

A Numerical Method for Solving Elasticity Equations with Interfaces

Songming Hou^{1,*}, Zhilin Li², Liqun Wang¹ and Wei Wang¹

¹ Department of Mathematics and Statistics, Louisiana Tech University, Ruston, LA, 71272, USA.

² Department of Mathematics, North Carolina State University, Raleigh, NC 27695, USA; and Nanjing Normal University, China.

Received 16 September 2010; Accepted (in revised version) 13 July 2011

Available online 20 February 2012

Abstract. Solving elasticity equations with interfaces is a challenging problem for most existing methods. Nonetheless, it has wide applications in engineering and science. An accurate and efficient method is desired. In this paper, an efficient non-traditional finite element method with non-body-fitting grids is proposed to solve elasticity equations with interfaces. The main idea is to choose the test function basis to be the standard finite element basis independent of the interface and to choose the solution basis to be piecewise linear satisfying the jump conditions across the interface. The resulting linear system of equations is shown to be positive definite under certain assumptions. Numerical experiments show that this method is second order accurate in the L^∞ norm for piecewise smooth solutions. More than 1.5th order accuracy is observed for solution with singularity (second derivative blows up) on the sharp-edged interface corner.

AMS subject classifications: 65N30

Key words: Elasticity equations, non-body fitted mesh, finite element method, jump condition.

1 Introduction

The importance of elasticity interface problems has been well recognized in a variety of disciplines. However, designing highly efficient methods for these problems is a difficult job, especially when the interface is not smooth.

Consider an open bounded domain $\Omega \subset R^d$. Let Γ be an interface of co-dimension $d-1$, which divides Ω into disjoint open subdomains, Ω^- and Ω^+ , hence $\Omega = \Omega^- \cup \Omega^+ \cup \Gamma$. Assume that the boundary $\partial\Omega$ and the boundary of each subdomain $\partial\Omega^\pm$ are Lipschitz

*Corresponding author. *Email addresses:* shou@latech.edu (S. Hou), zhilin@math.ncsu.edu (Z. Li), wliqunhmily@gmail.com (L. Wang), htprww@gmail.com (W. Wang)

continuous as submanifolds. Since $\partial\Omega^\pm$ are Lipschitz continuous, so is Γ . A unit normal vector of Γ can be defined a.e. on Γ , see Section 1.5 in [10].

We seek solutions of the variable coefficient elliptic equation away from the interface Γ given by

$$\begin{cases} -\nabla \cdot (\beta_1(x) \nabla u_1(x)) - \nabla \cdot (\beta_2(x) \nabla u_2(x)) = f_1(x) \\ -\nabla \cdot (\beta_3(x) \nabla u_1(x)) - \nabla \cdot (\beta_4(x) \nabla u_2(x)) = f_2(x), \quad x \in \Omega \setminus \Gamma, \end{cases} \quad (1.1)$$

in which $x = (x_1, \dots, x_d)$ denotes the spatial variables and ∇ is the gradient operator. The coefficient $\beta(x)$ is assumed to be a $d \times d$ matrix that is uniformly elliptic on each disjoint subdomain, Ω^- and Ω^+ , and its components are continuously differentiable on each disjoint subdomain, but they may be discontinuous across the interface Γ . The right-hand side $f(x)$ is assumed to lie in $L^2(\Omega)$.

Given functions a and b along the interface Γ , we prescribe the jump conditions

$$\begin{cases} [u_1]_\Gamma(x) \equiv u_1^+(x) - u_1^-(x) = a_1(x), \\ [u_2]_\Gamma(x) \equiv u_2^+(x) - u_2^-(x) = a_2(x), \\ n \cdot (\beta_1^+(x) \nabla u_1^+(x) + \beta_2^+(x) \nabla u_2^+(x)) \\ \quad - n \cdot (\beta_1^-(x) \nabla u_1^-(x) + \beta_2^-(x) \nabla u_2^-(x)) = b_1(x), \\ n \cdot (\beta_3^+(x) \nabla u_1^+(x) + \beta_4^+(x) \nabla u_2^+(x)) \\ \quad - n \cdot (\beta_3^-(x) \nabla u_1^-(x) + \beta_4^-(x) \nabla u_2^-(x)) = b_2(x). \end{cases} \quad (1.2)$$

The " \pm " superscripts refer to limits taken from within the subdomains Ω^\pm .

Finally, we prescribe boundary conditions

$$\begin{cases} u_1(x) = g_1(x), \quad x \in \partial\Omega, \\ u_2(x) = g_2(x), \quad x \in \partial\Omega, \end{cases} \quad (1.3)$$

for a given function g on the boundary $\partial\Omega$.

For simplicity, this paper discusses $d = 2$ case. The three dimensional $d = 3$ case is under investigation. The setup of the problem is illustrated in Fig. 1.

An elasticity system can be solved by both finite difference or finite element method. Due to the cross derivative term, usually the linear system of equations using the finite element formulation is better conditioned compared with that obtained using a finite difference discretization.

To solve the interface problem, first we need to generate a mesh. One approach is to use a body fitted mesh coupled with a finite element discretization, see for example, [1,3,5,6] for scalar elliptic partial differential equations. Recently, Cartesian meshes have become popular especially for moving interface problems to overcome the cost in the grid generation at every or every other time steps.

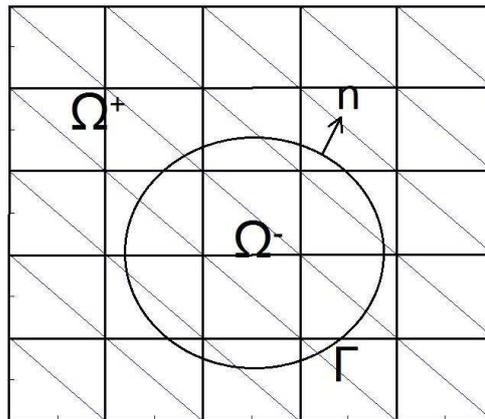


Figure 1: Setup of the problem with a uniform triangulation.

Finite difference methods are proposed in [22, 23] with non-homogeneous jump conditions. While second order accuracy was achieved, the condition number of the discrete system is quite large especially in the nearly incompressible case (λ is large) compared with that obtained from finite element formulations. In [22, 23], a first order immersed interface finite element method (IIFEM) was proposed using Cartesian meshes for the elasticity problem with *homogeneous* jump conditions. In general, the discretization using a finite element discretization has better conditioned system of equations compared with that obtained from a finite difference method. The Sobolev space theory provides strong theoretical foundations for convergence analysis for finite element methods.

In [8], an immersed-interface finite element method was developed for *scalar* elliptic interface problems with *non-homogeneous* jump conditions. In [9], the immersed-interface finite element method was developed for the elasticity system with homogeneous and non-homogeneous jump conditions. There are some other approaches to solve the elliptic interface problems. In particular, some recent work [12, 20, 21, 24] can handle sharp edged interfaces for piecewise smooth solutions. However, these have not been developed to solve the elasticity interface problems.

In this paper, based on the method in [12], we propose a numerical method for solving the elasticity problem with interfaces. We proved that the resulting linear system is non-symmetric but positive definite if β (notation introduced later) is positive definite. The method is simpler compared with that developed in [9] and can be applied for more general problems since we allow β_i to be matrices. We also proved a theorem that under certain assumption satisfied by the example in [9], for certain semi-positive definite β , the resulting linear system is still non-symmetric but positive definite.

The paper is organized as follows. In Section 2, we give the weak form of the elasticity system, our new numerical method and some theoretical analysis. In Section 3, we present some examples to demonstrate the performance of our method. We conclude and make acknowledgement in Section 4.

