An Isoparametric Finite Element Method for Elliptic Interface Problems with Nonhomogeneous Jump Conditions

XUFA FANG
Department of Mathematics
Zhejiang University
38 Zheda Road, 310027 Hangzhou
CHINA
woshimethod@163.com

Abstract: Numerical solutions of boundary value problems for elliptic equations with discontinuities in the coefficients and flux across immersed interface are of special interest. This paper develop a three order isoparametric finite element method for 2D elliptic interface problems. To obtain a high order of accuracy presents some difficulty, especially if the immersed interface does not fit with the elements. For this purpose, based on an initial Cartesian mesh, a body-fitted mesh optimization strategy is proposed by introducing curved boundary elements near the interface, and a quadratic isoparametric finite element basis is constructed on the optimized mesh. Numerical examples with immersed interval interface demonstrate that the proposed method is efficient for elliptic interface problems with nonhomogeneous flux jump condition.

Key–Words: Isoparametric finite element, Elliptic interface problems, Curved boundary element, Body-fitted mesh, Nonhomogeneous jump conditions

1 Introduction

Interface problems often appear in fluid mechanics, solid mechanics, electrodynamics, material science, biochemistry, etc. It is the case when two distinct materials or fluids with different conductivities or densities or diffusions are involved. These interfaces could be fixed or moving material interfaces, phase boundaries, flame fronts, physical boundaries, etc. When partial or ordinary differential equations are used to model these problems, the coefficients in the governing differential equations are typically discontinuous across the immersed interface, and the source terms may singular. Because of the low global regularity and the irregular geometry of the interface, the solutions to the differential equations are typically non-smooth, or even discontinuous a result, the standard numerical methods, such as finite difference and finite element methods, may fail in giving satisfactory numerical results for such problems.

The elliptic problems with internal interfaces are considered as standard benchmark problems. In many simulation, such as in fluid dynamics, material science, and biological systems etc, solving an elliptic equation with discontinuous coefficients and singular sources is the most expensive part.

A wide range of numerical methods have been devised in the past few decades. Peskin [19] developed immersed boundary method (IBM) basically to model blood flow in the heart. The immersed interface method (IIM) posed by LeVeque and Li [4] is a remarkable scheme that achieves the second order accuracy and preserves the jump information at the interface. In their method, a local correction is employed to incorporate the interface jump conditions. On the other hand, a fast iterative immersed interface method [5] (FIIM) is developed for elliptic interface problems with piecewise constant coefficients. Berthelsen [30] presents a decomposed immersed interface method (DIIM). In this scheme, the numerical discretization is well-defined by introducing a correction term to the standard central difference stencil, and the coefficient matrix tends to be symmetric and positive definite. Alternatively, Maximum principle preserving immersed interface method (MIIM) proposed by Li and K. Ito [7] enforces the resulting coefficient matrix to be an M-matrix.

The ghost fluid method (GFM) proposed by Fedkiw et al. [28] is relatively simple and easy-to-use approach. A high-order method which combines the merits of GFM and FIIM is the explicit-jump immersed interface method (EJIIM) proposed by Wiegmann and Bube [29], in which the high-order jumps at the intersections of the interface and the coordinate directions are regarded as auxiliary unknowns.

The jump condition capturing finite difference scheme proposed by Wang [26], develops a simple fi-

nite difference scheme by using a body-fitted curvilinear coordinate system. Matched interface boundary (MIB) method [16, 18] was initially introduced for solving the Maxwell's equation with material interfaces, later generalized to the solution of elliptic equations with discontinuous coefficients and singular sources [13, 14, 15, 17]. In order to employ the standard finite difference scheme throughout the computational domain, the fictitious values are used on irregular points . Interface problems with only discontinuous coefficients can apply the method of harmonic averaging [1, 31], in which the accurate coefficients is evaluated by taking the harmonic averages of them on uniform grid. The coupling interface method (CIM) introduced by Chern and Shu [22], derives a coupling equation for the principal derivatives to avoid unnecessary one-side interpolation.

In finite element formulation, Babuška et al. [2, 3] developed the generalized finite element method. In their method, local basis functions formed in an element can capture important features of the exact solution. The immersed finite element (IFE) methods [6, 8, 10, 12] also fall into this framework. Other methods such as finite volume based methods [25], the piecewise-polynomial discretization [24], integral equation methods [23], and discontinuous Galerkin techniques [27] have also generated much interest.

This paper is motivated by [9, 20]. The main contribution of this paper contains: (i) we generate a body-fitted mesh with curved boundary triangles based on the locally modified Cartesian mesh [20], (ii) we construct a quadratic isoparametric finite element based on the optimized mesh, (iii) we use such curved meshes and quadratic basis to solve elliptic interface problem using isoparametric finite element method. In our curved meshes, curved elements are used near the interval interface, and the immersed interfaces are discretized with piecewise quadratic curves, so we can expect a good approximation of the curved internal interface. As is known to all, to obtain a high order of accuracy presents some difficulty, especially if the immersed interface does not fit with the elements. The curved mesh casts new light on high order finite element methods. Isoparametric finite element can recover domains with piecewise quadratic boundary exactly, approximate general domains with a few degrees of freedom. Therefore it is a good tool to approximate interface with piecewise smooth boundary.

An outline of the paper is as follows. In section 2, we give model equations and derive the weak formulation from the given equations. In section 3, we detail the mesh generation process and the construction of quadratic isoparametric finite element basis. In section 4, we present numerical results. Finally, in Section 5, we give a brief conclusion .

2 Finite element method for elliptic interface problem

2.1 The model problem

Assume, $\Omega \subset \mathbb{R}^2$ is a bounded domain with its boundary $\partial\Omega$. The immersed interface Γ is a sufficiently smooth curve separates Ω into two subdomains Ω^-, Ω^+ such that $\Omega^- \cap \Omega^+ = \emptyset$, $\overline{\Omega} = \overline{\Omega^-} \cup \overline{\Omega^+}$ and $\overline{\Omega^-} \cap \overline{\Omega^+} = \Gamma$, where an overline denotes the closure; see the sketch in Fig.1. Without loss of generality, we assume that Ω is a rectangular domain.

