A FINITE ELEMENT METHOD FOR ELASTICITY INTERFACE PROBLEMS WITH LOCALLY MODIFIED TRIANGULATIONS

HUI XIE, ZHILIN LI, AND ZHONGHUA QIAO

Abstract. A finite element method for elasticity systems with discontinuities in the coefficients and the flux across an arbitrary interface is proposed in this paper. The method is based on a Cartesian mesh with local modifications to the mesh. The total degrees of the freedom of the finite element method remains the same as that of the Cartesian mesh. The local modifications lead to a quasi-uniform body-fitted mesh from the original Cartesian mesh. The standard finite element theory and implementation are applicable. Numerical examples that involve discontinuous material coefficients and non-homogeneous jump in the flux across the interface demonstrate the efficiency of the proposed method.

Key Words. elasticity interface problem, body-fitted mesh, Cartesian mesh, discontinuous coefficient, locally modified triangulation, finite element method, jump conditions

1. Introduction

In this paper, we propose a finite element method for plane elasticity problems with interfaces in which the physical parameters and solutions may be discontinuous across an arbitrary interface. Elasticity interface problems have wide applications in continuum mechanics, particularly for problems that involve stresses and strains, see for example, [4, 13, 20].

We first introduce the problem of our interest. Let $\mathbf{x} = (x, y)$ be a point in space and $\mathbf{u} = (u_1(x, y), u_2(x, y))$ be the displacement of a plate which is composed of different materials. The relation between strains and displacements of the plate is given by

$$(1) \qquad \varepsilon_{11} = \frac{\partial u_1}{\partial x}, \quad \varepsilon_{22} = \frac{\partial u_2}{\partial y}, \quad \varepsilon_{12} = \varepsilon_{21} = \frac{1}{2} \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right).$$

Assuming that the material is linearly elastic and isotropic; and that the displacements are small, we have the following relation between stresses and strains, or the constitutive relation from the Hooke's law,

$$(2) \qquad \sigma_{ij} = \lambda \left(\nabla \cdot \mathbf{u} \right) \delta_{ij} + 2\mu \varepsilon_{ij}(\mathbf{u}), \quad i, j = 1, 2,$$

where λ and μ are the Lamé coefficients, and

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases}$$

$$\nabla \cdot \mathbf{u} = \frac{\partial u_1}{\partial x} + \frac{\partial u_2}{\partial y}.$$

Let $\sigma = (\sigma_{ij})$ be the stress tensor, $\mathbf{f}(\mathbf{x}) = (f_1, f_2)$ be the applied body forces, then the stress tensor satisfies the following partial differential equations,

(3) $-\nabla \cdot \sigma = \mathbf{f},$

i.e.,

(4)
$$\begin{cases} -\dfrac{\partial \sigma_{11}}{\partial x} - \dfrac{\partial \sigma_{12}}{\partial y} = f_1, \\ -\dfrac{\partial \sigma_{21}}{\partial x} - \dfrac{\partial \sigma_{22}}{\partial y} = f_2. \end{cases}$$

From (2)-(4), we can re-write the above system as the system of plane elasticity equations of the following,

(5)
$$-\left\{ (\lambda + 2\mu)\frac{\partial^2 u_1}{\partial x^2} + (\lambda + \mu)\frac{\partial^2 u_2}{\partial x \partial y} + \mu\frac{\partial^2 u_1}{\partial y^2} \right\} = f_1,$$
$$-\left\{ (\lambda + 2\mu)\frac{\partial^2 u_2}{\partial y^2} + (\lambda + \mu)\frac{\partial^2 u_1}{\partial x \partial y} + \mu\frac{\partial^2 u_2}{\partial x^2} \right\} = f_2.$$

In the vector form, it is

(6) $-\mu \triangle \mathbf{u} - (\lambda + \mu)\nabla \nabla \cdot \mathbf{u} = \mathbf{f}.$

Note that, in practice, it is common to use the Young's modulus E and Poisson's ratio ν instead of the Lamé coefficients λ and μ in the expression (2). The relations between λ and μ, and E and ν, are given by

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

(8) $\lambda = \dfrac{\nu E}{(1 + \nu)(1 - 2\nu)}$ (plane strain), $\lambda = \dfrac{\nu E}{1 - \nu^2}$ (plane stress).