2 The weak formulation and the numerical method

We modify the weak formulation in [11] and [12]. The weak formulation for the system case is as follows:

$$\begin{aligned} & \int_{\Omega^+} (\beta_1 \nabla u_1 \cdot \nabla \psi_1 + \beta_2 \nabla u_2 \cdot \nabla \psi_1) + \int_{\Omega^-} (\beta_1 \nabla u_1 \cdot \nabla \psi_1 + \beta_2 \nabla u_2 \cdot \nabla \psi_1) \\ &= \int_{\Omega} f_1 \psi_1 + \int_{\Gamma} b_1 \psi_1, \end{aligned} \quad (2.1a)$$

$$\begin{aligned} & \int_{\Omega^+} (\beta_3 \nabla u_1 \cdot \nabla \psi_2 + \beta_4 \nabla u_2 \cdot \nabla \psi_2) + \int_{\Omega^-} (\beta_3 \nabla u_1 \cdot \nabla \psi_2 + \beta_4 \nabla u_2 \cdot \nabla \psi_2) \\ &= \int_{\Omega} f_2 \psi_2 + \int_{\Gamma} b_2 \psi_2, \end{aligned} \quad (2.1b)$$

where the test functions ψ_1, ψ_2 are in $H_0^1(\Omega)$. Note that the solution has jump across the interface.

For ease of discussion, we introduce some notations to write the system in a clean form.

We define

$$\begin{aligned} u &\equiv \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, & f &\equiv \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, & g &\equiv \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}, \\ a &\equiv \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, & b &\equiv \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, & \beta &\equiv \begin{bmatrix} \beta_1 & \beta_2 \\ \beta_3 & \beta_4 \end{bmatrix}, \end{aligned}$$

and choose the test function

$$\psi = \begin{bmatrix} \psi^1 \\ 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 0 \\ \psi^2 \end{bmatrix}.$$

We also redefine the gradient and divergence operator

$$\nabla \equiv \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ \frac{\partial}{\partial y} & 0 \\ 0 & \frac{\partial}{\partial x} \\ 0 & \frac{\partial}{\partial y} \end{bmatrix}, \quad \nabla \cdot \equiv \begin{bmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \end{bmatrix}.$$

Then Eq. (1.1) can be written as

$$-\nabla \cdot (\beta(x) \nabla u(x)) = f(x), \quad x \in \Omega \setminus \Gamma, \quad (2.2)$$

the jump condition Eq. (1.2) can be reformed as

$$\begin{cases} [u]_{\Gamma}(x) \equiv u^+(x) - u^-(x) = a(x), \\ n \cdot (\beta^+(x) \nabla u^+(x)) - n \cdot (\beta^-(x) \nabla u^-(x)) = b(x), \end{cases} \quad (2.3)$$

and boundary condition is

$$u(x) = g(x), \quad x \in \partial\Omega. \quad (2.4)$$

For simplicity we will discuss the following properties under the form of Eqs. (2.2)-(2.4).

For ease of discussion in this section, and for accuracy testing in the next section, we assume that a , b and c are smooth on the closure of Ω , β and f are smooth on Ω^+ and Ω^- , but they may be discontinuous across the interface Γ . However $\partial\Omega$, $\partial\Omega^-$ and $\partial\Omega^+$ are kept to be Lipschitz continuous. We assume that there is a Lipschitz continuous and piecewise smooth level-set function ϕ on Ω , where $\Gamma = \{\phi = 0\}$, $\Omega^- = \{\phi < 0\}$ and $\Omega^+ = \{\phi > 0\}$. A unit vector $n = \nabla\phi / |\nabla\phi|$ can be obtained on $\bar{\Omega}$, which is a unit normal vector of Γ at Γ pointing from Ω^- to Ω^+ .

In this paper, we restrict ourselves to a rectangular domain $\Omega = (x_{min}, x_{max}) \times (y_{min}, y_{max})$ in the plane, and β is a 2×2 matrix that is uniformly elliptic in each subdomain. Given positive integers I and J , set $\Delta x = (x_{max} - x_{min}) / I$ and $\Delta y = (y_{max} - y_{min}) / J$. We define a uniform Cartesian grid $(x_i, y_j) = (x_{min} + i\Delta x, y_{min} + j\Delta y)$ for $i = 0, \dots, I$ and $j = 0, \dots, J$. Each (x_i, y_j) is called a grid point. For the case $i = 0, I$ or $j = 0, J$, a grid point is called a boundary point, otherwise it is called an interior point. The grid size is defined as $h = \max(\Delta x, \Delta y) > 0$.

Two sets of grid functions are needed and they are denoted by

$$\begin{aligned} H_{\pm}^{1,h} &= \{ \omega^h = (\omega_{i,j}) : 0 \leq i \leq I, 0 \leq j \leq J \}, \\ H_0^{1,h} &= \{ \omega^h = (\omega_{i,j}) \in H_{\pm}^{1,h} : \omega_{i,j} = 0 \text{ if } i = 0, I \text{ or } j = 0, J \}. \end{aligned}$$

We cut every rectangular region $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$ into two pieces of right triangular regions: one is bounded by $x = x_i, y = y_j$ and

$$y = \frac{y_{j+1} - y_j}{x_i - x_{i+1}}(x - x_{i+1}) + y_j,$$

and the other is bounded by $x = x_{i+1}, y = y_{j+1}$ and

$$y = \frac{y_{j+1} - y_j}{x_i - x_{i+1}}(x - x_{i+1}) + y_j.$$

Collecting all those triangular regions, we obtain a uniform triangulation $T^h: \bigcup_{K \in T^h} K$, see Fig. 1. We can also choose the hypotenuse to be

$$y = \frac{y_{j+1} - y_j}{x_{i+1} - x_i}(x - x_i) + y_j,$$

and get another uniform triangulation from the same Cartesian grid. There is no conceptual difference for our method on these two triangulations.

If $\phi(x_i, y_j) \leq 0$, we count the grid point (x_i, y_j) as in $\overline{\Omega^-}$; otherwise we count it as in Ω^+ . We call an edge (an edge of a triangle in the triangulation) an interface edge if two of its ends (vertices of triangles in the triangulation) belong to different subdomains; otherwise we call it a regular edge.

We call a cell K an interface cell if its vertices belong to different subdomains. In the interface cell, we write $K = K^+ \cup K^-$. K^+ and K^- are separated by a straight line segment, denoted by Γ_K^h . The two end points of the line segment Γ_K^h are located on the interface Γ and their locations can be calculated from the linear interpolations of the discrete level-set functions $\phi^h = \phi(x_i, y_j)$. The vertices of K^+ are located in $\Omega^+ \cup \Gamma$ and the vertices of K^- are located in $\Omega^- \cup \Gamma$. K^+ and K^- are approximations of the regions of $K \cap \Omega^+$ and $K \cap \Omega^-$, respectively. We call a cell K a regular cell if all its vertices belong to the same subdomain, either Ω^+ or Ω^- . For a regular cell, we also write $K = K^+ \cup K^-$, where $K^- = \{\}$ (empty set) if all vertices of K are in Ω^+ , and $K^+ = \{\}$ if all vertices of K are in Ω^- . Clearly $\Gamma_K^h = \{\}$ in a regular cell, and K^+ and K^- are approximations of the regions $K \cap \Omega^+$ and $K \cap \Omega^-$, respectively. We use $|K^+|$ and $|K^-|$ to represent the areas of K^+ and K^- , respectively.