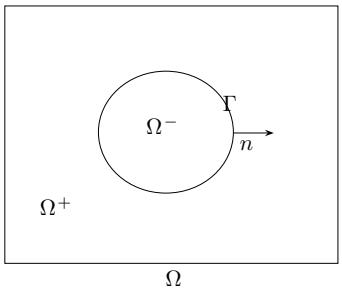

Figure 1: A rectangular domain Ω with an immersed interface Γ

In this paper, we consider the following elliptic interface problem:

$$-\nabla \cdot (\beta^- \nabla \boldsymbol{u}) = \boldsymbol{f} \quad \text{in } \Omega^-, \tag{1}$$

$$-\nabla \cdot (\beta^+ \nabla \boldsymbol{u}) = \boldsymbol{f} \quad \text{in } \Omega^+, \tag{2}$$

$$[\boldsymbol{u}]_\Gamma = 0, \tag{3}$$

$$\left[\beta \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}\right]_\Gamma = Q, \tag{4}$$

$$\boldsymbol{u} = g \qquad \text{on } \partial\Omega. \tag{5}$$

where $\boldsymbol{n}$ is the unit normal direction of the interface pointing outward, $\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}$ is the normal derivative of the solution $\boldsymbol{u}(\boldsymbol{x}, \boldsymbol{y})$.

The jumps are defined as the difference of the limiting values from each side of the interface. For example, the jump in the flux $[\beta \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}]_\Gamma$ at a point $\mathbf{P}$ on the interface is defined as

$$\begin{aligned}\left[\beta \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}\right]_{\mathbf{P}\in\boldsymbol{\Gamma}} &\stackrel{\text{def}}{=} \\ &\left(\beta^+ \lim_{\boldsymbol{x}\to\mathbf{P},\boldsymbol{x}\in\boldsymbol{\Omega}^+} \nabla \boldsymbol{u}(\boldsymbol{x}) - \right. \\ &\left. \beta^- \lim_{\boldsymbol{x}\to\mathbf{P},\boldsymbol{x}\in\boldsymbol{\Omega}^-} \nabla \boldsymbol{u}(\boldsymbol{x})\right) \cdot \boldsymbol{n}, \\ &= \beta^+ \frac{\partial \boldsymbol{u}^+}{\partial \boldsymbol{n}} - \beta^- \frac{\partial \boldsymbol{u}^-}{\partial \boldsymbol{n}}, \end{aligned} \tag{6}$$

2.2 The weak formulation

To derive the weak form of the interface problem (1)–(5), we multiply both sides of the equations (1) and (2) by a test function $v(\boldsymbol{x}, \boldsymbol{y}) \in H_0^1(\Omega)$, where $H_0^1(\Omega) = \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$, and integrate over the domains Ω^+ and Ω^- respectively

$$\iint_{\Omega^+} [-\nabla \cdot (\beta \nabla \boldsymbol{u})] v dxdy = \iint_{\Omega^+} \boldsymbol{f} v dxdy, \quad (7)$$

$$\iint_{\Omega^-} [-\nabla \cdot (\beta \nabla \boldsymbol{u})] v dxdy = \iint_{\Omega^-} \boldsymbol{f} v dxdy. \quad (8)$$

Applying Green's formula in the domains Ω^+ and Ω^- respectively, we obtain

$$\iint_{\Omega^+} \beta \nabla \boldsymbol{u} \nabla v dxdy + \int_\Gamma \beta^+ v^+ \frac{\partial \boldsymbol{u}^+}{\partial \boldsymbol{n}^+} ds - \int_{\partial\Omega} \beta v \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} ds = \iint_{\Omega^+} \boldsymbol{f} v dxdy, \quad (9)$$

$$\iint_{\Omega^-} \beta \nabla \boldsymbol{u} \nabla v dxdy - \int_\Gamma \beta^- v^- \frac{\partial \boldsymbol{u}^-}{\partial \boldsymbol{n}^+} ds = \iint_{\Omega^-} \boldsymbol{f} v dxdy, \quad (10)$$

where $\boldsymbol{n}^+ = \boldsymbol{n}$ is the unit normal direction of the interface Γ pointing outward. By applying the zero boundary condition $v|_{\partial\Omega} = 0$ and adding (9) and (10) together, we get

$$\iint_{\Omega^+} \beta \nabla \boldsymbol{u} \nabla v dxdy + \iint_{\Omega^-} \beta \nabla \boldsymbol{u} \nabla v dxdy + \int_\Gamma \beta^+ v^+ \frac{\partial \boldsymbol{u}^+}{\partial \boldsymbol{n}} ds - \int_\Gamma \beta^- v^- \frac{\partial \boldsymbol{u}^-}{\partial \boldsymbol{n}} ds = \iint_{\Omega^+} \boldsymbol{f} v dxdy + \iint_{\Omega^-} \boldsymbol{f} v dxdy. \quad (11)$$

According to the definition of the jump in the flux:

$$\left[\beta \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}\right] = \beta^+ \frac{\partial \boldsymbol{u}^+}{\partial \boldsymbol{n}} - \beta^- \frac{\partial \boldsymbol{u}^-}{\partial \boldsymbol{n}} = Q, \quad (12)$$

and the jump condition (3), we obtain

$$\iint_\Omega \beta \nabla \boldsymbol{u} \nabla v dxdy = \iint_\Omega \boldsymbol{f} v dxdy - \int_\Gamma Q v ds, \quad \text{for all } v \in H_0^1(\Omega). \quad (13)$$

Let

$$a(u, v) = \iint_\Omega \beta \nabla \boldsymbol{u} \nabla v dxdy \quad (14)$$

$$\langle \boldsymbol{f}, v \rangle = \iint_\Omega \boldsymbol{f} v dxdy - \int_\Gamma Q v ds. \quad (15)$$

The weak form of the interface problem (1) – (5) is to find $\boldsymbol{u} \in H_0^1(\Omega)$ such that

$$a(\boldsymbol{u}, v) = \langle \boldsymbol{f}, v \rangle, \quad \text{for all } v \in H_0^1(\Omega). \quad (16)$$

The weak form allows discontinuities in the coefficient and the normal derivatives of the solution.

