An Immersed Finite Element Method for Elasticity Equations with Interfaces

Zhilin Li and Xingzhou Yang

Abstract. The immersed finite element method based on a uniform Cartesian mesh has been developed for the linear elasticity equations with discontinuous coefficients across an interface in this paper. The interface does not have to be aligned with the mesh. The main idea is to modify the basis function over those triangles in which the interface cuts through so that the natural interface conditions are satisfied. The standard linear basis functions are used for other triangles. The interface is represented by a level set function. Numerical examples are also presented.

1. Introduction

In this paper, we develop the immersed finite element method (IFEM) based on a uniform Cartesian mesh for the two-phase elasticity system of the following:

$$\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{F} = 0 \quad \text{in } \Omega, \tag{1.1}$$

$$[\, \boldsymbol{u} \,] = \boldsymbol{0} \quad \text{on } \Gamma, \tag{1.2}$$

$$[\, \boldsymbol{\sigma n} \,] = \boldsymbol{0} \quad \text{on } \Gamma, \tag{1.3}$$

$$\boldsymbol{u} = \boldsymbol{u_0} \quad \text{on } \partial\Omega, \tag{1.4}$$

where $\boldsymbol{\sigma}$ is the stress tensor, see the definition in Section 2, $\boldsymbol{F} = (F_1, F_2)^T : \Omega \to \mathbb{R}^2$ is the body force, $\boldsymbol{u} = (u_1, u_2)^T$ is the displacement field, Γ is a smooth interface that divides the domain Ω into two parts Ω^+ and Ω^-, $\boldsymbol{n} = (n_1, n_2)^T$ is the unit normal vector of the interface Γ, pointing from the Ω^- phase to the Ω^+ phase, and $\boldsymbol{u_0}$ is a given vector-valued function that represents the displacement on the boundary $\partial\Omega$. Across the interface Γ, the physical parameters such as Young's modulus and Lamé constants have a finite jump discontinuity. For a function v, We use $[v](\boldsymbol{X}) = v^+(\boldsymbol{X}) - v^-(\boldsymbol{X})$ to denote the jump of v across the interface

2000 *Mathematics Subject Classification.* Primary 65N06; Secondary 74B10, 74S30.

Key words and phrases. Immersed finite element method, elasticity system, interface problems, Cartesian grids, jump conditions, level set method.

The first author was partially supported by NSF grants DMS-0201094 and DMS0412654, and by an ARO grant 39676-MA.

Γ at $\boldsymbol{X} \in \Gamma$, where $v^+(\boldsymbol{X})$, for example, is the limiting value of $v(\boldsymbol{x})$ from Ω^+ side defined as

$$v^+(\boldsymbol{X}) = \lim_{\boldsymbol{x} \to \boldsymbol{X},\, \boldsymbol{x} \in \Omega^+} .v(\boldsymbol{X}) \tag{1.5}$$

The jump condition (1.2) means that $\boldsymbol{u}$ is continuous across the interface.

Multi-phase elasticity problems often arise in materials science, see for example, [**14, 19**]. We refer the reader to [**1, 8, 11, 21**] for other applications and related references. However, solving such an elasticity system is often difficult due to the presence of an interface and the discontinuities in the coefficients and the gradient of the solution.

There exist several numerical methods for solving general elasticity systems that do not have interfaces. Among them, finite element methods and boundary integral/element methods appear to be very successful, cf. e.g., [**2, 16, 23**] and the references therein. However, for an elasticity problem with an interface, a body fitted mesh may be needed for those methods in order to get accurate approximation. The grid generation process may become prohibitive expensive for elasticity problems with moving interfaces. Thus a fixed grid, such as a fixed uniform Cartesian grid may be preferred [**9**]. There exist almost none references in the literature about using the finite element methods that use Cartesian grids for solving elasticity problems with interfaces.

In [**21**], a second order immersed interface method was developed for the elasticity system with an interface based on Cartesian grids. However, due to lack of the maximum principle, the stability of the method is still under investigation. The linear solvers for the finite difference equations may not converge satisfactorily. The goal of this paper is to develop an immersed finite element method for solving the two-phase elasticity system. The idea using a Cartesian mesh to solve a single elliptic interface problems via a finite element formulation can be found in [**13, 12, 5**], but it has not been applied to the elasticity systems with interfaces.

This paper is organized as follows: In Section 2, we derive the weak form of the elasticity system. In Section 3, we discuss how to construct the basis functions for non-interface and interface elements. In Section 4, we explain how to use a level set function to represent the interface and to get necessary geometric information. Several numerical examples are given in Section 5.

2. The derivation of the weak form of the elasticity system

In this section, we derive the weak formulation of the elasticity system. We refer the readers to [**3, 22**] for background information, physical implications, and other details. We use a more general boundary condition below that includes (1.4) as a special case

$$\boldsymbol{\sigma n} = \boldsymbol{t}, \quad \text{on } \partial\Omega_1, \tag{2.1}$$

$$\boldsymbol{u} = \boldsymbol{u_0}, \quad \text{on } \partial\Omega_2, \tag{2.2}$$

where $\partial\Omega = \partial\Omega_1 \cup \partial\Omega_2$. Using a finite element formulation, it is more convenient to rewrite the strain $\boldsymbol{\varepsilon}$ and stress $\boldsymbol{\sigma}$ in vector forms below

$$\boldsymbol{\varepsilon} = \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{pmatrix}.$$