Two extension operators are needed. The first one is $T^h : H_{\pm}^{1,h} \rightarrow H_0^1(\Omega)$. For any $\psi^h \in H_0^{1,h}$, $T^h(\psi^h)$ is a standard continuous piecewise linear function, which is a linear function in every triangular cell and $T^h(\psi^h)$ matches ψ^h on grid points. Clearly such a function set, denoted by $H_0^{1,h}$, is a finite dimensional subspace of $H_0^1(\Omega)$. The second extension operator U^h is constructed as follows. For any $u^h \in H_{\pm}^{1,h}$ with $u^h = g^h$ at boundary points, $U^h(u^h)$ is a piecewise linear function and matches u^h on grid points. It is a linear function in each regular cell, just like the first extension operator $U^h(u^h) = T^h(u^h)$ in a regular cell. In each interface cell, it consists of two pieces of linear functions, one is on K^+ and the other is on K^- . The location of its discontinuity in the interface cell is the straight line segment Γ_K^h . Note that two end points of the line segment are located on the interface Γ , and hence the interface condition $[u] = a$ could be and is enforced exactly at these two end points. In each interface cell, the interface condition $[\beta \nabla u \cdot n] = b$ is enforced with the value b at the middle point of Γ_K^h . The extension U^h is unique, see Theorem 3.1 in [12].

We propose the following method:

Method 2.1. Find a discrete function $u^h \in H_{\pm}^{1,h}$ such that $u^h = g^h$ on boundary points and so that for all $\psi^h \in H_0^{1,h}$, we have

$$\begin{aligned} & \sum_{K \in T^h} \left(\int_{K^+} \beta \nabla U^h(u^h) \cdot \nabla T^h(\psi^h) + \int_{K^-} \beta \nabla U^h(u^h) \cdot \nabla T^h(\psi^h) \right) \\ &= \sum_{K \in T^h} \left(\int_{K^+} f T^h(\psi^h) + \int_{K^-} f T^h(\psi^h) + \int_{\Gamma_K^h} b T^h(\psi^h) \right). \end{aligned} \quad (2.5)$$

Note that we redefined the gradient and divergence operators so that the formula

above is in a clean form. Next we discuss in the component form. $u_{1x}^-, u_{1y}^-, u_{1x}^+, u_{1y}^+, u_{2x}^-, u_{2y}^-, u_{2x}^+$ and u_{2y}^+ can be written in the following form

$$u_{1x}^+ = c_{1x,1}^+ u(p_{11}) + c_{1x,2}^+ u(p_{12}) + c_{1x,3}^+ u(p_{13}) + c_{1x,4}^+ a(p_{14}) + c_{1x,5}^+ a(p_{15}) + c_{1x,6}^+ b(p_{16}) + c_{1x,7}^+ u(p_{21}) + c_{1x,8}^+ u(p_{22}) + c_{1x,9}^+ u(p_{23}) + c_{1x,10}^+ a(p_{24}) + c_{1x,11}^+ a(p_{25}) + c_{1x,12}^+ b(p_{26}), \quad (2.6a)$$

$$u_{1y}^+ = c_{1y,1}^+ u(p_{11}) + c_{1y,2}^+ u(p_{12}) + c_{1y,3}^+ u(p_{13}) + c_{1y,4}^+ a(p_{14}) + c_{1y,5}^+ a(p_{15}) + c_{1y,6}^+ b(p_{16}) + c_{1y,7}^+ u(p_{21}) + c_{1y,8}^+ u(p_{22}) + c_{1y,9}^+ u(p_{23}) + c_{1y,10}^+ a(p_{24}) + c_{1y,11}^+ a(p_{25}) + c_{1y,12}^+ b(p_{26}), \quad (2.6b)$$

$$u_{1x}^- = c_{1x,1}^- u(p_{11}) + c_{1x,2}^- u(p_{12}) + c_{1x,3}^- u(p_{13}) + c_{1x,4}^- a(p_{14}) + c_{1x,5}^- a(p_{15}) + c_{1x,6}^- b(p_{16}) + c_{1x,7}^- u(p_{21}) + c_{1x,8}^- u(p_{22}) + c_{1x,9}^- u(p_{23}) + c_{1x,10}^- a(p_{24}) + c_{1x,11}^- a(p_{25}) + c_{1x,12}^- b(p_{26}), \quad (2.6c)$$

$$u_{1y}^- = c_{1y,1}^- u(p_{11}) + c_{1y,2}^- u(p_{12}) + c_{1y,3}^- u(p_{13}) + c_{1y,4}^- a(p_{14}) + c_{1y,5}^- a(p_{15}) + c_{1y,6}^- b(p_{16}) + c_{1y,7}^- u(p_{21}) + c_{1y,8}^- u(p_{22}) + c_{1y,9}^- u(p_{23}) + c_{1y,10}^- a(p_{24}) + c_{1y,11}^- a(p_{25}) + c_{1y,12}^- b(p_{26}), \quad (2.6d)$$

$$u_{2x}^+ = c_{2x,1}^+ u(p_{11}) + c_{2x,2}^+ u(p_{12}) + c_{2x,3}^+ u(p_{13}) + c_{2x,4}^+ a(p_{14}) + c_{2x,5}^+ a(p_{15}) + c_{2x,6}^+ b(p_{16}) + c_{2x,7}^+ u(p_{21}) + c_{2x,8}^+ u(p_{22}) + c_{2x,9}^+ u(p_{23}) + c_{2x,10}^+ a(p_{24}) + c_{2x,11}^+ a(p_{25}) + c_{2x,12}^+ b(p_{26}), \quad (2.6e)$$

$$u_{2y}^+ = c_{2y,1}^+ u(p_{11}) + c_{2y,2}^+ u(p_{12}) + c_{2y,3}^+ u(p_{13}) + c_{2y,4}^+ a(p_{14}) + c_{2y,5}^+ a(p_{15}) + c_{2y,6}^+ b(p_{16}) + c_{2y,7}^+ u(p_{21}) + c_{2y,8}^+ u(p_{22}) + c_{2y,9}^+ u(p_{23}) + c_{2y,10}^+ a(p_{24}) + c_{2y,11}^+ a(p_{25}) + c_{2y,12}^+ b(p_{26}), \quad (2.6f)$$

$$u_{2x}^- = c_{2x,1}^- u(p_{11}) + c_{2x,2}^- u(p_{12}) + c_{2x,3}^- u(p_{13}) + c_{2x,4}^- a(p_{14}) + c_{2x,5}^- a(p_{15}) + c_{2x,6}^- b(p_{16}) + c_{2x,7}^- u(p_{21}) + c_{2x,8}^- u(p_{22}) + c_{2x,9}^- u(p_{23}) + c_{2x,10}^- a(p_{24}) + c_{2x,11}^- a(p_{25}) + c_{2x,12}^- b(p_{26}), \quad (2.6g)$$

$$u_{2y}^- = c_{2y,1}^- u(p_{11}) + c_{2y,2}^- u(p_{12}) + c_{2y,3}^- u(p_{13}) + c_{2y,4}^- a(p_{14}) + c_{2y,5}^- a(p_{15}) + c_{2y,6}^- b(p_{16}) + c_{2y,7}^- u(p_{21}) + c_{2y,8}^- u(p_{22}) + c_{2y,9}^- u(p_{23}) + c_{2y,10}^- a(p_{24}) + c_{2y,11}^- a(p_{25}) + c_{2y,12}^- b(p_{26}), \quad (2.6h)$$

where the locations of p_{ij} are shown in Fig. 2.

Compare with equations in [12], these equations are more complicated. When we calculate the gradient of u_1 , we need to take u_1, u_2 and the jump conditions into consideration. The formulas for the 96 coefficients are much more tedious to derive compared with the elliptic equation case in [12]. These coefficients are all independent of a, b . They depend on β , the grid size and where the interface cuts through the triangle.

