Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **9**, No. 5, pp. 1189-1205

DOI: 10.4208/aamm.2015.m1236
October 2017

A Weak Formulation for Solving the Elliptic Interface Problems with Imperfect Contact

Liqun Wang[1], Songming Hou[2] and Liwei Shi[3,*]

[1] *Department of Mathematics, College of Science, China University of Petroleum (Beijing), Beijing 102249, China*
[2] *Department of Mathematics and Statistics, Louisiana Tech University, Ruston, LA 71272, USA*
[3] *Department of Science and Technology Teaching, China University of Political Science and Law, Beijing 102249, China*

Received 2 August 2015; Accepted (in revised version) 20 December 2016

Abstract. We propose a non-traditional finite element method with non-body-fitting grids to solve the matrix coefficient elliptic equations with imperfect contact in two dimensions, which has not been well-studied in the literature. Numerical experiments demonstrated the effectiveness of our method.

AMS subject classifications: 65N30

Key words: Elliptic equation, jump condition, matrix coefficient, imperfect contact.

1 Introduction and formulations

Elliptic interface problems occur in a variety of disciplines when there are multi-physics and multi-phase materials, such as in electromagnetics, material science, and fluid dynamics. In this paper we consider a class of special elliptic interface problems with imperfect contact conditions, which occurs in heat transfer in composite media, heat transfer in building [1], transient behavior for the thermoelastic contact of two rods of dissimilar materials [2], etc.

The numerical model of the elliptic interface problem with imperfect contact is as follows: for ease of discussion, we consider a rectangular domain $\Omega = (x_{\min},x_{\max}) \times (y_{\min},y_{\max})$ (If the boundary is of general geometry, we can imbed the research domain in a larger rectangular box. Please refer to [3] for details on dealing with boundaries with general geometry). Γ is an interface prescribed by the zero level-set $\{(x,y) \in \Omega \mid \phi(x,y) =$

*Corresponding author.
Email: wliqunhmily@gmail.com (L. Q. Wang), shou@latech.edu (S. M. Hou), sliweihmily@gmail.com (L. W. Shi)

 ©2017 Global Science Press

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

$0\}$ of a level-set function $\phi(x,y)$. The advantage of using the level-set function is to represent interface cut locations on the grids without having to parameterize the interface. The unit normal vector of Γ is $\mathbf{n}=\frac{\nabla\phi}{|\nabla\phi|}$ pointing from $\Omega^-=\{(x,y)\in\Omega \mid \phi(x,y)\leq 0\}$ to $\Omega^+=\{(x,y)\in\Omega \mid \phi(x,y)\geq 0\}$. Consider the problem with imperfect contact condition:

$$\begin{aligned}
&-\nabla\cdot(\beta(x)\nabla u(x))=f(x), && x\in\Omega^{\pm},\\
&[u(x)]=\lambda(x)\beta^{+}(x)\nabla u^{+}(x)\cdot\mathbf{n}, && x\in\Gamma,\\
&[(\beta(x)\nabla u(x))\cdot\mathbf{n}]=0, && x\in\Gamma,\\
&u(x)=g(x), && x\in\partial\Omega,
\end{aligned}$$

in which $x=(x_1,x_2)$ denotes the spatial variables and ∇ is the gradient operator. The coefficient $\beta(x)$ is assumed to be a 2×2 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 $L^2(\Omega)$. The bracket $[\ ]$ means the jump, i.e., the difference from two sides of the interface.

We introduce the weak solution by the standard procedure of multiplying by a test function and integrating by parts. For the problem with Dirichlet boundary condition, we have

$$\int_{\Omega^+}\beta\nabla u\cdot\nabla\psi+\int_{\Omega^-}\beta\nabla u\cdot\nabla\psi=\int_{\Omega}f\psi, \tag{1.1}$$

where ψ is in H_0^1. The traditional finite element method has both the test and trial functions using the same basis. We will discuss later that our method does not use the same basis.

Here we briefly summarize the history of interface problems. The pioneering work on interface problems was the immersed boundary method [4, 5] by Peskin in 1977. It uses a numerical approximation of the δ-function, which smears out the solution in a narrow band around the interface Γ. In [6], the immersed boundary method was combined with the level set method, resulting in a first order numerical method that is simple to implement, even in multiple spatial dimensions. However, for both methods, the numerical smearing at the interface forces continuity of the solution at the interface, regardless of the interface condition $[u]=a$, where a might not be zero.