We want to obtain the numerical solution of the elasticity system that has an interface Γ in the solution domain. Across the interface Γ, the material coefficients may have finite jumps; so does the flux $\sigma\mathbf{n}$, see Fig. 1 for an illustration. Now the problem can be written as follows:

(9) $-\nabla \cdot \sigma \;\; = \;\; \mathbf{f} \quad \text{in } \Omega^+ \cup \Omega^-$

(10) $[\mathbf{u}]_\Gamma \;\; = \;\; 0,$

(11) $[\sigma\mathbf{n}]_\Gamma \;\; = \;\; \mathbf{q},$

(12) $\mathbf{u}|_{\partial\Omega} \;\; = \;\; \mathbf{u}_0,$

where $\mathbf{f} = (f_1, f_2)$, $\mathbf{q} = (q_1, q_2)$, $\mathbf{u}_0 = (u_{0_1}, u_{0_2})$ are known vector functions, and $\Gamma \in C^2$ is a closed interface between the subdomains Ω^+ and Ω^-. The jump $[\cdot]_\Gamma$ is defined as the difference of the limiting values from the outside of the interface to the inside, and $\mathbf{n}$ is the unit normal direction of the interface Γ pointing outward. We refer the reader to [15, 16] for more information of the elasticity problems.

It is always challenging to solve the interface problems. Several different approaches have been developed based on different formulations. A common and simple approach is to use a body-fitted mesh and a finite element method. This
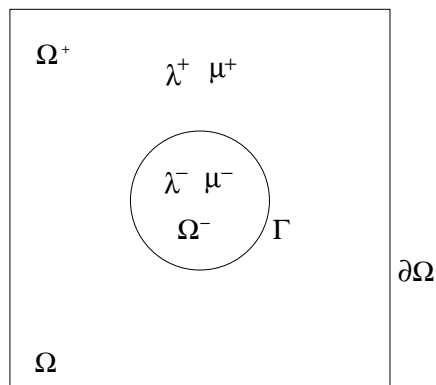
FIGURE 1. A rectangular domain Ω with an immersed interface Γ.

approach may be prohibitively expensive if the interface Γ is moving with time. Many Cartesian grid methods have been developed recently to avoid the cost in the grid generation. Among them, the finite difference method [18, 19], which is second order accurate but the linear system of equations is not symmetric and ill-conditioned for the case when ν is close to $1/2$; the first order finite element method [12] with modified basis functions, which requires homogeneous flux jump condition; and the new second order conforming finite element methods [5, 6, 7], which allow both the solution and the flux to have non-homogeneous jumps. But these methods are somewhat complicated.

This paper is motivated by the locally modified triangulations proposed in [2, 17] for Poisson equations on irregular domains. The main contribution of this paper is that we use such meshes to solve the elasticity problem with interfaces using the Galerkin finite element method. We want to take advantage of Cartesian meshes and the finite element method using body-fitted meshes. The Cartesian meshes have several benefits over non-structured meshes. In the literature several finite element methods have been proposed using Cartesian meshes for elasticity interface problems. The non-conforming finite element method proposed in [11] is simple and enforces the homogeneous jump condition but it is not fully second order accurate because the basis functions are non-conforming. The conforming finite element method proposed in [6, 11] is second order accurate but it is not easy to implement as the basis functions have wider support in the neighborhood of the interface. The finite element method proposed in this paper is an alternative approach based on body-fitted meshes with modification to the Cartesian mesh only in the neighborhood of the interface. Thus, the coefficient matrix is altered only in the elements that are near the interface. This nature is fully taken into account in developing an efficient algebraic solver in [9] using sparse subspace iterative method. In [17], the finite element method using the locally modified mesh was also compared with the finite element method based on a locally enriched mesh in which the intersections between the grid lines and the interface are added as additional nodal points. Both methods can lead to second order accurate solution for elliptic interface problems. But the efficient iterative solver proposed in [9] in general can not be applied for the locally enriched mesh because the structure of the matrix has been changed. Other numerical simulations based on the locally modified mesh can be found in [8, 10].

The paper is organized as follows. In the next section, we discuss the weak formulation. In Section 3, we discuss the most important part of our method: the mesh generation. Numerical results are presented in Section 4. We conclude in Section 5.