Instead of explicitly deriving the 96 formulas, we could implement in the following simpler way: When a triangle is cut by the interface (See Fig. 2), we have four jump conditions. Therefore,

$$n_1 \cdot (\beta_1^+(p_{16}) \nabla u_1^+(x) + \beta_2^+(p_{16}) \nabla u_2^+(x)) - n_1 \cdot (\beta_1^-(p_{16}) \nabla u_1^-(x) + \beta_2^-(p_{16}) \nabla u_2^-(x)) = b_{p_{16}}(x), \quad (2.7a)$$

$$n_2 \cdot (\beta_3^+(p_{26}) \nabla u_1^+(x) + \beta_4^+(p_{26}) \nabla u_2^+(x)) - n_2 \cdot (\beta_3^-(p_{26}) \nabla u_1^-(x) + \beta_4^-(p_{26}) \nabla u_2^-(x)) = b_{p_{26}}(x), \quad (2.7b)$$

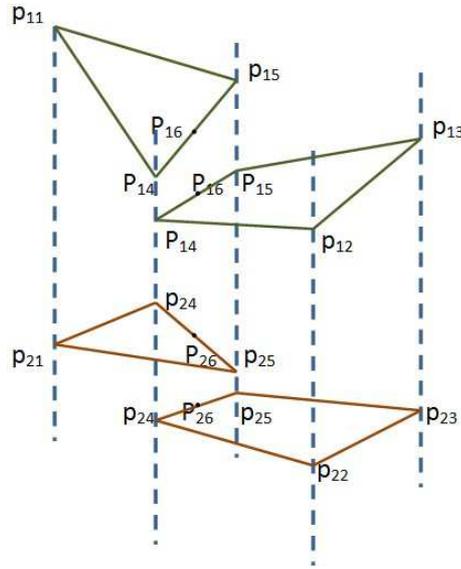


Figure 2: Local construction.

where

$$u_1^+(p_{14}) = u_1^-(p_{14}) + a_1(p_{14}), \tag{2.8a}$$

$$u_1^+(p_{15}) = u_1^-(p_{15}) + a_1(p_{15}), \tag{2.8b}$$

$$u_2^+(p_{14}) = u_2^-(p_{14}) + a_2(p_{14}), \tag{2.8c}$$

$$u_2^+(p_{15}) = u_2^-(p_{15}) + a_2(p_{15}). \tag{2.8d}$$

And $\nabla u_{1,2}^\pm$ can be written as a linear combination of the value of corner points on the corresponding triangle piece. Suppose $u(p_{11}), u(p_{12}), u(p_{13}), u(p_{21}), u(p_{22}), u(p_{23})$ are already known. $u(p_{15})$ can be denoted by $u(p_{14}), u(p_{12}), u(p_{13})$. $u(p_{25})$ can be denoted by $u(p_{24}), u(p_{22}), u(p_{23})$. Then we have Eq. (2.7), and two unknowns $u(p_{14}), u(p_{24})$. Solving these is equivalent to deriving the coefficient formulas in Eq. (2.6).

We use the Gaussian quadrature rule for integrals. The idea is the same as in [12]. If T is separated into two pieces by the interface $\overline{u_4 u_5}$, we connect u_3 and u_4 , then we get three triangles: $T_1 = \triangle u_1 u_4 u_5$, $T_2 = \triangle u_2 u_3 u_4$, $T_3 = \triangle u_3 u_4 u_5$. For each triangle, we label the center point p_{ij} of each edge $\overline{u_i u_j}$. In numerical computation, we apply the average of three $f(p_{ij})$ in each triangle. Numerical results show an improvement over [11], where fewer sample points were used.

Since our solution bases and test function bases are different, the matrix A for the linear system generated by Method 2.1 is not symmetric in the presence of an interface. However, we can prove it is positive definite.

Theorem 2.1. *If β is positive definite, then the matrix A for the linear system generated by Method 2.1 is positive definite.*

Proof. For all vector $c \in R^{2n}$, we have

$$c^T A c = B \left[\sum_{i=1}^{2n} c_i u_i, \sum_{i=1}^{2n} c_i \phi_i \right].$$

Here we can choose specific jump conditions to let

$$\sum_{i=1}^n c_i u^i = \sum_{i=1}^n c_i \psi^i = w,$$

(for detailed justification, see [12]), then since

$$B[w, w] = \int_{\Omega} (\beta(x) \nabla w(x))^T \nabla w(x) dx.$$

So if β is positive definite, then $\nabla w(x)^T \beta^T \nabla w(x) > 0, \forall x \in \Omega$. Therefore $B[w, w] > 0$, which implies $c^T A c > 0$. Thus, A is positive definite. \square

In some applications such as the epitaxial growth of thin films [2, 4], the matrix β is only semi-positive definite with zero determinant. The above Theorem does not apply. Numerical results for such problems is reported in [9]. However, the linear system was not proved to be positive definite. We prove below that when the matrix β is of certain form frequently appeared in applications and semi-positive definite, the matrix A generated by our method is still positive definite.

Theorem 2.2. *If $\lambda > 0, \mu > 0$ and*

$$\beta_1 = \begin{bmatrix} \lambda + 2\mu & 0 \\ 0 & \mu \end{bmatrix}, \quad \beta_2 = \begin{bmatrix} 0 & \lambda \\ \mu & 0 \end{bmatrix}, \quad \beta_3 = \begin{bmatrix} 0 & \mu \\ \lambda & 0 \end{bmatrix}, \quad \beta_4 = \begin{bmatrix} \mu & 0 \\ 0 & \lambda + 2\mu \end{bmatrix},$$

then the matrix A for the linear system generated by Method 2.1 is positive definite.

Proof. Suppose for a contradiction that A is not positive definite. Then there is a vector $c \in R^{2n}$ and $c \neq 0$ such that $c^T A c \leq 0$. Let

$$w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \sum_{i=1}^{2n} c_i \psi_i = \sum_{i=1}^{2n} c_i u_i,$$

then

$$\begin{aligned} & B[w, w] \leq 0 \\ \Rightarrow & \int_{\Omega} (\beta \nabla w(\vec{x}))^T \nabla w(x) d\vec{x} \leq 0 \\ \Rightarrow & \int_{\Omega} \begin{bmatrix} \frac{\partial w_1}{\partial x} & \frac{\partial w_1}{\partial y} & \frac{\partial w_2}{\partial x} & \frac{\partial w_2}{\partial y} \end{bmatrix} \begin{bmatrix} \lambda + 2\mu & 0 & 0 & \lambda \\ 0 & \mu & \mu & 0 \\ 0 & \mu & \mu & 0 \\ \lambda & 0 & 0 & \lambda + 2\mu \end{bmatrix} \begin{bmatrix} \frac{\partial w_1}{\partial x} \\ \frac{\partial w_1}{\partial y} \\ \frac{\partial w_2}{\partial x} \\ \frac{\partial w_2}{\partial y} \end{bmatrix} d\vec{x} \leq 0. \end{aligned}$$

Since for all $a = [a_1, a_2, a_3, a_4]^T \in R^4$,

$$a^T \beta a = (a_1 + a_4)^2 \lambda + 2(a_1^2 + a_4^2) \mu + (a_2 + a_3)^2 \mu \geq 0.$$

Hence $a^T \beta a = 0$ if and only if $a_1 = a_4 = 0$ and $a_2 = -a_3$. Then

$$\frac{\partial w_1}{\partial x}(\vec{x}) = a_1 = 0, \quad \forall \vec{x} \in \Omega.$$

However, $w_1 = \sum_{i=1}^n c_i \psi_i^1$ implies $\partial w_1 / \partial x = \sum_{i=1}^n c_i \partial \psi_i^1 / \partial x$. Since $c = [c_1, c_2, \dots, c_{2n}]^T \neq 0$, without loss of generality, we assume that $c_1 \neq 0$. If we choose point $\vec{x} \in \Omega$ such that $\partial \psi_1^1(\vec{x}) / \partial x \neq 0$ and $\partial \psi_i^1(\vec{x}) / \partial x = 0, i = 2, 3, \dots, n$, then $\sum_{i=1}^n c_i \partial \psi_i^1 / \partial x \neq 0$, a contradiction.