If we introduce operator $\boldsymbol{A}$ as follows

$$\boldsymbol{A} = \begin{pmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{pmatrix},$$

then the strain-displacement and stress-strain relations can be rewritten as

$$\boldsymbol{\varepsilon} = \begin{pmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = \boldsymbol{A}\boldsymbol{u}\,, \tag{2.3}$$

$$\boldsymbol{\sigma} = \boldsymbol{D}\,\boldsymbol{\varepsilon}, \tag{2.4}$$

where $\boldsymbol{D}$ is the **elasticity matrix** (or constitutive stress-strain matrix),

$$\boldsymbol{D} = \begin{pmatrix} \lambda+2\mu & \lambda & 0 \\ \lambda & \lambda+2\mu & 0 \\ 0 & 0 & \mu \end{pmatrix}, \tag{2.5}$$

and λ and μ are Lamé constants. Let E be the Young's modulus, and ν be the Poisson's ratio, then we have

$$\mu = \frac{E}{2(1+\nu)},$$

$$\lambda = \frac{E\nu}{1-\nu^2} \quad \text{(plane stress)}, \qquad \lambda = \frac{E\nu}{(1-2\nu)(1+\nu)} \quad \text{(plane strain)}.$$

The equilibrium, constitutive, and strain-displacement equations then become

$$\boldsymbol{A}^T\boldsymbol{\sigma} = -\boldsymbol{F}, \tag{2.6}$$

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\varepsilon}\,, \tag{2.7}$$

$$\boldsymbol{\varepsilon} = \boldsymbol{A}\boldsymbol{u}\,. \tag{2.8}$$

Eliminating $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ gives the "displacement" formulations

$$\boldsymbol{A^T D A}\;\boldsymbol{u} = -\boldsymbol{F}. \tag{2.9}$$

Equation (2.9) will be used to derive the stiffness matrix in the finite element method.

Let us consider the potential energy Π of an elastic body. Π is defined as the sum of the total strain energy (U) and the work potential (WP)

$$\Pi = \text{Strain energy} + \text{Work Potential}.$$

For linear elasticity materials, the strain energy per unit volume in the body is $\frac{1}{2}\,\boldsymbol{\sigma^T\varepsilon}$. For a elastic body, the energy U is given by

$$U = \frac{1}{2}\int_\Omega \boldsymbol{\sigma^T\varepsilon}\;d\Omega = \frac{1}{2}\int_\Omega \boldsymbol{\varepsilon^T D\varepsilon}\;d\Omega.$$

The work potential is given by

$$WP = -\int_\Omega \boldsymbol{u^T F}\;d\Omega - \int_{\partial\Omega_1} \boldsymbol{u^T t}\;ds.$$

The total potential for the general elastic body is

$$\Pi = \frac{1}{2}\int_\Omega \boldsymbol{\varepsilon^T D\varepsilon}\;d\Omega - \int_\Omega \boldsymbol{u^T F}\;d\Omega - \int_{\partial\Omega_1} \boldsymbol{u^T t}\;ds. \tag{2.10}$$

By the principle of minimum potential energy (cf. [**4, 6, 7, 10, 18**]), we obtain the following "weak form" for two-dimensional stress analysis. The solution $\boldsymbol{u}$ is such a vector function that for every $\boldsymbol{v}$ in $H^1(\Omega)$, the following statement holds

$$\int_{\Omega} \boldsymbol{\sigma}^{\boldsymbol{T}} \boldsymbol{v} \; dV = \int_{\Omega} \boldsymbol{v}^{\boldsymbol{T}} \boldsymbol{F} \; d\Omega + \int_{\partial\Omega_1} \boldsymbol{v}^{\boldsymbol{T}} \boldsymbol{t} \; ds. \tag{2.11}$$

Note that $\boldsymbol{u}$ is involved in $\boldsymbol{\sigma}$, and $\boldsymbol{\sigma} = \boldsymbol{D\varepsilon}$.

2.1. The equations for the global and local stiffness matrices and load vectors. We assume that the domain Ω is a rectangle, but the interface Γ can be arbitrary. We use a uniform triangulation (right triangles) regardless of the interface Γ. Therefore, the interface generally is not aligned with the edges of the triangulation. We need to find the basis functions with local support on each right triangle of the partition.

For a given triangle element $\Delta^{(\epsilon)}$, let $\boldsymbol{u}^*$ be the vector of the nodal displacements:

$$\boldsymbol{u}^* = (u_1 \quad v_1 \quad u_2 \quad v_2 \quad u_3 \quad v_3)^T$$

at the three vertices. The displacements at a point inside the triangle $\boldsymbol{u}$ can be determined in terms of the nodal displacements $\boldsymbol{u}^*$ and the basis, or shape functions $\boldsymbol{N}$:

$$\boldsymbol{u} = \boldsymbol{N}\boldsymbol{u}^* .$$

Strains and stresses can also be determined at nodal displacements:

$$\boldsymbol{\varepsilon} = \boldsymbol{A}\boldsymbol{u} = \boldsymbol{A}\boldsymbol{N}\boldsymbol{u}^* = \boldsymbol{B}\boldsymbol{u}^* ,$$

$$\boldsymbol{\sigma} = \boldsymbol{D}\boldsymbol{\varepsilon} = \boldsymbol{D}\boldsymbol{B}\boldsymbol{u}^* ,$$

where $\boldsymbol{B} = \boldsymbol{A}\boldsymbol{N}$ is called the displacement differentiation matrix. So we can obtain the following expressions:

$$\left(\int_{\Omega} \boldsymbol{B}^{\boldsymbol{T}} \boldsymbol{D} \boldsymbol{B} \; dxdy \right) \vec{\boldsymbol{u}}^{\,*} = \int_{\Omega} \boldsymbol{N}^{\boldsymbol{T}} \boldsymbol{F} \; dx\, dy + \int_{\partial\Omega_1} \boldsymbol{N}^{\boldsymbol{T}} \boldsymbol{t} \; ds,$$

where $\vec{\boldsymbol{u}}^{\,*}$ is the vector formed from all the nodal displacement $\boldsymbol{u}^*$. On each element $\Delta^{(e)}$, we have

$$\left(\int_{\Delta^{(e)}} \boldsymbol{B}^{\boldsymbol{T}} \boldsymbol{D} \boldsymbol{B} \; dxdy \right) \boldsymbol{u}^* = \int_{\Delta^{(e)}} \boldsymbol{N}^{\boldsymbol{T}} \boldsymbol{F} \; dx\, dy + \int_{\partial\Omega_1 \cap \Delta^{(e)}} \boldsymbol{N}^{\boldsymbol{T}} \boldsymbol{t} \; ds.$$

We need to know the basis function $\boldsymbol{N}$ on each element.