2. The weak formulation

First we apply the standard Galerkin finite element method [3, 21] to a homogeneous Dirichlet boundary value problem to derive the weak form. We multiply both sides of the equation (9) by a test vector function $\mathbf{v} = (v_1, v_2)$, $v_1, v_2 \in H_0^1(\Omega)$ and integrate over the domain Ω^+ and Ω^- to obtain

$$
(13) \quad
\begin{aligned}
&-\iint_{\Omega^+} \left\{ \left(\frac{\partial \sigma_{11}}{\partial x} + \frac{\partial \sigma_{12}}{\partial y} \right) v_1 + \left(\frac{\partial \sigma_{21}}{\partial x} + \frac{\partial \sigma_{22}}{\partial y} \right) v_2 \right\} dxdy \\
&= \iint_{\Omega^+} (f_1 v_1 + f_2 v_2) dxdy.
\end{aligned}
$$

$$
(14) \quad
\begin{aligned}
&-\iint_{\Omega^-} \left\{ \left(\frac{\partial \sigma_{11}}{\partial x} + \frac{\partial \sigma_{12}}{\partial y} \right) v_1 + \left(\frac{\partial \sigma_{21}}{\partial x} + \frac{\partial \sigma_{22}}{\partial y} \right) v_2 \right\} dxdy \\
&= \iint_{\Omega^-} (f_1 v_1 + f_2 v_2) dxdy.
\end{aligned}
$$

Then integrating each term by parts and rearranging on the above equations and using $v_1|_{\partial\Omega} = 0, v_2|_{\partial\Omega} = 0$, we get

$$
(15) \quad
\begin{aligned}
&\int_\Gamma \left\{ (\sigma_{11} n_1 + \sigma_{12} n_2) v_1 + (\sigma_{12} n_1 + \sigma_{22} n_2) v_2 \right\} ds \\
&+ \iint_{\Omega^+} \left\{ \sigma_{11} \frac{\partial v_1}{\partial x} + \sigma_{12} \left(\frac{\partial v_2}{\partial x} + \frac{\partial v_1}{\partial y} \right) + \sigma_{22} \frac{\partial v_2}{\partial y} \right\} dxdy \\
&= \iint_{\Omega^+} (f_1 v_1 + f_2 v_2) dxdy,
\end{aligned}
$$

and

$$
(16) \quad
\begin{aligned}
&-\int_\Gamma \left\{ (\sigma_{11} n_1 + \sigma_{12} n_2) v_1 + (\sigma_{12} n_1 + \sigma_{22} n_2) v_2 \right\} ds \\
&+ \iint_{\Omega^-} \left\{ \sigma_{11} \frac{\partial v_1}{\partial x} + \sigma_{12} \left(\frac{\partial v_2}{\partial x} + \frac{\partial v_1}{\partial y} \right) + \sigma_{22} \frac{\partial v_2}{\partial y} \right\} dxdy \\
&= \iint_{\Omega^-} (f_1 v_1 + f_2 v_2) dxdy,
\end{aligned}
$$

where $\mathbf{n} = (n_1, n_2)$ is the unit normal direction of the interface Γ pointing outward. Adding (15) to (16), we obtain

$$
(17)
$$
$$
\int_\Gamma \left\{ q_1 v_1 + q_2 v_2 \right\} ds + \iint_\Omega \left\{ \sigma_{11} \frac{\partial v_1}{\partial x} + \sigma_{12} \left(\frac{\partial v_2}{\partial x} + \frac{\partial v_1}{\partial y} \right) + \sigma_{22} \frac{\partial v_2}{\partial y} \right\} dxdy
$$
$$
= \iint_\Omega (f_1 v_1 + f_2 v_2) dxdy
$$

using the fact that $[\mathbf{u}]_\Gamma = 0$ and $[\sigma \mathbf{n}]_\Gamma = \mathbf{q}$. Thus we have arrived at the weak form:

$$
(18) \quad
\begin{aligned}
&\iint_\Omega \left\{ \sigma_{11} \frac{\partial v_1}{\partial x} + \sigma_{12} \left(\frac{\partial v_2}{\partial x} + \frac{\partial v_1}{\partial y} \right) + \sigma_{22} \frac{\partial v_2}{\partial y} \right\} dxdy \\
&= \iint_\Omega (f_1 v_1 + f_2 v_2) dxdy - \int_\Gamma \left\{ q_1 v_1 + q_2 v_2 \right\} ds.
\end{aligned}
$$

The problem then is to find $\mathbf{u} \in H^1(\Omega) \times H^1(\Omega)$ such that the weak form above is true for all $\mathbf{v} \in H_0^1(\Omega) \times H_0^1(\Omega)$.

3. The mesh generation processes

The generation of a body-fitted triangulation from a Cartesian grid is an essential part of our algorithm. For completeness, we excerpt the mesh generation processes from [17].

The idea of the locally modified mesh is to perturb the triangulation while keeping the number of nodal points, or the degrees of the freedom, unchanged. The procedure is described in some detail below:

Step 1 Generate a rectangular grid. We denote the grid points (x_i, y_j), $0 \le i \le M$, $0 \le j \le N$. For simplicity, we assume the step sizes are the same in both x and y direction. We denote the step size by h.