Therefore $c^T A c > 0 \forall c \neq 0$. Consequently, A is positive definite. □

Remark 2.1. A positive definite matrix has positive determinant, and is therefore invertible. It also has an LDM^T factorization where $D = \text{diag}(d_i)$ and $d_i > 0$, and L, M are lower triangular. The linear system $Ax = b$ can be solved efficiently.

More implementation details can be found in [11, 12]. Although the coupled system is more complicated to implement than the scalar case, the simplicity of our method makes the implementation manageable.

3 Numerical experiments

Consider the problem

$$\begin{cases} -\nabla \cdot (\beta_1 \nabla u_1) - \nabla \cdot (\beta_2 \nabla u_2) = f_1, & \text{in } \Omega^\pm, \\ -\nabla \cdot (\beta_3 \nabla u_1) - \nabla \cdot (\beta_4 \nabla u_2) = f_2, & \text{in } \Omega^\pm, \end{cases} \tag{3.1a}$$

with

$$\begin{cases} [u_1] = a_1, \quad [u_2] = a_2, & \text{on } \Gamma, \\ [(\beta_1 \nabla u_1 + \beta_2 \nabla u_2) \cdot n] = b_1, & \text{on } \Gamma, \\ [(\beta_3 \nabla u_1 + \beta_4 \nabla u_2) \cdot n] = b_2, & \text{on } \Gamma, \\ u_1 = g_1, \quad u_2 = g_2, & \text{on } \partial\Omega, \end{cases} \tag{3.1b}$$

on the rectangular domain $\Omega = (x_{\min}, x_{\max}) \times (y_{\min}, y_{\max})$. Γ is an interface and prescribed by the zero level-set $\{(x, y) \in \Omega \mid \phi(x, y) = 0\}$ of a level-set function $\phi(x, y)$. The unit normal vector of Γ is $n = \nabla \phi / |\nabla \phi|$ pointing from $\Omega^- = \{(x, y) \in \Omega \mid \phi(x, y) \leq 0\}$ to $\Omega^+ = \{(x, y) \in \Omega \mid \phi(x, y) \geq 0\}$.

In all numerical experiments below, the level-set function $\phi(x, y)$, the coefficients $\beta^\pm(x, y), p^\pm(x, y), q^\pm(x, y)$ and the solutions

$$\begin{cases} u_1 = u_1^+(x, y), \quad u_2 = u_2^+(x, y), & \text{in } \Omega^+, \\ u_1 = u_1^-(x, y), \quad u_2 = u_2^-(x, y), & \text{in } \Omega^-, \end{cases} \tag{3.2}$$

are given. Hence

$$\begin{cases} f_1 = -\nabla \cdot (\beta_1 \nabla u_1) - \nabla \cdot (\beta_2 \nabla u_2), \\ f_2 = -\nabla \cdot (\beta_3 \nabla u_1) - \nabla \cdot (\beta_4 \nabla u_2), \\ a_1 = u_1^+ - u_1^-, \quad a_2 = u_2^+ - u_2^-, \\ b_1 = (\beta_1^+ \nabla u_1^+ + \beta_2^+ \nabla u_2^+) \cdot n - (\beta_1^- \nabla u_1^- + \beta_2^- \nabla u_2^-) \cdot n, \\ b_2 = (\beta_3^+ \nabla u_1^+ + \beta_4^+ \nabla u_2^+) \cdot n - (\beta_3^- \nabla u_1^- + \beta_4^- \nabla u_2^-) \cdot n, \end{cases} \quad (3.3)$$

on the whole domain Ω , and g is obtained as a proper Dirichlet boundary condition, since the solutions are given.

All errors in solutions are measured in the L^∞ norm in the whole domain Ω . We present three numerical examples to demonstrate the effectiveness of our method.

Example 3.1. Our first example has a smooth interface. The level-set function $\phi(x, y) = x^2 + y^2 - 0.25$, the coefficients $\beta_1^\pm(x, y), \beta_2^\pm(x, y), \beta_3^\pm(x, y), \beta_4^\pm(x, y)$ and the solution $u_1^\pm(x, y), u_2^\pm(x, y)$ are given as follows:

$$\begin{aligned} \beta_1^+(x, y) &= \begin{bmatrix} x^2 + 3 & \sin(x+y) + 1 \\ 0.5\sin(x+y) + 0.7 & y^2 + 5 \end{bmatrix}, & \beta_1^-(x, y) &= \begin{bmatrix} x^2 + y^2 + 3 & \sin(xy) + 1 \\ \sin(x+y) + 1 & y^2 + 4 \end{bmatrix}, \\ \beta_2^+(x, y) &= \begin{bmatrix} \cos(x)^2 + 0.1 & (x+y)^2 + 2 \\ 2x^2 & 0.6\cos(x) + 1 \end{bmatrix}, & \beta_2^-(x, y) &= \begin{bmatrix} \cos(y) + 1 & (x+y)^2 + 1 \\ 2x^2 + 1 & 0.5\cos(x)^2 \end{bmatrix}, \\ \beta_3^+(x, y) &= \begin{bmatrix} \cos(x+y)^2 & 3x^2y^2 \\ x^2 + 1 & \cos(y) + 1 \end{bmatrix}, & \beta_3^-(x, y) &= \begin{bmatrix} 2\cos(x+y)^2 & 3x^2y^2 + 0.1 \\ 2x^2 & 2\cos(xy) + 2 \end{bmatrix}, \\ \beta_4^+(x, y) &= \begin{bmatrix} x^2y^2 + 5 & (\sin(x+2y))^2 \\ \sin(x+2y) + 1 & y^2 + x^2 + 3 \end{bmatrix}, & \beta_4^-(x, y) &= \begin{bmatrix} 0.5x^2y^2 + 4 & \sin(x) + 1 \\ \sin(x+y) + 1 & y^2 + x^2 + 4 \end{bmatrix}, \\ u_1^+(x, y) &= x^2 + y^2 - \sin(x+y), & u_1^-(x, y) &= (\sqrt{(x^2 + y^2)})^2, \\ u_2^+(x, y) &= 2y(x^3) + y^2, & u_2^-(x, y) &= (\sqrt{(x^2 + y^2)})^3. \end{aligned}$$

Figs. 3 and 4 show the numerical solution with our method using a 25×25 grid. Table 1 shows the error on different grids. The numerical result shows second order accuracy in the L^∞ norm for the solution.

Table 1: Circle shape interface.

$n_x \times n_y$	Err in U	Order
24×24	0.00558	
48×48	0.00147	1.92
96×96	3.76e-004	1.97
192×192	9.48e-005	1.99
384×384	2.39e-005	1.99

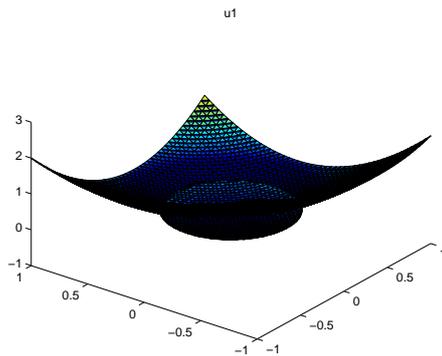


Figure 3: The solution u_1 for elasticity problem with a smooth circular interface.