3. Constructing the basis functions at interface triangles

The triangles in our partition are classified into two categories: the *interface triangle* if the interface divides the triangle into two subsets, and *non-interface triangle* otherwise. For non-interface triangles, we use the standard linear basis functions. For interface triangles, we use an undetermined coefficients method to determine the basis functions by enforcing the natural interface conditions (1.2)-(1.3). Below, we discuss how to construct such piecewise linear basis functions for interface triangular elements.

Let $B = (x_1, y_1)$, $C = (x_2, y_2)$, $A = (x_3, y_3)$ be the three vertices of an interface triangle ΔABC. Let $D = (x_d, y_d)$ and $E = (x_e, y_e)$ be the intersections of the interface and the edges of the triangle, see the diagram in Figure 1 for an illustration. As a common practice, the interface is approximated by the line segment $\overline{DE}$. For convenience, we denote $T^+ \subset \Omega^+$, the region above the interface, and $T^- \subset \Omega^-$, the region below it.

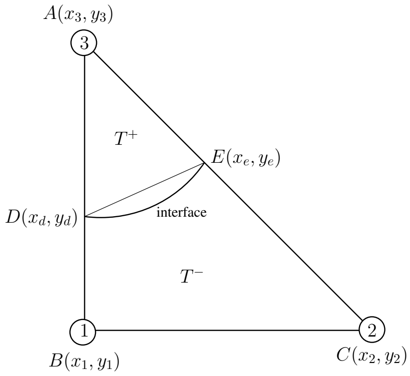


FIGURE 1. A typical interface triangle ΔABC.

Given values u, v at the three vertices of the element T, we construct the following piecewise linear function

$$u(x,y) = \begin{cases} a_1 + a_2 x + a_3 y, & \text{if } (x,y) \in T^+, \\ b_1 + b_2 x + b_3 y, & \text{if } (x,y) \in T^-; \end{cases} \tag{3.1}$$

$$v(x,y) = \begin{cases} c_1 + c_2 x + c_3 y, & \text{if } (x,y) \in T^+, \\ d_1 + d_2 x + d_3 y, & \text{if } (x,y) \in T^-, \end{cases} \tag{3.2}$$

where a_i's, b_i's, c_i's, d_i's, $(i = 1, 2, 3)$ are undetermined coefficients. Let the values of u, v at the vertices A, B, C are u_1, u_2, u_3, v_1, v_2, and v_3, respectively. Plugging in these values to (3.1) and (3.2), we have

$$A(x_3,y_3): \begin{cases} a_1 + a_2 x_3 + a_3 y_3 = u_3, \\ c_1 + c_2 x_3 + c_3 y_3 = v_3; \end{cases} \tag{3.3}$$

$$B(x_1,y_1): \begin{cases} b_1 + b_2 x_1 + b_3 y_1 = u_1, \\ d_1 + d_2 x_1 + d_3 y_1 = v_1; \end{cases} \tag{3.4}$$

$$C(x_2,y_2): \begin{cases} b_1 + b_2 x_2 + b_3 y_2 = u_2, \\ d_1 + d_2 x_2 + d_3 y_2 = v_2; \end{cases} \tag{3.5}$$

$$D(x_d,y_d): \begin{cases} a_1 + a_2 x_d + a_3 y_d - b_1 - b_2 x_d - b_3 y_d = 0, \\ c_1 + c_2 x_d + c_3 y_d - d_1 - d_2 x_d - d_3 y_d = 0; \end{cases} \tag{3.6}$$

$$E(x_e,y_e): \begin{cases} a_1 + a_2 x_e + a_3 y_e - b_1 - b_2 x_e - b_3 y_e = 0, \\ c_1 + c_2 x_e + c_3 y_e - d_1 - d_2 x_e - d_3 y_e = 0. \end{cases} \tag{3.7}$$

The interface condition,

$$[\,\boldsymbol{\sigma n}\,] = \mathbf{0}\,, \qquad \text{on the interface } \Gamma,$$

gives two more constraints

$$\left[(\lambda+2\mu)\frac{\partial u}{\partial x}n_1+\lambda\frac{\partial v}{\partial y}n_1+\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)n_2\right]\Big|_\Gamma=0, \tag{3.8}$$

$$\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)n_1+\lambda\frac{\partial u}{\partial x}n_2+(\lambda+2\mu)\frac{\partial v}{\partial y}n_2\right]\Big|_\Gamma=0, \tag{3.9}$$

where (assuming the plane deformation)

$$\mu=\frac{E}{2(1+\nu)},\qquad\text{and}\qquad\lambda=\frac{E\nu}{(1+\nu)(1-2\nu)}.$$

If we use $\boldsymbol{G}$ to represent the coefficient matrix for the equations (3.3) $\sim$ (3.9), and use $\boldsymbol{Z}$ to represent the vector formed by a_i, b_i, c_i, d_i, $i=1,2,3$, then (3.3) $\sim$ (3.9) can be rewritten as a matrix-vector form:

$$\boldsymbol{G}\boldsymbol{Z} \;=\; \boldsymbol{C}\boldsymbol{u}^* \;, \tag{3.10}$$

where

$$\boldsymbol{C} \;=\; \begin{pmatrix} 0&0&0&0&1&0\\ 0&0&0&0&0&1\\ 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0 \end{pmatrix}, \qquad \boldsymbol{u}^* \;=\; \begin{pmatrix} u_1\\ v_1\\ u_2\\ v_2\\ u_3\\ v_3 \end{pmatrix}.$$

Solving (3.10) gives

$$\boldsymbol{Z} \;=\; \boldsymbol{G}^{-1}\boldsymbol{C}\boldsymbol{u}^* \;. \tag{3.11}$$