We assume V_h is a finite dimensional subspace of $H_0^1(\Omega)$, then the Galerkin method for approximating the solution of problem (16) is to find a discrete solution $\boldsymbol{u}_h \in V_h$, such that

$$a(\boldsymbol{u}_h, v_h) = \langle f, v_h \rangle, \quad \text{for all } v_h \in V_h. \quad (17)$$

Let $(w_k)_{k=1}^L$ be a basis in the space V_h, then the solution of problem (17) can be written as $\boldsymbol{u}_h = \sum_{k=1}^L \boldsymbol{u}_k w_k$, where the coefficients $\boldsymbol{u}_k$ are solutions of the linear system

$$\sum_{k=1}^L a(w_k, w_l) \boldsymbol{u}_k = \langle f, w_l \rangle, \quad 1 \leqslant l \leqslant L. \quad (18)$$

The choices of V_h and its basis are very important in finite element theory. From numerical standpoint, a good choice of the basis is to make the resulting matrix possess as many zeros as possible. In the following section, we will construct a quadratic finite element based on a body-fitted mesh.

3 Construction of isoparametric finite element space

3.1 Generation of body-fitted mesh

The generation of a body-fitted triangulation with curved boundary elements based on a locally modified mesh [9, 20] is the main part of our algorithm, we detail the generation process in the following. For the purpose of illustration, in Fig.2 we show a locally modified mesh. For brevity, we omit the generation processes of locally modified mesh but refer the readers to [20] for the detail.

Assume that $\mathcal{T}_h$ is a body-fitted triangulation generated from a Cartesian grid using algorithm in [9, 20], associates with the step size h. We call a triangle element $T \in \mathcal{T}_h$ interface element, if more than two vertices of T are on the interface, and denote by T_I the union of all interface elements. To make a better approximation of the immersed interface, we introduce curved element. Firstly, find the intersections of the midnormal of the interface edge and interface, we call them as curved points below. Then, we apply a quadratic curve passes through two endpoints of this interface edge and the intersection replacing original interface edge.

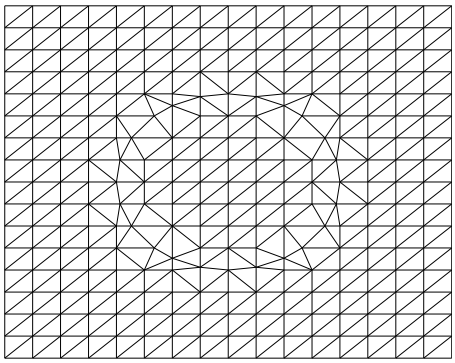

Figure 2: A locally modified mesh with straight boundary triangles.

For illustration, we detail the process in Fig.3. Assume ΔABC is a interface element, points B and C are on the interface Γ (the solid arc BDC), l is the midnormal of the line segment $\overline{BC}$, and D is the intersection of l and Γ. We apply a quadratic curve (the imaginary arcBDC) passes through points B, C and D replacing line segment $\overline{BC}$. Then we get a curved boundary triangle with two straight side: $\overline{AB}$ and $\overline{AC}$, and a quadratic curve edge: arc BDC (the imaginary curve Γ_r). For the ease of reference, we still call it interface element. The representation of the quadratic curve and interface element will be discussed in the next subsection.

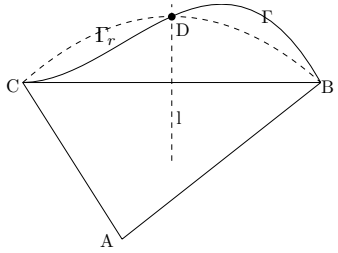

Figure 3: The construction of curved boundary triangle.

Fig.4 shows a optimized mesh derived from Fig.2. It is easy to check that the new body-fitted curved mesh $\mathcal{T}_h$ satisfies the following properties:

- The closure of Ω is covered exactly by $\mathcal{T}_h$, i.e., $\overline{\Omega} = \cup_{T\in\mathcal{T}_h} T$.
- If $T_1 \cap T_2 = \{x\}$ for $T_1, T_2 \in \mathcal{T}_h$ and some $\boldsymbol{x} \in \mathbb{R}^2$ then $\boldsymbol{x}$ is a vertex of both elements T_1 and T_2.
- If $T_1 \cap T_2 \supseteq \{\boldsymbol{x}, \boldsymbol{y}\}$ for $T_1, T_2 \in \mathcal{T}_h$ and distinct $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^2$ then T_1 and T_2 share an entire side.
- For each $T \in \mathcal{T}_h$, the set T is closed and the interior int(T) is not empty.

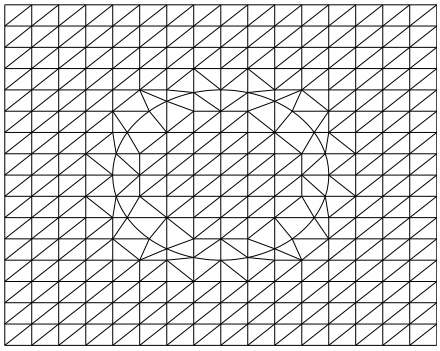

Figure 4: A locally modified mesh with curved boundary triangles.

- The interior of the elements is non-intersecting, i.e., int$(T_1)\cap$ int$(T_2) = \emptyset$ for each distinct $T_1, T_2 \in \mathcal{T}_h$.
- The boundary ∂T of each $T \in \mathcal{T}_h$ is Lipschitz-continuous.