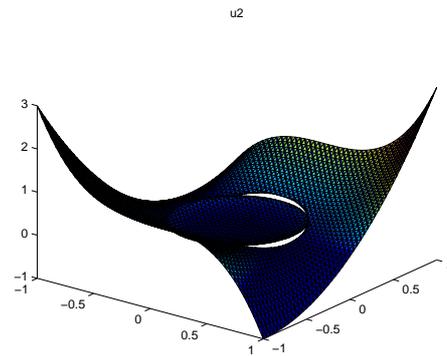


Figure 4: The solution u_2 for elasticity problem with a smooth circular interface.

Example 3.2. Our second example is a “happy face” interface with corners. The level-set function $\phi(x,y) = \max(\min(\phi_1, \phi_2, \phi_3), \phi_4, \phi_5, \phi_6, \min(\phi_7, \phi_8))$, the coefficients $\beta_1^\pm(x,y)$, $\beta_2^\pm(x,y)$, $\beta_3^\pm(x,y)$, $\beta_4^\pm(x,y)$ and the solution $u_1^\pm(x,y)$, $u_2^\pm(x,y)$ are given as follows:

$$\begin{aligned}\phi_1(x,y) &= x^2 + y^2 - 0.75^2 - 0.15^2, \\ \phi_2(x,y) &= (x - 0.75)^2 + y^2 - 0.15^2, \\ \phi_3(x,y) &= (x + 0.75)^2 + y^2 - 0.15^2, \\ \phi_4(x,y) &= -\frac{0.1}{0.12}(x - 0.2)^2 - \frac{0.12}{0.1}(y - 0.22)^2 + 0.12 \cdot 0.1, \\ \phi_5(x,y) &= -\frac{0.1}{0.12}(x + 0.2)^2 - \frac{0.12}{0.1}(y - 0.22)^2 + 0.12 \cdot 0.1, \\ \phi_6(x,y) &= -x^2 - (y + 0.08)^2 + 0.12^2, \\ \phi_7(x,y) &= -x^2 - (y + 0.625)^2 + 0.425^2, \\ \phi_8(x,y) &= -x^2 - (y + 0.25)^2 + 0.2^2,\end{aligned}$$

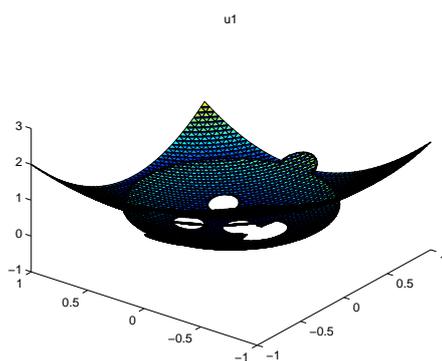


Figure 5: The solution u_1 for elasticity problem with a “Happy face” interface.

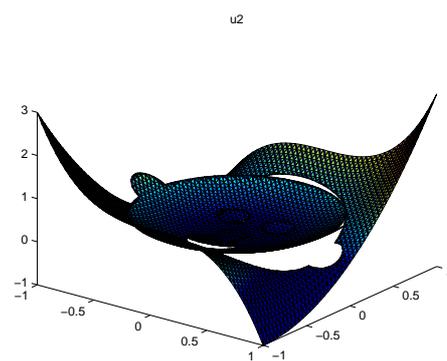


Figure 6: The solution u_2 for elasticity problem with a “Happy face” interface.

$$\begin{aligned}
 \beta_1^+(x,y) &= \begin{bmatrix} x^2+3 & \sin(x+y)+1 \\ 0.5\sin(x+y)+0.7 & y^2+5 \end{bmatrix}, & \beta_1^-(x,y) &= \begin{bmatrix} x^2+y^2+3 & \sin(xy)+1 \\ \sin(x+y)+1 & y^2+4 \end{bmatrix}, \\
 \beta_2^+(x,y) &= \begin{bmatrix} \cos(x)^2+0.1 & (x+y)^2+2 \\ 2x^2 & 0.6\cos(x)+1 \end{bmatrix}, & \beta_2^-(x,y) &= \begin{bmatrix} \cos(y)+1 & (x+y)^2+1 \\ 2x^2+1 & 0.5\cos(x)^2 \end{bmatrix}, \\
 \beta_3^+(x,y) &= \begin{bmatrix} \cos(x+y)^2 & 3x^2y^2 \\ x^2+1 & \cos(y)+1 \end{bmatrix}, & \beta_3^-(x,y) &= \begin{bmatrix} 2\cos(x+y)^2 & 3x^2y^2+0.1 \\ 2x^2 & 2\cos(xy)+2 \end{bmatrix}, \\
 \beta_4^+(x,y) &= \begin{bmatrix} x^2y^2+5 & (\sin(x+2y))^2 \\ \sin(x+2y)+1 & y^2+x^2+3 \end{bmatrix}, & \beta_4^-(x,y) &= \begin{bmatrix} 0.5x^2y^2+4 & \sin(x)+1 \\ \sin(x+y)+1 & y^2+x^2+4 \end{bmatrix}, \\
 u_1^+(x,y) &= x^2+y^2-\sin(x+y), & u_1^-(x,y) &= (\sqrt{(x^2+y^2)})^2, \\
 u_2^+(x,y) &= 2y(x^3)+y^2, & u_2^-(x,y) &= (\sqrt{(x^2+y^2)})^3.
 \end{aligned}$$

Figs. 5 and 6 show the numerical solution with our method using a 25×25 grid. Table 2 shows the error on different grids. The numerical result shows second order accuracy in the L^∞ norm for the solution and first order accuracy in the L^∞ norm for the gradient.

Table 2: Face shape interface.

$n_x \times n_y$	Err in U	Order
24×24	0.00663	
48×48	0.00178	1.89
96×96	4.71e-004	1.92
192×192	1.21e-004	1.96
384×384	3.16e-005	1.94

Example 3.3. Our third example is a “star” interface. The level-set function $\phi(x,y)$, the coefficients $\beta_1^\pm(x,y)$, $\beta_2^\pm(x,y)$, $\beta_3^\pm(x,y)$, $\beta_4^\pm(x,y)$ and the solution $u_1^\pm(x,y)$, $u_2^\pm(x,y)$ are given as follows:

$$\begin{aligned}
 \phi(r,\theta) &= \frac{R\sin(\theta_t/2)}{\sin(\theta_t/2+\theta-\theta_r-2\pi(i-1)/5)} - r, & \theta_r + \pi(2i-2)/5 \leq \theta < \theta_r + \pi(2i-1)/5, \\
 \phi(r,\theta) &= \frac{R\sin(\theta_t/2)}{\sin(\theta_t/2-\theta+\theta_r-2\pi(i-1)/5)} - r, & \theta_r + \pi(2i-3)/5 \leq \theta < \theta_r + \pi(2i-2)/5,
 \end{aligned}$$

with $\theta_t = \pi/5$, $\theta_r = \pi/7$, $R = 6/7$ and $i = 1, 2, 3, 4, 5$.