The immersed interface method presented in [7] is a finite difference method with second-order accuracy. This method incorporates the interface conditions into the finite difference stencil, provided that neither of the two jump conditions are zero. The corresponding linear system is sparse, but not symmetric or positive definite. Various applications and extensions of the immersed interface method are discussed in [8].

In [9], on basis of the "immersed interface" method, a fast iterative method was proposed to solve constant coefficient problems with the interface conditions $[u]=0$ and $[\beta u_{\mathbf{n}}]\neq 0$. Non-body-fitting Cartesian grids are used, and then associated uniform triangulations are added on. Interfaces are not necessarily aligned with cell boundaries.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

Numerical results show that this method's conforming version achieves second order accuracy in the L^∞ norm, and higher than first order for its non-conforming version.

In order to develop a method that is robust and simple to implement, the boundary condition capturing method [10] uses the Ghost fluid method (GFM) [11] to capture the boundary conditions. A multi-grid method [12] is applied to speed up. The convergence proof for the method is provided in [13]. The boundary condition capturing method can be obtained from discretizing the weak formulation provided in [13]. The convergence proof follows naturally. The method can solve the elliptic equation with interface conditions $[u] \neq 0$ and $[\beta u_{\mathbf{n}}] \neq 0$ in two and three dimensions. However, the original version is only first order accurate. In a recent work [14], the method is improved to second order accuracy for smooth interfaces.

In [15], the immersed finite element method (IFE) is introduced for interface problems with homogeneous jump conditions. The main idea is that the test and trial function basis are continuous but not smooth across the interface in an interface triangle. In [16, 17], the method is generalized to deal with nonhomogeneous flux jump condition. In [18], the partially penalized IFE method is introduced to penalize the discontinuity of the IFE function on neighboring interface triangles.

In [19], a non-traditional finite element formulation for solving elliptic equations with smooth or sharp-edged interfaces was proposed with non-body-fitting grids for $[u] \neq 0$ and $[\beta u_{\mathbf{n}}] \neq 0$. The basis of the test function and the trial function are different in this method, the jump conditions are non-homogeneous and are involved in the basis of trial function. It is capable of dealing with matrix coefficient problems. It achieved second order accuracy in the L^∞ norm for smooth interfaces and about 0.8th order accuracy for sharp-edged interfaces. It was later improved to second order accuracy for sharp-edged interface in [20] and had various generalizations [21–24]. In [25], the matched interface and boundary (MIB) method was proposed to solve elliptic equations with smooth interfaces. In [26], the MIB method was generalized to treat sharp-edged interfaces.

The finite volume method is also studied for elliptic interface problems, such as [27, 28] for two dimensional problems and [29] for three dimensional problems.

To summarize, elliptic interface problem can be solved using finite difference, finite element and finite volume type methods. Non-body-fitted grids can save the computational cost of mesh generation.

Although elliptic interface problem has been well-studied, all the above mentioned work are for perfect contact jump condition. For imperfect contact condition, there are very few references [1, 30] in the literature and they are either for one dimensional problems or for a very special case of two dimensional problems with straight lines parallel to grid as interface and can be viewed as tensor product of one dimensional problems. The general two dimensional problem has not been studied due to its complexity.

In this paper, we modify the method introduced in [20] to deal with imperfect contact condition. In the previous work, the jump conditions are the jump in the solution and jump in the flux as given functions. For the imperfect contact, the jump in the solution is related to the flux, making it more difficult to handle. We carefully set up the local

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

equation to deal with this new type of jump condition. Under the jump conditions in our previous work, the resulting large sparse matrix is non-symmetric but positive definite. With the new setup, we will show using an example that the resulting large sparse matrix is no longer positive definite, but it still has the 7-diagonal structure as before. Extensive numerical experiments demonstrate the effectiveness of the method. The condition number of the coefficient matrix is also studied. The growth rate is the same as that without the interface.

2 Numerical method

In this paper, we restrict ourselves to a rectangular domain $\Omega=(x_{\min},x_{\max})\times(y_{\min},y_{\max})$ in the plane, 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 $\Delta y=(y_{\max}-y_{\min})/J$. Define $(x_i,y_j)=(x_{\min}+i\Delta x,y_{\min}+j\Delta y)$ for $i=0,\cdots,I$, and $j=0,\cdots,J$, as a uniform Cartesian grid. 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.