3.2 Representation of isoparametric elements

We assume that $\overline{\Omega}$ is decomposed into finitely many triangular domains by triangulation $\mathcal{T}_h$, denote by T_I the union of interface elements, and let T_S be the union of straight boundary triangles, such that $\mathcal{T}_h = T_I \cup T_S$. The curved edge of each element $T \in T_I$ is a quadratic curve, which can be defined through a quadratic reference parameterization. If A and B are the endpoints of a curved edge E, and C is specified on E, then E is given by

$$\Phi_E : E_{ref} \to \mathbb{R}^2,$$
$$t \mapsto A\frac{t(t-1)}{2} + B\frac{t(1+t)}{2} + C(1-t)(1+t),$$

where $E_{ref} = [-1, 1]$. Obviously, the restriction of Φ_E to $(-1, 1)$ is an immersion since A, B and C are distinct. The endpoints of E_{ref} are mapped to the end points of E under the map Φ_E as in Fig.5.

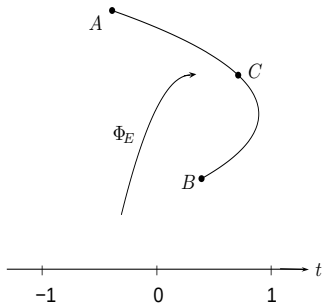

Figure 5: Immersion Φ_E that parameterizes an edge E defined through the points A, B, C.

For the representation of any element $T \in \mathcal{T}_h$ with three prescribed vertices $P_1^{(T)}$, $P_2^{(T)}$, $P_3^{(T)}$, the boundary ∂T consists of three smooth parameterized curves. Each of those curves interpolates vertices $A = P_i^{(T)}$ and $B = P_j^{(T)}$, where $i, j \in \{1, 2, 3\}$, and a point $C = P_{ij}^{(T)}$. Without loss of generality, we assume $i<j$. If both A and B are on the interface Γ, C is a curved point obtained in section 3.1, otherwise C is the midpoint of A and B, i.e.

$$P_{ij}^{(T)} = (P_i^{(T)} + P_j^{(T)})/2. \tag{19}$$

For a representation of the element T, we define functions $\psi_1, \psi_2, \psi_3 \in H^1(T_{ref})$ and $\psi_{12}, \psi_{23}, \psi_{13} \in H^1(T_{ref})$ such that each $T \in \mathcal{T}_h$ is the image of the map

$$\Psi_T : T_{ref} \to \mathbb{R}^2,$$
$$\Psi_T = \sum_{j=1}^{3} P_j^{(T)} \psi_j + \sum_{1 \leqslant i < j \leqslant 3} P_{ij}^{(T)} \psi_{ij}, \tag{20}$$

where $\psi_i, i = 1, 2, 3$ are the standard quadratic local

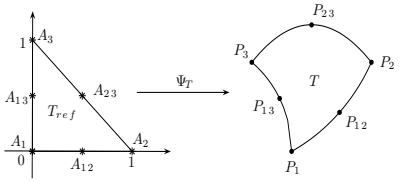

Figure 6: Diffeomorphisms Ψ_T on the reference triangle T_{ref} onto a curved boundary triangle.

nodal basis functions associated with the vertices A_i of T_{ref}:

$$\psi_i = \lambda_i(2\lambda_i - 1), \quad i = 1, 2, 3, \tag{21}$$

ψ_{ij}, are the standard quadratic local nodal basis functions associated with the midpoint A_{ij}:

$$\psi_{ij} = 4\lambda_i\lambda_j, \quad 1 \leqslant i < j \leqslant 3 \tag{22}$$

$\lambda_i, i = 1, 2, 3$ in (21) and (22) are the barycentric coordinates:

$$\lambda_1(r, s) = 1 - r - s, \tag{23}$$
$$\lambda_2(r, s) = r, \tag{24}$$
$$\lambda_3(r, s) = s. \tag{25}$$

Specially, if $T \in T_S$, (20) is equivalent to an affine transformation

$$\Psi = \sum_{j=1}^{3} P_j^{(T)} \lambda_j.$$

The vertices of T_{ref} are mapped to the vertices of T under the mapping Ψ_T, see Fig.6. Fig.7 displays the shape functions ψ_1, and ψ_{12}.

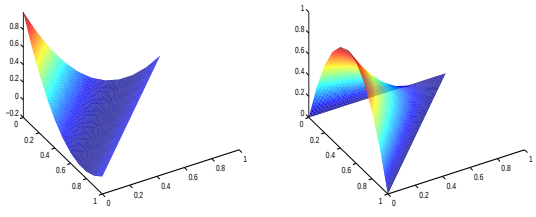

Figure 7: Shape functions ψ_1 (left), and ψ_{12} (right) on the reference triangle.

3.3 Quadratic isoparametric finite element space

With the help of the diffeomorphisms Ψ_T, and the functions ψ_1, ψ_2, ψ_3 and $\psi_{12}, \psi_{23}, \psi_{13}$, we define a discrete subspace $\mathcal{S}_h \subseteq H^1(\Omega)$ as follows. We denote by $\mathcal{N}$ the union of all vertices, curved points and midpoints, i.e.

$$\mathcal{N} := \{z \in \mathbb{R}^2 : \exists T \in \mathcal{T}_h, \exists j \in K_T, z = P_j^{(T)}\},$$

where

$$K_T = \{1, 2, 3, 12, 23, 13\}.$$

Given a node $z \in \mathcal{N}$, an element $T \in \mathcal{T}_h$ and $j \in K_T$, such that $z = P_j^{(T)} \in T$ we define

$$N_z|_T := \begin{cases} \psi_j \circ \Psi_T^{-1}, & \text{if } z \in T\,, \\ 0, & \text{if } z \notin T. \end{cases}$$