$$\begin{aligned}
 \beta_1^+(x,y) &= \begin{bmatrix} x^2+3 & \sin(x+y)+1 \\ 0.5\sin(x+y)+0.7 & y^2+5 \end{bmatrix}, & \beta_1^-(x,y) &= \begin{bmatrix} x^2+y^2+3 & \sin(xy)+1 \\ \sin(x+y)+1 & y^2+4 \end{bmatrix}, \\
 \beta_2^+(x,y) &= \begin{bmatrix} \cos(x)^2+0.1 & (x+y)^2+2 \\ 2x^2 & 0.6\cos(x)+1 \end{bmatrix}, & \beta_2^-(x,y) &= \begin{bmatrix} \cos(y)+1 & (x+y)^2+1 \\ 2x^2+1 & 0.5\cos(x)^2 \end{bmatrix}, \\
 \beta_3^+(x,y) &= \begin{bmatrix} \cos(x+y)^2 & 3x^2y^2 \\ x^2+1 & \cos(y)+1 \end{bmatrix}, & \beta_3^-(x,y) &= \begin{bmatrix} 2\cos(x+y)^2 & 3x^2y^2+0.1 \\ 2x^2 & 2\cos(xy)+2 \end{bmatrix}, \\
 \beta_4^+(x,y) &= \begin{bmatrix} x^2y^2+5 & (\sin(x+2y))^2 \\ \sin(x+2y)+1 & y^2+x^2+3 \end{bmatrix}, & \beta_4^-(x,y) &= \begin{bmatrix} 0.5x^2y^2+4 & \sin(x)+1 \\ \sin(x+y)+1 & y^2+x^2+4 \end{bmatrix},
 \end{aligned}$$

$$u_1^+(x,y) = x^2 + y^2 - \sin(x+y), \quad u_1^-(x,y) = \left(\sqrt{x^2 + y^2}\right)^2,$$

$$u_2^+(x,y) = 2y(x^3) + y^2, \quad u_2^-(x,y) = \left(\sqrt{x^2 + y^2}\right)^3.$$

Figs. 7 and 8 show the numerical solution with our method using a 25×25 grid. Table 3 shows the error on different grids. The numerical result shows second order accuracy in the L^∞ norm for the solution and first order accuracy in the L^∞ norm for the gradient.

Table 3: Star shape interface.

$n_x \times n_y$	Err in U	Order
24×24	0.00533	
48×48	0.00159	1.75
96×96	4.22e-004	1.91
192×192	1.10e-004	1.94
384×384	2.90e-005	1.93

Example 3.4. The solution in fourth example has singularity on the interface corner. The level-set function $\phi(x,y) = (x-0.4)^2 + y^2 - 0.16$, the coefficients $\beta_1^\pm(x,y)$, $\beta_2^\pm(x,y)$, $\beta_3^\pm(x,y)$, $\beta_4^\pm(x,y)$ and the solution $u_1^\pm(x,y)$, $u_2^\pm(x,y)$ are given as follows:

$$\beta_1^+(x,y) = \begin{bmatrix} x^2+3 & \sin(x+y)+1 \\ 0.5\sin(x+y)+0.7 & y^2+5 \end{bmatrix}, \quad \beta_1^-(x,y) = \begin{bmatrix} x^2+y^2+3 & \sin(xy)+1 \\ \sin(x+y)+1 & y^2+4 \end{bmatrix},$$

$$\beta_2^+(x,y) = \begin{bmatrix} \cos(x)^2+0.1 & (x+y)^2+2 \\ 2x^2 & 0.6\cos(x)+1 \end{bmatrix}, \quad \beta_2^-(x,y) = \begin{bmatrix} \cos(y)+1 & (x+y)^2+1 \\ 2x^2+1 & 0.5\cos(x)^2 \end{bmatrix},$$

$$\beta_3^+(x,y) = \begin{bmatrix} \cos(x+y)^2 & 3x^2y^2 \\ x^2+1 & \cos(y)+1 \end{bmatrix}, \quad \beta_3^-(x,y) = \begin{bmatrix} 2\cos(x+y)^2 & 3x^2y^2+0.1 \\ 2x^2 & 2\cos(xy)+2 \end{bmatrix},$$

$$\beta_4^+(x,y) = \begin{bmatrix} x^2y^2+5 & (\sin(x+2y))^2 \\ \sin(x+2y)+1 & y^2+x^2+3 \end{bmatrix}, \quad \beta_4^-(x,y) = \begin{bmatrix} 0.5x^2y^2+4 & \sin(x)+1 \\ \sin(x+y)+1 & y^2+x^2+4 \end{bmatrix},$$

$$u_1^+(x,y) = (x^2+y^2)^{5/6}, \quad u_1^-(x,y) = 1, \quad u_2^+(x,y) = x, \quad u_2^-(x,y) = 0.$$

Figs. 9 and 10 show the numerical solution with our method using a 25×25 grid. Table 4 shows the error on different grids.

Table 4: Singular point on the interface.

$n_x \times n_y$	Err in U	Order
24×24	0.00347	
48×48	0.00118	1.55
96×96	4.05e-004	1.55
192×192	1.39e-004	1.54
384×384	4.78e-005	1.54

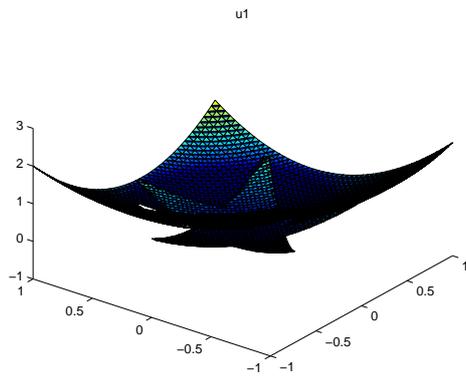


Figure 7: The solution u_1 for elasticity problem with a "star" interface.

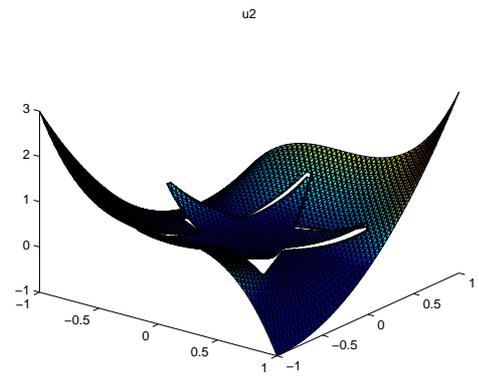


Figure 8: The solution u_2 for elasticity problem with a "star" interface.

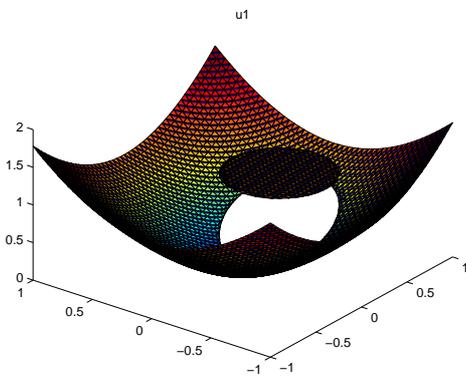


Figure 9: The solution u_1 for elasticity problem with a singular point on the interface.

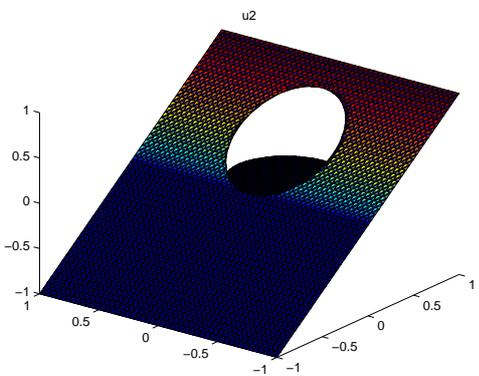


Figure 10: The solution u_2 for elasticity problem with a singular point on the interface.

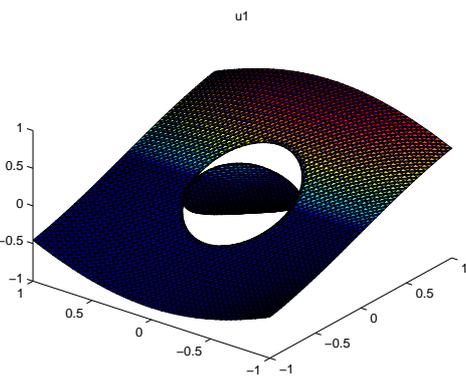


Figure 11: The solution u_1 for elasticity problem with special form of coefficients.