In other words, from (3.3) $\sim$ (3.7), and the interface conditions (1.2)-(1.3), we can express a_i, b_i, c_i, d_i, $i=1,2,3$, in terms of u_i, v_i, $i=1,2,3$, or $\{\ \boldsymbol{u}^{\,*}\}$. Therefore, $u(x,y)$ and $v(x,y)$ can be expressed as

$$u(x,y) \;=\; \begin{cases} \left(\overrightarrow{\boldsymbol{a_1}}+x\,\overrightarrow{\boldsymbol{a_2}}+y\,\overrightarrow{\boldsymbol{a_3}}\right)\cdot\ \boldsymbol{u}^*\ , & \text{if } (x,y)\in T^+\\ \left(\overrightarrow{\boldsymbol{b_1}}+x\,\overrightarrow{\boldsymbol{b_2}}+y\,\overrightarrow{\boldsymbol{b_3}}\right)\cdot\ \boldsymbol{u}^*\ , & \text{if } (x,y)\in T^- \end{cases} \tag{3.12}$$

$$v(x,y) \;=\; \begin{cases} \left(\overrightarrow{\boldsymbol{c_1}}+x\,\overrightarrow{\boldsymbol{c_2}}+y\,\overrightarrow{\boldsymbol{c_3}}\right)\cdot\ \boldsymbol{u}^*\ , & \text{if } (x,y)\in T^+\\ \left(\overrightarrow{\boldsymbol{d_1}}+x\,\overrightarrow{\boldsymbol{d_2}}+y\,\overrightarrow{\boldsymbol{d_3}}\right)\cdot\ \boldsymbol{u}^*\ , & \text{if } (x,y)\in T^- \end{cases} \tag{3.13}$$

where $\overrightarrow{\boldsymbol{a_1}} = (a_{11}, a_{12}, a_{13}, a_{14}, a_{15}, a_{16})$ is the first row of matrix $\boldsymbol{G^{-1}C}$ in (3.11), and so forth for $\overrightarrow{\boldsymbol{a_2}}, \cdots, \overrightarrow{\boldsymbol{d_3}}$. If we set

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} N_{11} & N_{12} & N_{13} & N_{14} & N_{15} & N_{16} \\ N_{21} & N_{22} & N_{23} & N_{24} & N_{25} & N_{26} \end{pmatrix} \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \end{pmatrix} \stackrel{def}{=} \boldsymbol{Nu^*} ,$$

where N_{ij}'s $(i = 1, 2,\ j = 1, 2, \cdots, 6)$ are piecewise linear functions, then we have

$$\boldsymbol{N} = \begin{cases} \begin{pmatrix} \overrightarrow{\boldsymbol{a_1}} + x\,\overrightarrow{\boldsymbol{a_2}} + y\,\overrightarrow{\boldsymbol{a_3}} \\ \overrightarrow{\boldsymbol{c_1}} + x\,\overrightarrow{\boldsymbol{c_2}} + y\,\overrightarrow{\boldsymbol{c_3}} \end{pmatrix}, & \text{if } (x, y) \in T^+, \\ \begin{pmatrix} \overrightarrow{\boldsymbol{b_1}} + x\,\overrightarrow{\boldsymbol{b_2}} + y\,\overrightarrow{\boldsymbol{b_3}} \\ \overrightarrow{\boldsymbol{d_1}} + x\,\overrightarrow{\boldsymbol{d_2}} + y\,\overrightarrow{\boldsymbol{d_3}} \end{pmatrix}, & \text{if } (x, y) \in T^-. \end{cases} \tag{3.14}$$

Thus we have determined the shape functions $\boldsymbol{N}$. Figures 2 shows the mesh and contour plots of a pair of the shape functions in an interface triangle using Matlab. Note that the shape functions are only defined on the triangle. But for convenience of plotting in Matlab, they are extended to the entire square by zero. Some wiggles of the plots are due to the discontinuity in the extension, but not from the shape function. Figures 3 shows the mesh and contour plots of a global basis function u on its entire support.

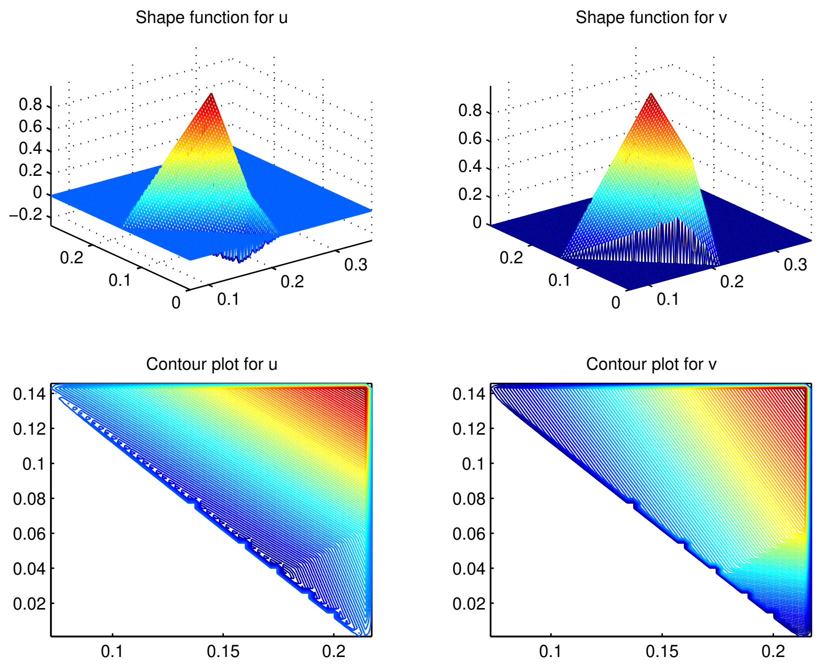
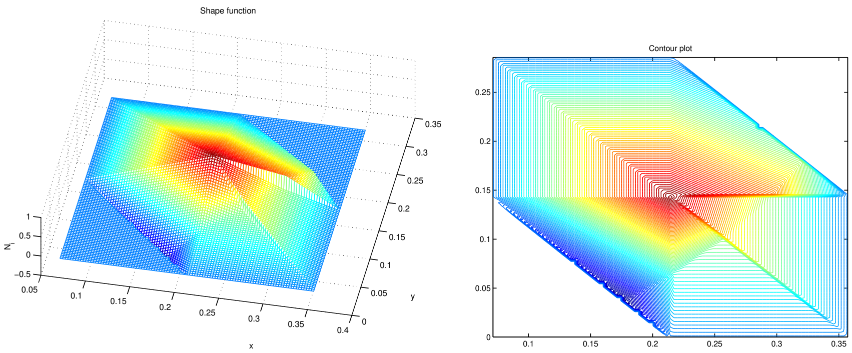