One easily checks $N_z \in H^1(\Omega)$. Thus, $\mathcal{S}_h$ consists of all functions which are linear combinations of functions N_z,

$$\begin{aligned} \mathcal{S}_h &:= \left\{ \sum_{z \in \mathcal{N}} \alpha_z N_z : \forall z \in \mathcal{N}, \alpha_z \in \mathbb{R} \right\} \\ &= \left\{ v \in H^1(\Omega) : \forall T \in \mathcal{S}_h, \forall j \in K_T, \right. \\ &\qquad \left. \exists \beta_j \in \mathbb{R}, v|_T = \sum_{j \in K_T} \beta_j \psi_j \circ \Psi_T^{-1} \right\}. \end{aligned}$$

Let $\mathcal{K} := \mathcal{N} \setminus \partial\Omega$, the space $S_h = \mathcal{S}_h \cap H_0^1(\Omega)$ is the span of all N_z with $z \in \mathcal{K}$,

$$S_h := \left\{ \sum_{z \in \mathcal{K}} \alpha_z N_z : \forall z \in \mathcal{K}, \alpha_z \in \mathbb{R} \right\}.$$

then the solution of discrete problem (17) can be written as

$$\boldsymbol{u}_h = \sum_{z\in\mathcal{K}} \boldsymbol{u}_z N_z \tag{26}$$

where $\boldsymbol{u}_z$ are solutions of linear system

$$\sum_{z\in\mathcal{K}} a(N_z, N_{z'})\boldsymbol{u}_z \;=\; \langle \boldsymbol{f}, N_{z'}\rangle \;\text{ for all } z' \in \mathcal{K}. \tag{27}$$

4 Numerical results

We present numerical experiments for the model problem (1)–(5) using the proposed locally modified mesh with curved boundary triangles and quadratic isoparametric finite element space discussed in section 3.3. Examples with known analytic solutions are considered, so that we can compute the error and check the convergence rate.

In the test problems below, the interface Γ being the zero level set of the function $\phi(\boldsymbol{x},\boldsymbol{y}) = 0$. The domain $\Omega = [-1,1]\times[-1,1]$, is divided into two subdomains: $\Omega^- := \{(\boldsymbol{x},\boldsymbol{y}) \in \Omega |\ \phi(\boldsymbol{x},\boldsymbol{y}) < 0\}$ and $\Omega^+ := \{(\boldsymbol{x},\boldsymbol{y}) \in \Omega |\ \phi(\boldsymbol{x},\boldsymbol{y}) > 0\}$. The exact solution $\boldsymbol{u}$ is a piecewise function

$$\boldsymbol{u}(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} \boldsymbol{u}^-(\boldsymbol{x},\boldsymbol{y}), & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^-, \\ \boldsymbol{u}^+(\boldsymbol{x},\boldsymbol{y}), & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^+. \end{cases} \tag{28}$$

The jump conditions, boundary conditions and the source terms are then determined from the exact solution.

Example 1. *This example is taken from [5]. We consider the problem (1)–(5) with the interface Γ is a circle centered at origin with radius $r_0 = 0.5$. The boundary condition and source term $\boldsymbol{f}$ are determined from the exact solution.*

$$\begin{aligned}
\phi(\boldsymbol{x},\boldsymbol{y}) &= \boldsymbol{x}^2 + \boldsymbol{y}^2 - (0.5)^2, \\
\beta(\boldsymbol{x},\boldsymbol{y}) &= \begin{cases} \beta^-, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^-, \\ \beta^+, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^+, \end{cases} \\
\boldsymbol{u}^-(\boldsymbol{x},\boldsymbol{y}) &= \frac{r^2}{\beta^-}, \\
\boldsymbol{u}^+(\boldsymbol{x},\boldsymbol{y}) &= \left(\frac{r_0^2}{\beta^-} - \frac{r_0^4 - 0.1\log(2r_0)}{\beta^+}\right) \\
&\quad + \frac{r^4 - 0.1\log(2r)}{\beta^+},
\end{aligned}$$

where $r = \sqrt{\boldsymbol{x}^2 + \boldsymbol{y}^2}$.

Fig.4 shows a mesh grid for this problems. Fig.8 shows plots of numerical solution and the error when $\beta^- = 1, \beta^+ = 2$. Tables 1–3 show the numerical results for different β^- and β^+. In this tables, $E_h = \boldsymbol{u} - \boldsymbol{u}_h$ is the error, where $\boldsymbol{u}$ is the exact solution and $\boldsymbol{u}_h$ is the numerical solution associate with the step size h, $\|\cdot\|_\infty$ and $\|\cdot\|_2$ are the standard L^∞–norm and L^2–norm respectively.

$$Order = \frac{\log(\| E_h \|_s \,/\, \| E_{h/2} \|_s)}{\log 2}, \; s = 2, \text{ or } \infty.$$

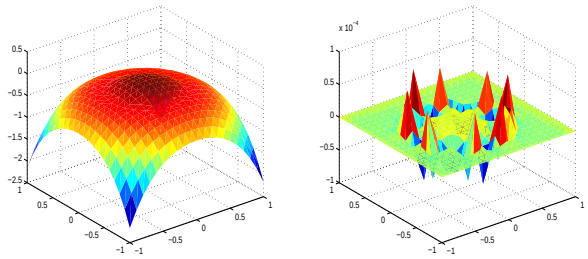

Figure 8: The plots of $-\boldsymbol{u}_h$ (Left) and error (Right) with $h = 1/10$, $\beta^- = 1, \beta^+ = 2$ for example 1.