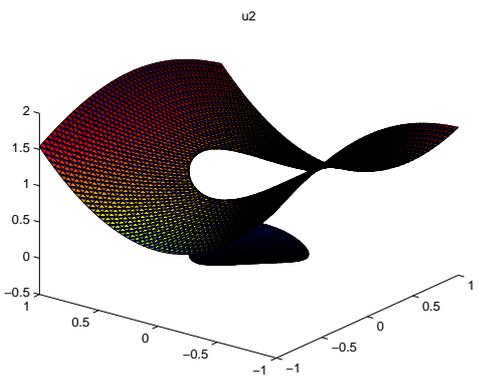


Figure 12: The solution u_2 for elasticity problem with special form of coefficients.

Example 3.5. Our final example has special form of coefficients that satisfies the hypothesis of Theorem 2.2. The level-set function $\phi(x, y) = x^2 + y^2 - 0.16$, the coefficients $\beta_1^\pm(x, y)$, $\beta_2^\pm(x, y)$, $\beta_3^\pm(x, y)$, $\beta_4^\pm(x, y)$ and the solution $u_1^\pm(x, y)$, $u_2^\pm(x, y)$ are given as follows:

$$\begin{aligned} \beta_1^+(x, y) &= \begin{bmatrix} 8 & 0 \\ 0 & 4 \end{bmatrix}, & \beta_1^-(x, y) &= \begin{bmatrix} 7 & 0 \\ 0 & 2 \end{bmatrix}, \\ \beta_2^+(x, y) &= \begin{bmatrix} 0 & 2 \\ 4 & 0 \end{bmatrix}, & \beta_2^-(x, y) &= \begin{bmatrix} 0 & 3 \\ 2 & 0 \end{bmatrix}, \\ \beta_3^+(x, y) &= \begin{bmatrix} 0 & 4 \\ 2 & 0 \end{bmatrix}, & \beta_3^-(x, y) &= \begin{bmatrix} 0 & 2 \\ 3 & 0 \end{bmatrix}, \\ \beta_4^+(x, y) &= \begin{bmatrix} 4 & 0 \\ 0 & 8 \end{bmatrix}, & \beta_4^-(x, y) &= \begin{bmatrix} 2 & 0 \\ 0 & 7 \end{bmatrix}, \\ u_1^+(x, y) &= \sin(x)\cos(y), & u_1^-(x, y) &= x\sin(y), \\ u_2^+(x, y) &= \cos(x) + y^2, & u_2^-(x, y) &= xy. \end{aligned}$$

Figs. 11 and 12 show the numerical solution with our method using a 25×25 grid. Table 5 shows the error on different grids.

Table 5: Special form of coefficients.

$n_x \times n_y$	Err in U	Order
24×24	0.00151	
48×48	4.44e-004	1.77
96×96	1.20e-004	1.89
192×192	3.30e-005	1.86
384×384	8.66e-006	1.93

4 Conclusion

In this paper, we modified the method in [12] to solve the elasticity problem with interfaces. We proved that the matrix for the linear system generated by our method is positive definite (but not symmetric) under certain assumptions. Through numerical experiments, our method achieved close to second order accuracy in the L^∞ norm for piecewise smooth solutions, and we can handle the difficulties of sharp-edged interfaces, with the solution being singular at the interface corner.

Acknowledgments

This research is partially supported by Louisiana Board of Regents RCS Grant No. LEQSF (2008-11)-RD-A-18. The second author is partially supported by the ARO grants 56349MA-

MA, the AFSOR grant FA9550-09-1-0520, the NSF grant DMS-0911434, the NIH grant 096195-01 and CNFS 11071123.

References

- [1] I. Babuška. The finite element method for elliptic equations with discontinuous coefficients. *Computing*, 5:207–213, 1970.
- [2] E. B'ansch, F. Hausser, O. Lakkis, B. Li, and A. Voigt. Finite element method for epitaxial growth with attachment-detachment kinetics. *J. Comput. Phys.*, 194:409–434, 2004.
- [3] J. Bramble and J. King. A finite element method for interface problems in domains with smooth boundaries and interfaces. *Advances in Comput. Math.*, 6:109–138, 1996.
- [4] W. K. Burton, N. Cabrera, and F. C. Frank. The growth of crystals and the equilibrium of their surfaces. *Phil. Trans. Roy. Soc. London Ser. A*, 243(866):299-358, 1951.
- [5] K. Chan, K. Zhang, X. Liao, J. Zou, and G. Schubert. A three-dimensional spherical nonlinear interface dynamo. *Astrophys. J.*, 596:663–679, 2003.
- [6] Z. Chen and J. Zou. Finite element methods and their convergence for elliptic and parabolic interface problems. *Numer. Math.*, 79:175–202, 1998.
- [7] S. Deng, K. Ito, and Z. Li. Three dimensional elliptic solvers for interface problems and applications. *J. Comput. Phys.*, 184:215–243, 2003.
- [8] Y. Gong, B. Li, and Z. Li. Immersed-interface finite-element methods for elliptic interface problems with non-homogeneous jump conditions. *SIAM J. Numer. Anal.*, 46:472–495, 2008.
- [9] Y. Gong and Z. Li. Immersed Interface Finite Element Methods for Elasticity Interface Problems with Non-homogeneous Jump Conditions *Numerical Mathematics: Theory, Methods and Applications*, in press.
- [10] P. Grisvard. *Elliptic Problems in Nonsmooth Domains - Monographs and Studies in Mathematics*. Pitman Advanced Publishing Program, ISSN 0743-0329, 1985.
- [11] S. Hou and X.-D. Liu. A numerical method for solving variable coefficient elliptic equations with interfaces. *J. Comput. Phys.*, 202:411–445, 2005.
- [12] S. Hou, W. Wang, L. Wang. Numerical method for solving matrix coefficient elliptic equation with sharp-edged interfaces. *J. Comput. Phys.*, 229:7162–7179, 2010.
- [13] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.
- [14] Z. Li. A fast iterative algorithm for elliptic interface problems. *SIAM J. Numer. Anal.*, 35:230–254, 1998.
- [15] Z. Li. The immersed interface method using a finite element formulation. *Applied Numer. Math.*, 27:253–267, 1998.
- [16] Z. Li, T. Lin, and X. Wu. New Cartesian grid methods for interface problem using finite element formulation. *Numer. Math.*, 96:61–98, 2003.
- [17] X.-D. Liu, R. P. Fedkiw, and M. Kang. A boundary condition capturing method for Poisson's equation on irregular domains. *J. Comput. Phys.*, 160 (1):151-178, 2000.
- [18] X.-D. Liu and T. Sideris. Convergence of the ghost fluid method for elliptic equations with interfaces. *Math. Comp.*, 72:1731-1746, 2003.
- [19] J. Necas. *Introduction to the theory of nonlinear elliptic equations*. Teubner-Texte zur Mathematik, Band 52, ISSN 0138-502X, 1983.
- [20] S. Yu, Y. Zhou, and G. W. Wei. Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces. *J. Comput. Phys.*, 224:729-756, 2007.

- [21] M. Oevermann, C. Scharfenberg, and R. Klein A sharp interface finite volume method for elliptic equations on Cartesian grids. *J. Comput. Phys.*, 228:5184-5206, 2009.
- [22] X. Yang. Immersed interface method for elasticity problems with interfaces. PhD thesis, North Carolina State University, 2004.
- [23] X. Yang, B. Li, and Z. Li. The immersed interface method for elasticity problems with interface. *Dynamics of Continuous, Discrete and Impulsive Systems*, 10:783–808, 2003.
- [24] J. Bedrossian, J. Von Brecht, S. Zhu, E. Sifakis, and J. Teran A Second Order Virtual Node Method for Poisson Interface Problems on Irregular Domains *J. Comput. Phys.*, 229:6405-6426, 2010