FIGURE 2. The mesh and contour plots of a pair of local basis functions (u, v) over an interface triangle. The parameters are $\lambda^+ = 80$, $\lambda^- = 160$, $\nu^+ = 0.35$, and $\nu^- = 0.15$.



A global basis function Contour plot

FIGURE 3. The mesh and contour plots of a global basis function u on its support. The parameters are $\lambda^+ = 40$, $\lambda^- = 90$, $\nu^+ = 0.35$, and $\nu^- = 0.15$.

Remarks: From the process of constructing the basis functions, we can construct the interpolation function $(u_I(x,y), v_I(x,y))$ that approximates the solution of the elasticity system $(u(x,y), v(x,y)$ with second order accuracy in the infinity norm. The derivation is long and it is almost the same as in [**13**], and therefore is omitted here.

While the global basis functions satisfy the natural interface conditions (1.2)-(1.3) and continuous in each element, they may be discontinuous across the edges of triangles which can be seen in Figure 3. Thus the finite element space is a non-conforming one.

The convergence of the corresponding Galerkin method is at least first order accurate. The optimal convergence rate is difficult to get and it is under investigation.

We then can compute the differentiation strain matrix $\boldsymbol{B}$ for interface elements:

$$\boldsymbol{B}^{+} = \boldsymbol{AN} = \begin{pmatrix} \overrightarrow{\boldsymbol{a_2}} \\ \overrightarrow{\boldsymbol{c_3}} \\ \overrightarrow{\boldsymbol{a_3}} + \overrightarrow{\boldsymbol{c_2}} \end{pmatrix}, \tag{3.15}$$

$$\boldsymbol{B}^{-} = \boldsymbol{AN} = \begin{pmatrix} \overrightarrow{\boldsymbol{b_2}} \\ \overrightarrow{\boldsymbol{d_3}} \\ \overrightarrow{\boldsymbol{b_3}} + \overrightarrow{\boldsymbol{d_2}} \end{pmatrix}. \tag{3.16}$$

The local stiffness matrix then is

$$\begin{aligned} \boldsymbol{K_m} &= \int_{\Delta ABC} \boldsymbol{B^T D B}\ dxdy \\ &= \int_{\Delta ADE} \boldsymbol{B^{+}}^{\boldsymbol{T}} \boldsymbol{D^{+} B^{+}}\ dxdy + \int_{\Delta ABC - \Delta ADE} \boldsymbol{B^{-}}^{\boldsymbol{T}} \boldsymbol{D^{-} B^{-}}\ dxdy \\ &= \boldsymbol{B^{+}}^{\boldsymbol{T}} \boldsymbol{D^{+} B^{+}} \cdot (\text{the area of the triangle } \Delta ADE) \\ &\quad + \boldsymbol{B^{-}}^{\boldsymbol{T}} \boldsymbol{D^{-} B^{-}} \cdot (\text{the area of } (\Delta ABC - \Delta ADE)), \end{aligned} \tag{3.17}$$

which is also a constant and symmetric matrix.

4. Representation of the interface using a level set function

For interface problems, we also need to represent the interface Γ. We need to know how to obtain the the intersections of the interface and edges of the triangles, such as D and E in figure 1. While a Lagrangian frame with marked particles (X_k, Y_k) can be used to represent the interface, we use the same Eulerian coordinates system so that we can deal with more complicated geometries such as multi-connected domain.

We use the zero level set of a Lipschitz continuous function $\varphi(x, y)$ to represent the interface Γ. Usually $\varphi(x, y)$ is chosen as an approximation of the signed distance function, $|\nabla\varphi(x, y)| = 1$ in a neighborhood of the interface Γ. The level set function is defined at grid points $\varphi_{ij} = \varphi(x_i, y_j)$[1]. With the representation of a level set function, it is easy to compute the geometric information that is needed to construct the basis functions. If $\varphi(A)\varphi(B) \leq 0$, then the interface cuts through the line segment $\overline{AB}$. The coordinates of the intersections can be determined from the following theorem.

THEOREM 4.1. *Assume the coordinates of A, B, C, D, E (see Figure 4) are* $A(x_i, y_{j+1})$, $B(x_i, y_j)$, $C(x_{i+1}, y_j)$, $D(x_D, y_D)$, $E(x_E, y_E)$. *Then we have*

$$x_D = x_i, \tag{4.1}$$

$$\begin{aligned} y_D &= y_j + \frac{\varphi_{i,j}\Delta y}{\varphi_{i,j} - \varphi_{i,j+1}} + O((\Delta y)^2) \\ &= y_{j+1} - \frac{\varphi_{i,j+1}\Delta y}{\varphi_{i,j+1} - \varphi_{i,j}} + O((\Delta y)^2), \end{aligned} \tag{4.2}$$