Step 2 Find the intersections of rectangular grid lines and the interface Γ. We call them as interface points below. If the coordinates of an interface point lie in $(x_i - h/2, x_i + h/2] \times (y_j - h/2, y_j + h/2]$, then we call such a grid point (x_i, y_j) an irregular point. Otherwise a grid point is called a regular one.

We move each irregular point to a new location on the interface as a new nodal point (replacing the original one). If there is only one interface point associated with the irregular point, then we simply replace the irregular point with the intersection as the new nodal point. If there are more than one interface points associated with the grid point (x_i, y_j), we choose the nearest intersection to the grid point as the new nodal point; see Fig. 2 for an illustration.
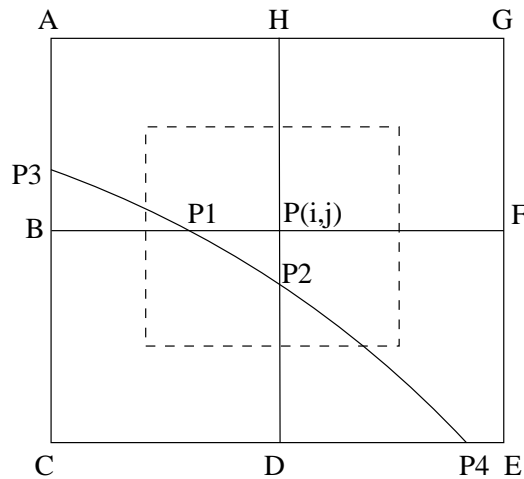

FIGURE 2. A typical irregular grid point P with coordinates (x_i, y_j) and the geometry. There are two interface points P1, P2 lie in the inner domain $(x_i - h/2, x_i + h/2] \times (y_j - h/2, y_j + h/2]$. So P is an irregular point. We will move P to its nearest interface point. Here we move P to $P2$ since $|PP2| < |PP1|$, where $|PP2|$ is the distance between P and $P2$ and so on.

In Fig. 2, the irregular grid point is P, the interface points are $P1$ and $P2$. Since P is closer to $P2$ than $P1$, we move P to $P2$ as a new nodal point. In Fig. 2, points C, D, F, H, G, and likely point A (we assume it is

for simplicity of the discussion), are regular grid points which means that there is no need to move them. We move the point B to $P3$, and E to $P4$, then we have four quadrilaterals around point $P2$; see Fig. 3.
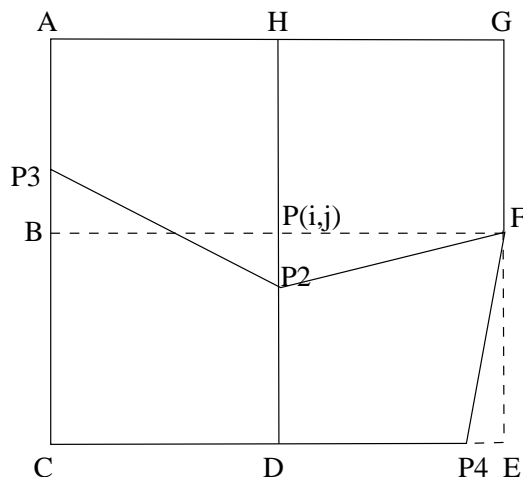

FIGURE 3. The geometry near a perturbed nodal point $P2$. We assume points P, B, E are irregular points and they are replaced by the new nodal points P2, P3, P4 respectively. After these perturbations, there are four quadrilaterals around nodal point $P2$ which may not be rectangles.

Step 3 Form the triangulation. The emphasis is how to generate triangles at irregular grid points. Now we discuss how to generate the triangles after we have moved irregular grid points to get new nodal points. We use Fig. 3 to illustrate the idea. Each of the quadrilaterals of the perturbed grid is divided into two triangles along one of its diagonals. Let K be the number of irregular points (perturbed nodal points) in a quadrilateral. We divide the quadrilateral to triangles according to the following rules:
  – If $K = 0$, then the quadrilateral is rectangular. We form the triangles by connecting two vertices from the lower left corner to the upper right corner.
  – If $K = 2$ and the two irregular points are the opposite corners, then we connect the irregular points; see, for example, $P2$ and $P4$ in Fig. 3.
  – If $K = 1$, or $K = 2$ and the two irregular points are not the opposite corners, then we connect the diagonal which gives better mesh quality. In Fig. 3, we would connect $P2$ and C, $P3$ and H, H and F to form the triangulation; see Fig. 4. Usually the criterion is to connect the shorter diagonal.
  – The cases $K = 3$ and $K = 4$ are special cases that are rarely happens if the interface Γ is smooth and the mesh is fine enough. We omit this discussion here but refer the readers to [2, 14] for the detail.