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

$$H_{\pm}^{1,h}=\{\omega^h=(\omega_{i,j}):0\le i\le I,\ 0\le j\le J\},$$

and

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

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 $y=\frac{y_{j+1}-y_j}{x_i-x_{i+1}}(x-x_{i+1})+y_j$, 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 K, 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 on these two triangulations for our method.

If $\phi(x_i,y_j)\le0$, we count the grid point (x_i,y_j) as in $\overline{\Omega^-}$; otherwise we count it as in Ω^+. An edge of a triangle in the triangulation is called 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.

A cell K is called an interface cell if its vertices belong to different subdomains. In an interface cell, we write $K=K^+\bigcup K^-$. 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 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 $\Omega^+\bigcup\Gamma$ and the vertices of K^- are located in $\Omega^-\bigcup\Gamma$. K^+ and $K^-$are approximations of the regions $K\bigcap\Omega^+$ and

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

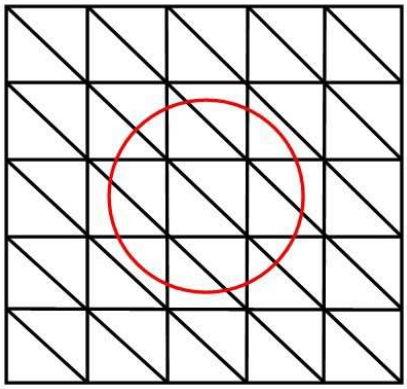


Figure 1: A uniform triangulation.

$K\bigcap\Omega^-$, respectively. A cell K is called a regular cell if all its vertices belong to the same subdomain, either Ω^+ or Ω^-. For a regular cell, we also write $K=K^+\bigcup K^-$, 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, K^+ and K^- are approximations of the regions $K\bigcap\Omega^+$ and $K\bigcap\Omega^-$, respectively. We use $|K^+|$ and $|K^-|$ to represent the areas of K^+ and K^-, respectively.

Two finite element isomorphism mappings are needed. The first one is $T^h:H_{\pm}^{1,h}\to H_0^1(\Omega)$. 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 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 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 interface Γ, hence the interface condition $[u]$ could be and is enforced exactly at these two end points. In each interface cell, the interface condition $[\beta\nabla u\cdot\mathbf{n}]$ is enforced at the middle point of Γ_K^h. Similar versions of such extension can be found in literature [10,31].

In order to construct the local system, two kinds of triangles need to be considered, see Fig. 2. The solution to the local system provides necessary information to obtain the coefficients for the large sparse linear system of the PDE.

Case 1: Because the test basis function used in this paper are linear functions, for the regular cell, see Fig. 2(a), we have

$$\begin{aligned} ap_{1x}+bp_{1y}+c&=u_{p_1},\\ ap_{2x}+bp_{2y}+c&=u_{p_2},\\ ap_{3x}+bp_{3y}+c&=u_{p_3}. \end{aligned}$$

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

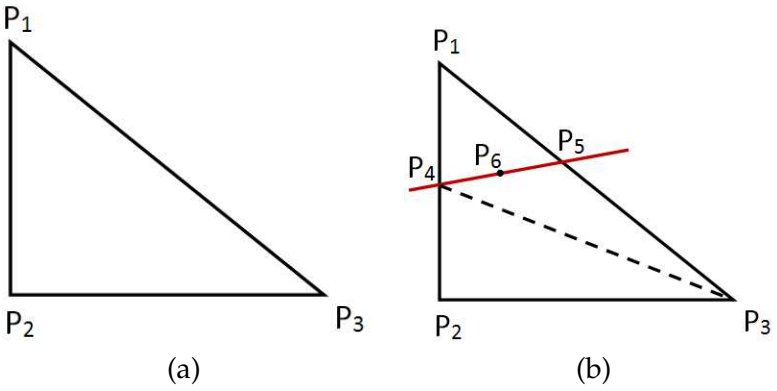


Figure 2: Two cases of triangles: (a) a regular triangle. (b) an interface triangle.