$$\begin{aligned} x_E &= x_i + \frac{\Delta x\varphi_{i,j+1}}{-\Delta x\varphi_x(E) + \Delta y\varphi_y(E)} + O(\Delta x\Delta y) \\ &= x_{i+1} + \frac{\Delta x\varphi_{i+1,j}}{-\Delta x\varphi_x(E) + \Delta y\varphi_y(E)} + O(\Delta x\Delta y), \end{aligned} \tag{4.3}$$

$$y_E = y_j + \frac{\Delta y}{\Delta x}(x_{i+1} - x_E), \tag{4.4}$$

where $\varphi_{i,j+1} = \varphi(x_i, y_{j+1})$, $\varphi_{i,j} = \varphi(x_i, y_j)$, $\varphi_{i+1,j} = \varphi(x_{i+1}, y_j)$, *and*

$$\varphi_x(E) = \frac{\varphi_{i,j} - \varphi_{i+1,j}}{x_i - x_{i+1}} + O(\Delta x\Delta y), \tag{4.5}$$

$$\varphi_y(E) = \frac{\varphi_{i,j+1} - \varphi_{i,j}}{y_{j+1} - y_j} + O(\Delta x\Delta y). \tag{4.6}$$

The above theorem gives all the information that is needed to construct the basis function on an interface triangle.

The proof of the theorem is straightforward using the Taylor expansion of $\varphi(x, y)$ at the intersections D and E.

Using the level set function, the unit normal is simply $\nabla\varphi/|\nabla\varphi|$. The other geometric information is also easy to compute using the level set function, see for example, [**15, 17**].

[1]For example, an ellipse $x^2/a^2 + y^2/b^2 = 1$ can be represented by $\varphi_{ij} = x_i^2/a^2 + y_j^2/b^2 - 1$.

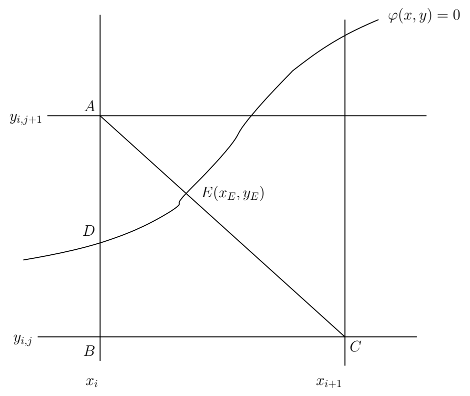


FIGURE 4. The geometric information of an interface triangle ΔABC and the level set function representation.

5. Numerical examples

In this section we present three numerical examples. Without loss of generality, the domain Ω is a rectangle $[-1,1]\times[-1,1]$, and the interface Γ is $x^2+y^2=\frac{1}{4}$. Since the analytic solution is not available, we compare the computed solutions obtained from the immersed finite element method with the solutions obtained from the immersed finite difference method developed in [**20, 21**] for example 5.1 to validate the IFEM method for the elasticity systems with interfaces. The immersed finite difference method developed in [**20, 21**] have been tested against the exact solution and well validated.

EXAMPLE 5.1. In this example, the differential equations, interface and boundary conditions are exactly the same as that in Example 2.5.3 in [**20**]. The body force is $\boldsymbol{F} = \mathbf{0}$. The Dirichlet boundary conditions are given by

$$\begin{aligned} u_0(x,y) &= -\frac{1}{10}\left(x^2+y^2\right)^2 + \frac{1}{100}\ln\left(2\sqrt{x^2+y^2}\right) - \frac{39}{160}, \\ v_0(x,y) &= \ln\left(1+x^2+3\,y^2\right) + \sin(xy) - 4\,x^2 - 4\,y^2 + 1. \end{aligned}$$

The parameters are

$$\begin{aligned} &\nu^- = 0.22, \quad \mu^- = 200, \quad E^- = 488, \\ &\nu^+ = 0.11, \quad \mu^+ = 60, \quad\ \ E^+ = 133.20. \end{aligned}$$

Figure 5 and Figure 6 are the mesh and contour plots of the solution (u,v) obtained by the IFEM method described in this paper using a 56 by 56 mesh. They are almost identical to the results obtained from the finite difference method, see Figure 2.8 and Figure 2.9 in [**20**]. This has also been confirmed by quantitative analysis. The finite difference method is slightly slower than the finite element method.

EXAMPLE 5.2. The set-up of this example is the same as in EXAMPLE 5.1, but with different elasticity constants

$$\begin{aligned} &\nu^- = 0.02, \quad \mu^- = 0.4902, \quad\ \ E^- = 1, \\ &\nu^+ = 0.49, \quad \mu^+ = 167.7852, \quad E^+ = 500, \end{aligned}$$

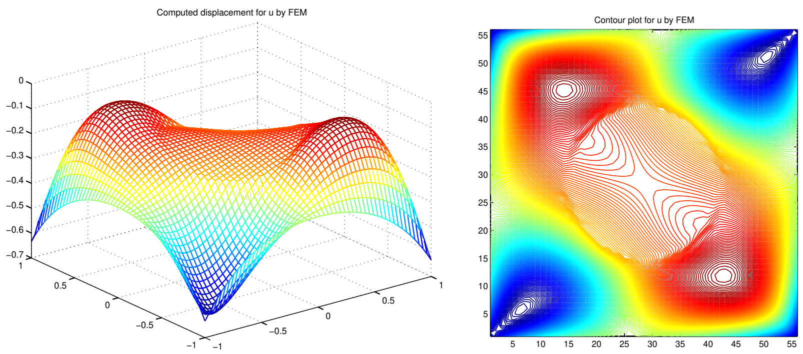


FIGURE 5. The mesh and contour plots of the computed solution u of EXAMPLE 5.1 obtained by the immersed finite element method developed in this paper.