It has been proved in [2] that for $\Gamma \in C^2$ this mesh generation algorithm leads to a quasi-uniform triangulation, that is, for each triangle the ratio between the length of the longest side and the length of the shortest side is bounded. Furthermore, the accuracy of the approximation of the interface Γ is $O(h^2)$.
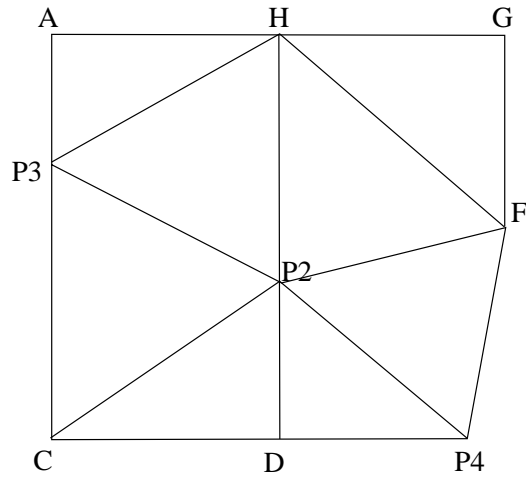
FIGURE 4. The triangulation near P2.

For the purpose of the illustration, in Fig. 5 we show a locally modified mesh. In the next section, we will present the errors of the finite element solutions using the locally modified mesh.
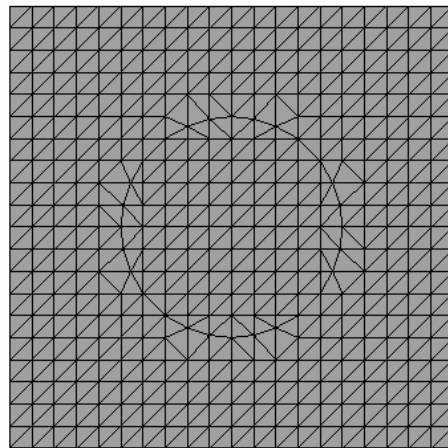

FIGURE 5. A locally modified mesh.

4. Numerical results

We present numerical experiments for the model problem (9)-(12) using the proposed locally modified mesh method. We use the standard piecewise linear Galerkin finite element method. Finite Element Program Generator (FEPG), see [1], a free trial FE software downloaded from the Internet, is used to generate the stiffness matrix and the load vector. We use the incomplete LU decomposition (ILU) along with the conjugate gradient method to solve the resulting linear system of equations.

Most of computations are done on an Intel Core 2 Duo 2.0GHz processor with 2GB of RAM notebook computer. For a 320×320 mesh, it takes about 0.2 seconds to generate the mesh. Most of simulations are done within seconds.

We present some examples (Examples 1, 2) with known exact solutions so that we can validate the computer codes and check the convergence rate. The interface Γ is the circle $x^2 + y^2 = 1/4$ within the computational domain $\Omega = (-1, 1) \times (-1, 1)$. For a 320×320 mesh, it takes 53 seconds to solve the resulting linear system using the ILU. In Example 3, we consider a more realistic problem with a non-convex general interface. In this example, we do not have an analytic solution. It takes about 3 minutes to solve the resulting linear system using the ILU.

Let $\tilde{u}$ be the finite element solution obtained from our method and

$$\tilde{u}(x, y) = \sum_{ij} U_{ij} \varphi_{ij}(x, y)$$

where φ_{ij} is the piecewise linear basis function centered at point (x_i, y_j). We define e_∞, e_1 as the errors in L^∞ and H^1 norms, respectively, thus

$$(19) \qquad\qquad e_\infty = \max_{i,j} |u(x_i, y_j) - U_{ij}|,$$

$$(20) \qquad\qquad e_1 = \sqrt{\int_\Omega \left\{ (u - \tilde{u})^2 + (u_x - \tilde{u}_x)^2 + (u_y - \tilde{u}_y)^2 \right\} d\Omega}.$$

We also scale e_∞, e_1 as follows to get the relative errors,

$$(21) \qquad\qquad r_\infty = \frac{e_\infty}{\max\limits_{i,j} |u(x_i, y_j)|},$$

$$(22) \qquad\qquad r_1 = \frac{e_1}{\max\limits_{i,j} |u(x_i, y_j)|}.$$

We will use this scaled error measurement only for Example 2.

