. A fourth-order kernel-free boundary integral method for the modified Helmholtz equation. *Journal of Scientific Computing*, 78(3):1632–1658, 2019.
24. Yaning Xie, Wenjun Ying, and Wei-Cheng Wang. A high-order kernel-free boundary integral method for the biharmonic equation on irregular domains. *Journal of Scientific Computing*, pages 1–19, 2019.
25. Yaning Xie and Wenjun Ying. A high-order kernel-free boundary integral method for incompressible flow equations in two space dimensions. *Numerical Mathematics-theory Methods and Applications*, 13(3):595–619, 2020.
26. Yaning Xie and Wenjun Ying. A fourth-order kernel-free boundary integral method for implicitly defined surfaces in three space dimensions. *Journal of Computational Physics*, 415, 2020.
27. Tim H. Beeson-Jones and Andrew W. Woods. On the selection of viscosity to suppress the saffman-taylor instability in a radially spreading annulus. *Journal of Fluid Mechanics*, 782:127–143, 2015.
28. Pedro H. A. Anjos and Shuwang Li. Weakly nonlinear analysis of the saffman-taylor problem in a radially spreading fluid annulus. *Phys. Rev. Fluids*, 5:054002, May 2020.
29. M. Zhao, Pedro H. A. Anjos, J. Lowengrub, and Shuwang Li. Pattern formation of the three-layer saffman-taylor problem in a radial hele-shaw cell. *Phys. Rev. Fluids*, 5:124005, Dec 2020.
30. C. Gin and P. Daripa. Stability results for multi-layer radial hele-shaw and porous media flows. *Physics of Fluids*, 27:012101, 2015.
31. M. Salameh, S. Singh, S. Li, and M. Krishnamurthy. Surrogate vibration modeling approach for design optimization of electric machines. *IEEE Transactions on Transportation Electrification*, 6:1126–1133, 2020.

-
32. Min-Jhe Lu, J. Lowengrub, W. Hao, and S. Li. Nonlinear simulation of vascular tumor growth with a necrotic core and chemotaxis. 2021.
 33. Min-Jhe Lu, C. Liu, J. Lowengrub, and S. Li. Complex far-field geometries determine the stability of solid tumor growth with chemotaxis. *Bull Math Biol*, 82:1–41, 2020.
 34. Y. Saad and M. H. Schultz. A generalized minimal residual method for solving non-symmetric linear systems. *SIAM J. Sci. Statist. Comput.*, 7:856–869, 1986.
 35. Y. Saad. *Iterative methods for sparse linear systems*. PWS Publishing Company, Boston, 1996.
 36. A. Greenbaum, L. Greengard, and G. B. McFadden. Laplace’s equation and the dirichlet-neumann map in multiply-connected domains. *Jour. Comp. Phys*, 105:267–278, 1993.
 37. J. T. Beale and A. T. Layton. On the accuracy of finite difference methods for elliptic problems with interfaces. *Commun. Appl. Math. Comput. Sci.*, 1:91–119, 2006.