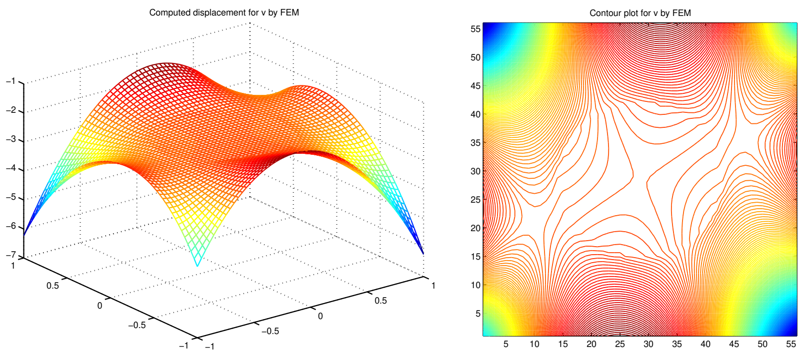


FIGURE 6. The mesh and contour plots of the computed solution v of EXAMPLE 5.1 obtained by the immersed finite element method developed in this paper.

and different boundary conditions

$$\begin{aligned} u_0(x,y) &= \cos\left((x+1)^2\right) + y + 1 + \sin\left((x+1)(y+1)\right), \\ v_0(x,y) &= (x+1)(y+1) + \sin\left(2\,(x+1)(y+1)\right). \end{aligned}$$

The jump ratios in the parameters are large than that in the previous example meaning a more difficult problem. Figure (7) and Figure (8) are the mesh and contour plots for the displacements.

In this example the finite difference method takes much longer time to converge than the finite element method does. This is because the linear system of equations obtained from the finite element method is symmetric and has better condition number compared with that obtained from the finite difference approach. This is one of motivations to develop the finite element method for elasticity problems with interfaces using a fixed Cartesian grid.

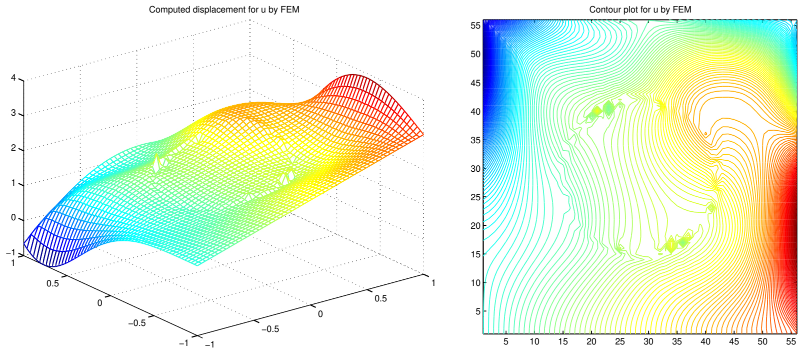


FIGURE 7. The mesh and contour plots of the computed solution u of EXAMPLE 5.2 obtained by the immersed finite element method developed in this paper.

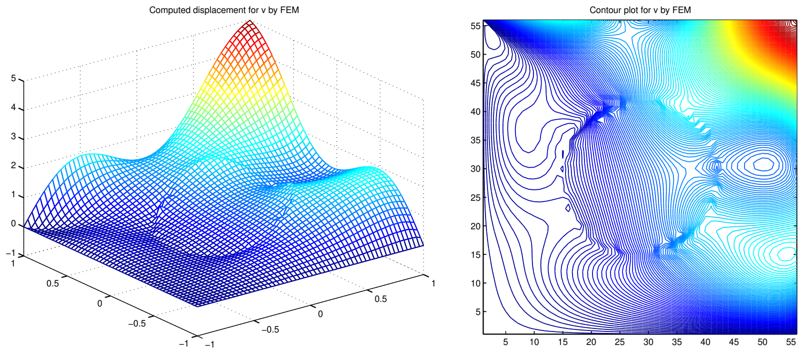


FIGURE 8. The mesh and contour plots of the computed solution v of EXAMPLE 5.2 obtained by the immersed finite element method developed in this paper.

EXAMPLE 5.3. In this example, the boundary condition is:

$$
\begin{aligned}
u_0(x,y) &= \ln\left(1.0+(x+1)^2+(y+1)^2\right)+\sin\left((x+1)\,(y+1)\right), \\
v_0(x,y) &= (x+1)\,(y+1)+\sin\left(2.0\;(x+1)\,(y+1)\right).
\end{aligned}
$$

The body force $\boldsymbol{F}$ is nonzero and determined by (1.1), that is $\boldsymbol{F} = -\nabla\cdot\boldsymbol{\sigma}$ with $\boldsymbol{\sigma}$ determined by $\boldsymbol{u}_0 = (u_0(x,y), v_0(x,y))$, and and the parameters from Ω^+ side. The parameters for this example are $E^+ = 150$, $E^- = 10$, $\nu^+ = .2$ and $\nu^- = .24$. This example is close to real applications with given force and boundary conditions. Note that $\boldsymbol{u}_0 = (u_0(x,y), v_0(x,y))$ is not the solution since the natural interface conditions (1.2)-(1.3) are not satisfied unless the parameters are continuous. Figure 7 and Figure 8 are the mesh and contour plots of the displacements obtained from the finite element method developed in this paper.

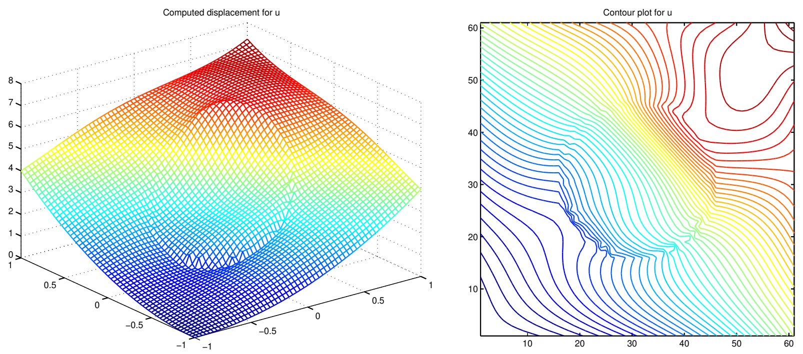


FIGURE 9. The mesh and contour plots of the computed solution u of EXAMPLE 5.3 obtained by the immersed finite element method developed in this paper.