Example 1. *The first example is taken from [5]. The parameters are $\lambda^- = \mu^- = 1$, and $\lambda^+ = \mu^+ = 100$. The body force term $\mathbf{f} = (f_1, f_2)$ and the Dirichlet boundary condition $\mathbf{u}_0 = (u_{0_1}, u_{0_2})$ are given from the exact solution $\mathbf{u} = (u_1, u_2)$:*

$$u_1 = u_2 = \begin{cases} r^2, & r \leq R; \\ \dfrac{r^2}{100} + (1 - \dfrac{1}{100})R^2, & otherwise, \end{cases}$$

where $R = \frac{1}{2}$ and $r = \sqrt{x^2 + y^2}$.

Note that the solution $\mathbf{u}$ is continuous across the interface $r = 1/2$ and $[\sigma \mathbf{n}]|_\Gamma = 0$, i.e., the exact solution satisfies the homogeneous jump conditions.

In Table 1 we show the result of a grid refinement analysis of the finite element solutions using the locally modified meshes. In the table, the first column N is the number of intervals in x and y directions; the second column is the error in the maximum norm. The third column is the ratio of the two consecutive errors. The ratio approaches number four for quadratic convergence and number two for linear convergence. The fourth column is the error in the H^1 norm. The other columns in this table and other tables have the similar meanings. The finite element methods are both roughly second order accurate in the maximum norm and first order accurate in the H^1 norm as expected.

TABLE 1. A grid refinement analysis in the maximum and H^1 norms of the finite element methods using the locally modified meshes for Example 1.

N	e_∞	ratio	e_1	ratio
20	2.4271E-03		1.1957E-02	
40	8.8263E-04	2.7498	4.4212E-03	2.7045
80	2.5543E-04	3.4555	1.7337E-03	2.5502
160	5.7255E-05	4.4613	6.3617E-04	2.7252
320	1.4157E-05	4.0443	2.2759E-04	2.7952

Example 2. *We set $\lambda^- = \mu^- = 1$ and $\lambda^+ = \mu^+ = b$ for some $b > 0$. And the body force term $\mathbf{f} = (f_1, f_2)$ and the Dirichlet boundary condition $\mathbf{u}_0 = (u_{0_1}, u_{0_2})$ are derived from the exact solution $\mathbf{u} = (u_1, u_2)$:*

$$u_1 = u_2 = \begin{cases} r^2 + \log(1 + r^2), & r \leq R; \\ \dfrac{r^2}{b/2} + \left(1 - \dfrac{1}{b/2}\right)R^2 + \dfrac{\log(1 + r^2)}{b} + \left(1 - \dfrac{1}{b}\right)\log(1 + R^2), & otherwise. \end{cases}$$

where again $R = \frac{1}{2}$ and $r = \sqrt{x^2 + y^2}$.

For any b, the solution $\mathbf{u}$ is continuous across the interface $r = 1/2$ and $[\sigma\mathbf{n}]|_\Gamma = \mathbf{q} = (q_1, q_2)$,

$$q_1 = \frac{2(3x^2 + 2xy + y^2)}{\sqrt{x^2 + y^2}}, \qquad q_2 = \frac{2(3y^2 + 2xy + x^2)}{\sqrt{x^2 + y^2}}.$$

In Table 2 we show the result of a grid refinement analysis of our method. We can see once again that we have first order accuracy in the H^1 norm, and second order accuracy in L^∞ norm as we would expect. The results also show that our method can handle small or large jumps in the physical parameters and in the solution. The convergence order is almost independent of the jumps in the coefficients for the relative error. Note that the magnitude of the solution is getting larger as the parameter $b = \lambda^+ = \mu^+$ is getting smaller, or vice versa. Nevertheless, our method can handle these situations very well.

Example 3. *We consider a plate consist of two materials with a non-circle interface in a state of plane strain. The interface is given by $r = 0.5 + 0.2\sin\theta$ in polar coordinates. The plate is $2m \times 2m$ with the bottom fixed and 1e7 N/m force applied on the top, see Fig. 1. The Young's modulus and Poisson's ratio for the inner and outer materials are $E^- = 1e9Pa$, $\nu^- = 0.3$, and $E^+ = 1e10Pa$, $\nu^+ = 0.3$ respectively. We wish to compute the displacement distribution of the plate.*

The purpose of this example is to show that our method can handle complicated geometry and large coefficients. Under the pressure, the plate will undergo a compression in y-direction and will react by stretching in the x-direction in order to reduce the change in the volume.