Table 1: A grid refinement analysis for example 1 with $\beta^- = 1, \beta^+ = 2$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	5.3684e-4		1.0525e-3	
1/10	6.7633e-5	2.99	1.2927e-4	3.03
1/20	8.8465e-6	2.93	1.5961e-5	3.02
1/40	1.1584e-6	2.93	1.9822e-6	3.01
1/80	1.5667e-7	2.89	2.4744e-7	3.00

Table 2: A grid refinement analysis for example 1 with $\beta^- = 1, \beta^+ = 10000$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	3.5682e-4		1.5360e-4	
1/10	5.4062e-5	2.72	1.1569e-5	3.73
1/20	6.9804e-6	2.95	1.1403e-6	3.34
1/40	1.0492e-6	2.73	1.2670e-7	3.17
1/80	1.1270e-7	3.22	1.0433e-8	3.60

From tables 1 – 3 we can see that the convergence order of L^∞ and L^2 norm are all 3, which indicate that numerical solution $\boldsymbol{u}_h$ converges to the exact solution with convergence rate $O(h^3)$ in both L^∞ and L^2 norms.

Table 3: A grid refinement analysis for example 1. with $\beta^- = 10000, \beta^+ = 1$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	1.1313e-3		2.1125e-3	
1/10	1.2675e-4	3.16	2.5837e-4	3.03
1/20	1.7106e-5	2.89	3.1888e-5	3.02
1/40	2.3380e-6	2.87	3.9594e-6	3.01
1/80	3.1872e-7	2.87	4.9461e-7	3.00

Example 2. *This benchmark test is quoted from [8].*

$$\phi(\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{x}^2 + \boldsymbol{y}^2 - r_0^2,$$
$$\beta(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} \beta^-, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^-, \\ \beta^+, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^+, \end{cases}$$
$$\boldsymbol{u}^-(\boldsymbol{x},\boldsymbol{y}) = \frac{r^\alpha}{\beta^-},$$
$$\boldsymbol{u}^+(\boldsymbol{x},\boldsymbol{y}) = \frac{r^\alpha}{\beta^+} + (\frac{1}{\beta^-} - \frac{1}{\beta^+})r_0^\alpha + \delta(r - r_0),$$

where $r = \sqrt{\boldsymbol{x}^2 + \boldsymbol{y}^2}$, $r_0 = \pi/6.28$, $\alpha = 3$, β^-, β^+ *and* δ *are constants.*

Case 1. $\delta = 0$, $\boldsymbol{u}(\boldsymbol{x},\boldsymbol{y})$ satisfies the following jump conditions:

$$[\boldsymbol{u}]_\Gamma = 0, \quad \left[\beta\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}\right]_\Gamma = 0.$$

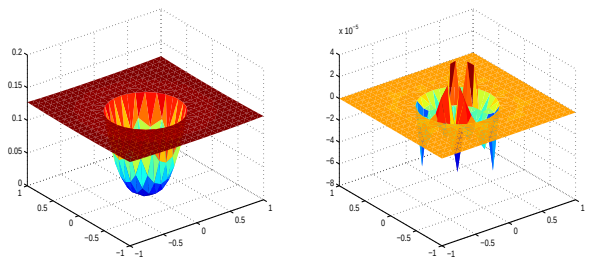

Figure 9: The numerical solution (Left) and error (Right) for case 1 with $\beta^- = 1, \beta^+ = 1000$

The numerical solution and error plots are shown in Fig.9. Numerical results for the cases $\beta^- = 1, \beta^+ = 1000$ and $\beta^- = 1000$, $\beta^+ = 1$ are listed in tables 4 and 5, respectively.

Case 2. $\delta = 10$, $\beta^- = 10000$, $\beta^+ = 10$, $\boldsymbol{u}(\boldsymbol{x},\boldsymbol{y})$ satisfies the following jump conditions:

$$[\boldsymbol{u}]_\Gamma = 0, \quad \left[\beta\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}}\right]_\Gamma = \delta\beta^+ = 100.$$

(Fig.10 and Table 6.)

Table 4: Numerical results for $\beta^- = 1$, $\beta^+ = 1000$ of case 1.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	8.5095e-4		3.1390e-4	
1/10	9.8245e-5	3.11	3.8262e-5	3.04
1/20	1.2464e-5	2.98	4.9347e-6	2.95
1/40	2.1891e-6	2.51	6.3705e-7	2.95
1/80	2.8467e-7	2.94	7.9281e-8	3.01

Table 5: Numerical results for example 2 $\beta^- = 1000$, $\beta^+ = 1$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	8.3152e-4		7.1782e-4	
1/10	8.4075e-5	3.31	8.6136e-5	3.06
1/20	1.1675e-5	2.85	1.0393e-5	3.05
1/40	1.5816e-6	2.88	1.2674e-6	3.04
1/80	2.1393e-7	2.89	1.5784e-7	3.01

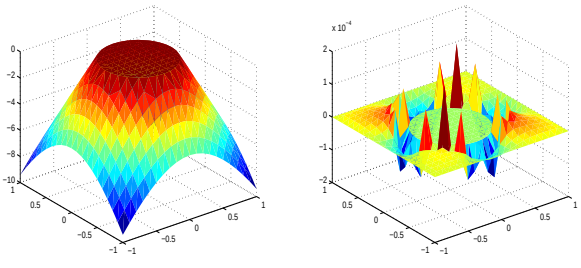

Figure 10: The plots of $-\boldsymbol{u}_h$ (Left) and error (Right) for case 2 with $h = 1/10$.

Table 6: Numerical results for case 2.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	1.5506e-3		2.3745e-3	
1/10	1.9460e-4	2.99	2.8878e-4	3.02
1/20	3.0252e-5	2.69	3.4838e-5	3.05
1/40	6.2922e-6	2.28	4.2070e-6	3.05
1/80	9.6685e-7	2.70	5.2720e-7	3.00

Example 3. *This example is quoted from [17, 22, 26]:*

$$\phi(\boldsymbol{x},\boldsymbol{y}) = \left(\frac{\boldsymbol{x}}{18/27}\right)^2 + \left(\frac{\boldsymbol{y}}{10/27}\right)^2 - 1,$$
$$\beta(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} \beta^-, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^-, \\ \beta^+, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^+, \end{cases}$$
$$\boldsymbol{u}(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} e^{\boldsymbol{x}}\cos\boldsymbol{y}, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^-, \\ 5e^{-\boldsymbol{x}^2-\boldsymbol{y}^2/2}, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^+, \end{cases}$$

We show the cases for $\beta^+ = 1$, $\beta^- = 10, 1000$ *as those in [17, 22, 26].*

Fig.11 gives a mesh with step size $h = 1/10$. Fig.12 shows the plots of numerical solution and error for the case $\beta^- = 10, \beta^+ = 1$. Numerical results are shown in tables 7 and 8.