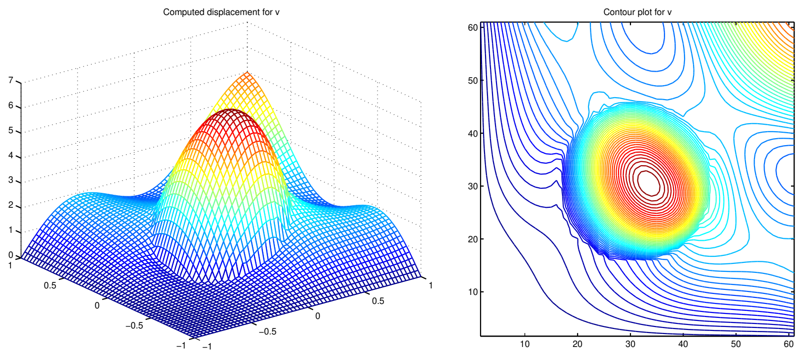


FIGURE 10. The mesh and contour plots of the computed solution v of EXAMPLE 5.3 obtained by the immersed finite element method developed in this paper.

6. Conclusions

In this paper, we have developed an immersed finite element method (IFEM) for a linear elasticity system of equations with interfaces and discontinuous parameters. The method is based on a fixed Cartesian grid and modifications of the basis functions to enforce the natural interface conditions at interface triangles. The resulting finite element space is a non-conforming one but has the same degree of the freedom as the standard finite element method apply to standard linear elasticity systems. Compared with the immersed finite difference method, the linear system derived from the IFEM is symmetric and had better condition number. We can construct an interpolation function that can approximate the solution to the elasticity system to second order accuracy in the infinity norm. It is not difficult to prove that the IFEM for the elasticity system is at least first order convergent. However, the optimal convergence rate may be difficult to obtain due to the nature of the problem and non-conforming IFE space. Such research is still under investigation.

References

1. R. J. Asaro and W. A. Tiller, *Surface morphology development during stress corrosion cracking: Part I: via surface diffusion*, Metall. Trans. **3** (1972), 1789–1796.
2. I. Babuška and T. Strouboulis, *The finite element method and its reliability*, Oxford University Press, 2001.
3. K. J. Bathe, *Numerical methods in finite element analysis*, 3rd ed., Prentice Hall Inc., Englewood Cliffs, NJ, 1996.
4. Tirupathi R. Chandrupatla and Ashok D. Belegundu, *Introduction to finite elements in engineering*, Prentice Hall, Upper Saddle River, N.J., 1997.
5. R. Ewing, Z. Li, T. Lin, and Y. Lin, *The immersed finite volume element method for the elliptic interface problems*, Mathematics and Computers in Simulation **50** (1999), 63–76.
6. Y. C. Fung, *Foundations of solid mechanics*, Prentice Hall, Englewood Cliffs, N. J., 1965.
7. Y. C. Fung and P. Tong, *Classical and computational solid mechanics*, World Scientific, Singapore, 2001.
8. M. A. Grinfeld, *The stress driven rearrangement instability in ellastic crystals: mathematical models and physical manifestations*, Nonlinear Sci. **3** (1993), 35–83.
9. T.Y. Hou, *Numerical solutions to free boundary problems*, Acta Numerica (1995), 335–415.
10. Kenneth H. Huebner, Earl A. Thornton, and Ted G. Byrom, *The finite element method for engineers*, third edition ed., A Wiley-Interscience Publication, John Wiley & Sons Inc., New York, 1995.
11. H.-J. Jou, H. Leo, and J. S. Lowengrub, *Microstructrual evolution in inhomogeneous elastic media*, J. Comp. Phy. **131** (1997), no. 1, 109–148.
12. Z. Li, T. Lin, Y. Lin, and R.C. Rogers, *Error estimates of an immersed finite element method for interace problems*, Numerical PDEs **20** (2004), no. 3, 338–367.
13. Zhilin Li, Tao Lin, and Xiaohui Wu, *New Cartesian grid methods for interface problems using the finite element formulation*, Numer. Math. **96** (2003), no. 1, 61–98. MR 2 018 791
14. T. Mura, *Micromechanics of defects in solids*, Martinus Nijhoff Publishers, 1982.
15. S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*, Springer, New York, 2002.
16. C. Pozrikidis, *Boundary element methods*, CRC Press, 2002.
17. J. A. Sethian, *Level set methods and fast marching methods*, Cambridge University Press, 2nd edition, 1999.
18. Jeffrey M. Steele, *Applied finite element modeling: Practical problem solving for engineers*, Mechanical engineering, 66, New York: M. Dekker, Inc, 1989.
19. A. P. Sutton and R. W. Balluffi, *Interfaces in crystalline materials*, Oxford University Press, 1996.
20. Xingzhou Yang, *Immersed interface method for elasticity problems with interfaces*, Ph.D. thesis, North Carolina State University, 2004.
21. Xingzhou Yang, Bo Li, and Zhilin Li, *The immersed interface method for elasticity problems with interfaces*, Dyn. Contin. Discrete Impuls. Syst. Ser. A Math. Anal. **10** (2003), no. 5, 783–808, Progress in partial differential equations (Pullman, WA, 2002). MR 2 013 653
22. O. C. Zienkiewicz and R. L. Taylor, *The finite element method. Vol. 1, the basis*, fifth ed., Butterworth-Heinemann, Oxford, 2000. MR **2002m:**65001a
23. ———, *The finite element method. Vol. 2, solid mechanics*, Butterworth-Heinemann, 2001.

Center for Research in Scientific Computation & Department of Mathematics, North Carolina State University, Raleigh, NC 27695, USA.

E-mail address: `zhilin@math.ncsu.edu`

310 Richardson Building, Center for Computational Science, Tulane University, New Orleans, LA 70118

E-mail address: `xyang4@tulane.edu`