We set the origin at the center of the plate and use a 320×320 mesh to do the simulation. We use

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

TABLE 2. A grid refinement analysis in the maximum and H^1 norms of the finite element methods using the locally modified meshes for Example 2 with different jump ratio.

	$\lambda^+ = \mu^+ = 10$			
N	r_∞	ratio	r_1	ratio
20	3.7083E-3		2.3834E-2	
40	1.4063E-3	2.6369	8.6210E-3	2.7646
80	4.1231E-4	3.4108	3.2720E-3	2.6348
160	9.2478E-5	4.4585	1.1867E-3	2.7573
320	2.3622E-5	3.9149	4.2083E-4	2.8198
	$\lambda^+ = \mu^+ = 1000$			
N	r_∞	ratio	r_1	ratio
20	9.2119E-3		4.4262E-2	
40	3.3598E-3	2.7418	1.6187E-2	2.7344
80	9.8181E-4	3.4220	6.2464E-3	2.5915
160	2.2333E-4	4.3962	2.2636E-3	2.7596
320	5.4229E-5	4.1183	8.0108E-4	2.8257
	$\lambda^+ = \mu^+ = 0.1$			
N	r_∞	ratio	r_1	ratio
20	1.8451E-3		1.0097E-2	
40	5.3678E-4	3.4373	3.6190E-3	2.7901
80	1.5390E-4	3.4878	1.3593E-3	2.6625
160	4.2369E-5	3.6324	4.1334E-4	3.2885
320	1.1251E-5	3.7658	1.4863E-4	2.7810
	$\lambda^+ = \mu^+ = 0.001$			
N	r_∞	ratio	r_1	ratio
20	1.9727E-3		1.0169E-2	
40	5.8714E-4	3.3598	3.6591E-3	2.7792
80	1.6904E-4	3.4734	1.3775E-3	2.6564
160	4.6768E-5	3.6144	4.1916E-4	3.2863
320	1.2460E-5	3.7535	1.5067E-4	2.7820

to compute the Lamé coefficients μ^-, λ^- and μ^+, λ^+ for this is a plane strain problem.

In Fig. 7, we show the contour plot for the displacement distribution. The results are symmetric along y-axis for x-component u of displacement and symmetric along x-axis for y-component v of displacement. In Fig. 8, we show the position of the original and deformed plate. Note that, the displacement is very small. To make it visible, we have rescaled the displacement 50 times larger in the plot. The result agrees with physical intuition and reasoning.

5. Conclusions

In this paper, we have proposed a finite element method for solving plane elasticity problems with interfaces using locally modified meshes. The locally modified mesh is easy to generate from a Cartesian grid in which some nodal points near the interface are moved on the interface. It can be observed from our numerical
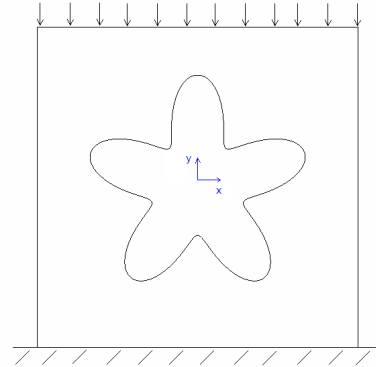
FIGURE 6. A rectangular plate with a non-circle interface $r = 0.5 + 0.2 \sin \theta$. A uniform force is applied at the top of the boundary.
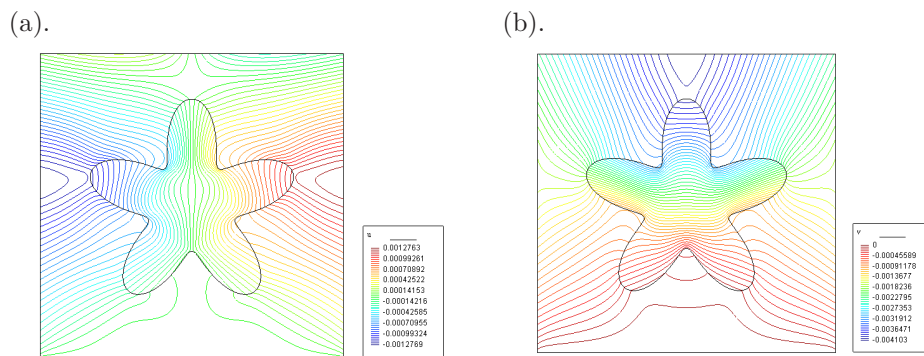
(a).   (b).