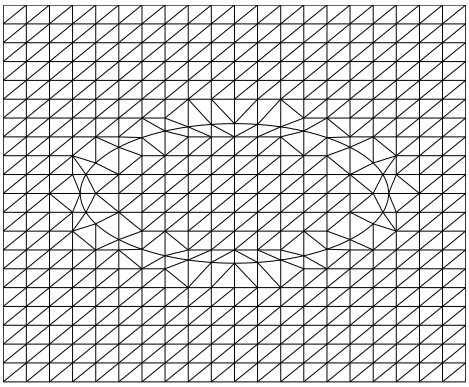

Figure 11: The mesh of example 3 with $h = 1/10$.

Table 7: Numerical results for example 3 with $\beta^- = 10, \beta^+ = 1$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	1.7024e-3		1.5917e-3	
1/10	1.9040e-4	3.16	1.7149e-4	3.21
1/20	5.8409e-5	1.70	2.2345e-5	2.94
1/40	3.6575e-6	4.00	2.3508e-6	3.25
1/80	5.8367e-7	2.65	2.8662e-7	3.04

Table 8: Numerical results for example 3 with $\beta^- = 1000, \beta^+ = 1$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	6.0616e-2		7.1417e-2	
1/10	4.8892e-3	3.63	5.7293e-3	3.64
1/20	8.0854e-4	2.60	9.1360e-4	2.65
1/40	3.4316e-5	4.56	3.8466e-5	4.57
1/80	2.4770e-6	3.80	2.5913e-6	3.89

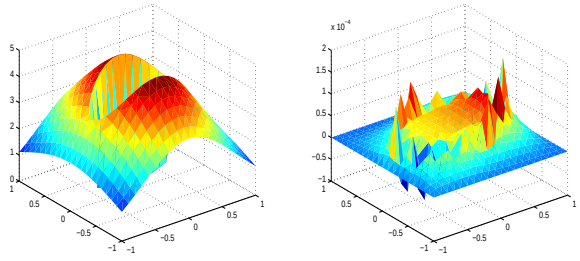

Figure 12: The solution and error of example 3 with $\beta^- = 10, \beta^+ = 1$.

Example 4. *This benchmark test is quoted from [4, 22]:*

$$\phi(\boldsymbol{x},\boldsymbol{y}) = r - 0.5,$$
$$\beta(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} 1 + r^2, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^-, \\ b, & (\boldsymbol{x},\boldsymbol{y}) \in \Omega^+, \end{cases}$$
$$\boldsymbol{u}^-(\boldsymbol{x},\boldsymbol{y}) = r^2,$$
$$\boldsymbol{u}^+(\boldsymbol{x},\boldsymbol{y}) = (r^4/2 + r^2 + 0.1\log(2r))/b - (0.5^4/2 + 0.5^2)/b + 0.5^2,$$
$$f(\boldsymbol{x},\boldsymbol{y}) = -8r^2 - 4,$$

where $r = \sqrt{\boldsymbol{x}^2 + \boldsymbol{y}^2}$ *and* b *is a parameter.*

We show the results for $b = 10$, 0.001 and 1000 (Fig13,Tables 9, 10 and 11).

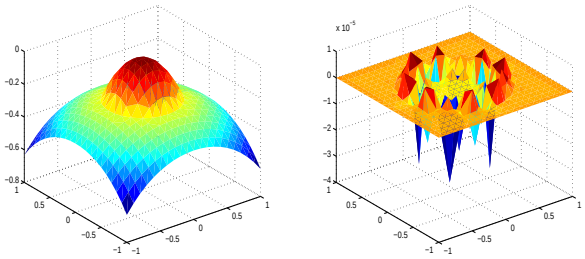

Figure 13: The plots of $-\boldsymbol{u}_h$ (Left) and error (Right) for example 4 with $h = 1/10$, $b = 10$.

Table 9: Numerical results for example 4 with $b = 10$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	3.5818e-4		1.8793e-4	
1/10	5.2587e-5	2.77	1.7700e-5	3.41
1/20	6.7062e-6	2.97	1.9867e-6	3.16
1/40	1.0277e-6	2.71	2.3546e-7	3.08
1/80	1.0723e-7	3.26	2.6932e-8	3.13

Table 10: Numerical results for example 4 with $b = 0.001$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	9.7880e-1		1.0993e-0	
1/10	8.2852e-2	3.56	1.3463e-1	3.03
1/20	1.1136e-2	2.90	1.6315e-2	3.04
1/40	1.5109e-3	2.88	1.9984e-3	3.03
1/80	2.3386e-4	2.66	2.4891e-4	3.01

Table 11: Numerical results for example 4 with $b = 1000$.

h	$\| E_h \|_\infty$	$Order$	$\| E_h \|_2$	$Order$
1/5	3.6634e-4		1.5300e-4	
1/10	5.4561e-5	2.75	1.1579e-5	3.72
1/20	7.0107e-6	2.96	1.1411e-6	3.34
1/40	1.0570e-6	2.73	1.2673e-7	3.17
1/80	1.1278e-7	3.23	1.0435e-8	3.60

5 Conclusion

In some applications one aims to approximate problems on rather general domains with a few degrees of freedom. Therefore, the approximation of non-polygonal domains is an important issue. Isoparametric finite elements can recover domains with piecewise quadratic boundary exactly and are therefore a good tool to approximate elliptic problems on domains with piecewise smooth boundary. In this paper, we construct a body-fitted triangular mesh with certain curved boundary triangles, which is generated from a uniform Cartesian mesh, then we use such mesh and isoparametric finite element method to solve elliptic interface problems with nonhomogeneous flux jump condition. A quadratic isoparametric finite element basis is discussed for computing the discrete solution. Numerical tests indicate that the proposed three order isoparametric finite element method is efficient and robust for solving elliptic interface problems, and has three order convergence in both L^∞ and L^2 norm. Compare to immersed finite element method [6, 8, 10, 12], the proposed method is simple and easy to use.