FIGURE 7. Contour plots of the displacement distribution $\mathbf{u} = (u, v)$. (a), the contour plot of u; (b), the contour plot of v.
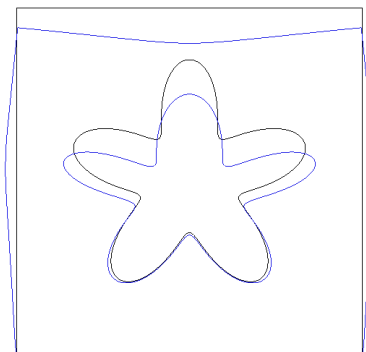

FIGURE 8. The original and deformed plate. To make it visible, we have rescaled the displacement 50 times larger in the plot.

experiments that the conforming finite element discretization based on this mesh

leads to second order accurate solutions for plane elasticity problems with inter-
faces. With the adaptation the triangles are quasi-uniform and the mesh maintains
the Cartesian structure. Particularly the Cartesian structure is useful when storing
matrices and developing iterative solvers.

References

[1] Finite element program generator. Http://www.fegensoft.com.
[2] C. Börgers. A triangulation algorithm for fast elliptic solvers based on domain imbedding. *SIAM J. Numer. Anal.*, 27:1187–1196, 1990.
[3] P.G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland Publishing Co., Amsterdam, 1978.
[4] K. Feng and Z. Shi. *Mathematical theory of elastic structures*. Science Press, 1996.
[5] Y. Gong. Immersed-interface finite-element methods for elliptic and elasticity interface problems. North Carolina State University, 2007.
[6] Y. Gong, B. Li, and Z. Li. Immersed-interface finite-element methods for elliptic interface problemswith non-homogeneous jump conditions. *SIAM J. Numer. Anal.*, 46:472–495, 2008.
[7] Y. Gong and Z. Li. Immersed interface finite element methods for elasticity interface problems with non-homogeneous jump conditions. *Numer. Math. Theo. Meth. Appl.*, 3:23–39, 2010.
[8] K. Ito, Z. Qiao, and J. Toivanen. A domain decomposition solver for acoustic scattering by elastic objects in layered media. *J. Comput. Phs.*, 227:8685–8698, 2008.
[9] K. Ito and J. Toivanen. Preconditioned iterative methods on sparse subspaces. *Appl. Math. Lett.*, 19:1191–1197, 2006.
[10] K. Ito and J. Toivanen. A fast iterative solver for scattering by elastic objects in layered media. *Appl. Numer. Math*, 57:811–820, 2007.
[11] Z. Li, T. Lin, and X. Wu. New Cartesian grid methods for interface problem using finite element formulation. *Numer. Math.*, 96:61–98, 2003.
[12] Z. Li and X. Yang. An immersed finite element method for elasticity equations with interfaces. In *Proc. Symp. Appl. Math.* Z-C. Shi et. al., editors. AMS, 2004.
[13] N. I. Muskhelishvili. *Some basic problems of the mathematical theory of elasticity*. Groningen, P. Noordhoff, 1963.
[14] W. Proskurowski and O.B. Widlund. A finite element capacitance matrix method for the neumann problem for laplace's equation. *SIAM J. Sci. Statist. Comput.*, 1:410–425, 1980.
[15] M.H. Sadd. *Elasticity: Theory, Application and Numerics*. Amsterdam: Elsevier Butterworth Heinemann, 2005.
[16] S. P. Timoshenko and J. N. Goodier. *Theory of Elasticity*. McGraw-Hill, New York, 1985.
[17] H. Xie, K. Ito, Z. Li, and J. Toivanen. A finite element method for interface problems with locally modified triangulation. *Contemporary Mathematics*, 466:179–190, 2008.
[18] X. Yang. Immersed interface method for elasticity problems with interfaces. North Carolina State University, 2004.
[19] 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.
[20] D. Yu. A system of plane elasticity canonical integral equations and its application. *J. Comp. Math.*, 4:200–211, 1986.
[21] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite Element Method: Its Basis And Fundamentals, 6th ed.* Elsevier, Oxford, 2005.

Department of Chemical Engineering & Materials Science, University of Minnesota, Minneapo-
lis, MN 55455-0132, USA
   E-mail: hxie@umn.edu

Center for Research in Scientific Computation & Department of Mathematics, North Carolina
State University, Raleigh, NC 27695-8205, USA
   E-mail: zhilin@math.ncsu.edu

Institute for Computational Mathematics & Department of Mathematics, Hong Kong Baptist
University, Kowloon Tong, Hong Kong
   E-mail: zqiao@hkbu.edu.hk