References:

[1] A. N. Tikhonov and A.A. Samarskii, Homogeneous difference schemes, USSR *Comput. Math. Math. Phys.* 1, 1962, pp. 5–67.

[2] I. Babuška, The finite element method for elliptic equations with discontinuous coefficients, *Comput.* 5, 1970, pp. 207–213.

[3] I. Babuška and J.E. Osborn, Finite element methods for the solution of problems with rough input data, In P. Grisvard, W. Wendland, and J.R. Whiteman, editors, *Singular and Constructive Methods for their Treatment, Lecture Notes in Mathematics, #1121,* Springer-Verlag, New York, 1985, pp. 1–18.

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

[5] Z.-L. Li, A fast iterative algorithm for elliptic interface problems, *SIAM J. Numer. Anal.* 35, 1998, pp. 230–254.

[6] Z.-L. Li, The immersed interface method using a finite element formulation, *Appl. Numer. Math.* 27, 1988, pp. 253–267.

[7] Z.-L. Li, K. Ito, Maximum principle preserving schemes for interface problems with discontinuous coefficients, *SIAM J. Sci. Comput.* 23, 2001, pp. 339–361.

[8] Z.-L. Li, T. Lin, and X. Wu, New Cartesian grid methods for interface problem using finite element formulation, *Numer. Math.* 96, 2003, pp. 61–98.

[9] H. Xie, K. Ito, Z.-L. Li, and J. Toivanen, A finite element method for interface problems with locally modified triangulation, *Contemporary Mathematics* 466, 2008, pp. 179–190.

[10] Y. Gong, B. Li, and Z.-L. Li, Immersed-interface finite-element methods for elliptic interface problemswith non-homogeneous jump conditions, *SIAM J. Numer. Anal.* 46, 2008, pp. 472–495.

[11] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems,* North-Holland Publishing Co., Amsterdam, 1978.

[12] X.-M. He, T. Lin and Y.-P. Lin, Immersed finite element methods for elliptic interface problems with non-homogeneous jump conditions, *Int. J. Numer. Anal. & Modeling* 8, 2011, pp. 284–301.

[13] S.-N. Yu, G.-W. Wei, Three-dimensional matched interface and boundary (MIB) method for treating geometric singularities, *J. Comput. Phys.* 227, 2007, pp. 602–632.

[14] K.-L. Xia, M. Zhan, D.-C. Wan, and G.–W. Wei, Adaptively deformed mesh based interface method for elliptic equations with discontinuous coefficients, *J. Comput. Phys.* 231, 2012, pp. 1440–1461.

[15] S.-N. Yu, Y.-C. Zhou, G.-W. Wei, Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces, *J. Comput. Phys.* 224(2), 2007, pp. 729–756.

[16] Y.-C. Zhou, G.-W. Wei, On the fictitious-domain and interpolation formulations of the matched interface and boundary (MIB) method, *J. Comput. Phys.* 219(1), 2006, pp. 228–246.

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

[18] S. Zhao, G.-W. Wei, Y. Xiang, Dsc analysis of free-edged beams by an iteratively matched boundary method, *J. Sound Vib.* 284(1-2), 2005, pp. 487–493.

[19] C. S. Peskin, Numerical analysis of blood flow in the heart, *J. Comput. Phys.* 25, 1977, pp. 220–252.

[20] C. Börgers, A triangulation algorithm for fast elliptic solvers based on domain imbedding, *SIAM J. Numer. Anal.* 27, 1990, pp. 1187–1196.

[21] J. Bramble, J. King, A finite element method for interface problems in domains with smooth boundaries and interfaces, *Advan. Comput. Math.* 6, 1996, pp. 109–138.

[22] I. Chern, Y.-C. Shu, A coupling interface method for elliptic interface problems, *J. Comput. Phys.* 225, 2007, pp. 2138–2174.

[23] A. Mayo, The fast solution of Poissons and the biharmonic equations on irregular regions, *SIAM J. Numer. Anal.* 21, 1984, pp. 285–299.

[24] T. Chen, J. Strain, Piecewise-polynomial discretization and Krylov-accelerated multigrid for elliptic interface problems, *J. Comput. Phys.* 227, 2008, pp. 7503–7542.

[25] M. Oevermann, R. Klein, A cartesian grid finite volume method for elliptic equations with variable coefficients and embedded interfaces, *J. Comput. Phys.* 219, 2006, pp. 749–769.

[26] W.-C. Wang, A jump condition capturing finite difference scheme for elliptic interface problems, *SIAM J. Sci. Comput.* 25, 2004, pp. 1479–1496.

[27] G. Guyomarch, C. O. Lee, A discontinuous Galerkin method for elliptic interface problems with application to electroporation, in:*AISTDAM Research Report*, 2004, pp. 04–14.

[28] R. P. Fedkiw, T. Aslam, B. Merriman and S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method), *J. Comput. Phys.* 152, 1999, pp. 457–492.

[29] A. Wiegmann, K. P. Bube, The explicit-jump immersed interface method: Finite difference methods for PDEs with piecewise smooth solutions, *SIAM J. Numer. Anal.* 37, 2000, pp. 827–862.

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

[31] G. R. Shubin and J. B. Bell, An analysis of the grid orientation effect in numerical simulation of miscible displacement, *Comput. Math. Appl. Mech. Eng.* 47, 1984, pp. 47–71.