Let

$$A=\begin{pmatrix} p_{1x} & p_{1y} & 1 \\ p_{2x} & p_{2y} & 1 \\ p_{3x} & p_{3y} & 1 \end{pmatrix},$$
$$\mathbf{u}=[u_{p_1},u_{p_2},u_{p_3}]^T.$$

We have

$$[a,b,c]^T=A^{-1}\mathbf{u}. \tag{2.1}$$

So the gradient $\nabla u=[a,b]^T$ can be given by a linear function of u_{p_1}, u_{p_2} and u_{p_3}.

Case 2: For problems with interfaces, solving the local system of the interface cell is very important. The imperfect interface problem is very different from the perfect contact interface problems we solved before in [20–24]. The jump conditions depends much more on the gradient. For an interface cell with imperfect contact condition, see Fig. 2(b), in order to compute the integral on the triangle $T_{p_1p_2p_3}$, a local system should be constructed.

Suppose triangle $T_{p_1p_4p_5}\in\Omega^+$ and quadrilateral $Q_{p_2p_3p_4p_5}\in\Omega^-$, we have

$$\begin{aligned} a_1p_{1x}+b_1p_{1y}+c_1&=u_{p_1},\\ a_1p_{4x}+b_1p_{4y}+c_1&=u^+_{p_4},\\ a_1p_{5x}+b_1p_{5y}+c_1&=u^+_{p_5}, \end{aligned}$$

and

$$\begin{aligned} a_2p_{2x}+b_2p_{2y}+c_2&=u_{p_2},\\ a_2p_{3x}+b_2p_{3y}+c_2&=u_{p_3},\\ a_2p_{4x}+b_2p_{4y}+c_2&=u^-_{p_4}. \end{aligned}$$

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

Let

$$A=\begin{pmatrix} p_{1x} & p_{1y} & 1 \\ p_{4x} & p_{4y} & 1 \\ p_{5x} & p_{5y} & 1 \end{pmatrix}, \qquad B=\begin{pmatrix} p_{2x} & p_{2y} & 1 \\ p_{3x} & p_{3y} & 1 \\ p_{4x} & p_{4y} & 1 \end{pmatrix},$$
$$\mathbf{u}^+=[u_{p_1},u^+_{p_4},u^+_{p_5}]^T, \qquad \mathbf{u}^-=[u_{p_2},u_{p_3},u^-_{p_4}]^T.$$

We have

$$[a_1,b_1,c_1]^T=A^{-1}\mathbf{u}^+, \tag{2.2a}$$
$$[a_2,b_2,c_2]^T=B^{-1}\mathbf{u}^-. \tag{2.2b}$$

So the gradient $\nabla u^+=[a_1,b_1]^T$ can be given by a linear function of u_{p_1}, $u^+_{p_4}$ and $u^+_{p_5}$, and the gradient $\nabla u^-=[a_2,b_2]^T$ can be given by a linear function of u_{p_2}, u_{p_3} and $u^-_{p_4}$.

Also, the local system can be given by

$$u^+_{p_4}-u^-_{p_4}=\lambda_{p_4}\beta^+_{p_4}\nabla u^+\cdot\mathbf{n}, \tag{2.3a}$$
$$u^+_{p_5}-u^-_{p_5}=\lambda_{p_5}\beta^+_{p_5}\nabla u^+\cdot\mathbf{n}, \tag{2.3b}$$
$$u^-_{p_5}=[p_{5x},p_{5y},1]B^{-1}\mathbf{u}^-, \tag{2.3c}$$
$$\beta^+_{p_6}\nabla u^+\cdot\mathbf{n}-\beta^-_{p_6}\nabla u^-\cdot\mathbf{n}=0. \tag{2.3d}$$

We can get the values of $u^\pm_{p_4}$ and $u^\pm_{p_5}$ in the form of a linear function of u_{p_1}, u_{p_2} and u_{p_3} by solving the local system (2.3). Put the linear combination of $u^\pm_{p_4}$ and $u^\pm_{p_5}$ in Eq. (2.2), the gradient $\nabla u^\pm$ now can be denoted by the linear function of u_{p_1}, u_{p_2} and u_{p_3}, so Theorem 3.1 in [20] holds for the method used in this paper.

After getting all the gradient ∇u, the following method is used to construct the linear system of the imperfect interface problem.

Method 2.1. Find a discrete function $u^h\in H^{1,h}_\pm$ such that $u^h=g^h$ on boundary points so that for all $\psi^h\in H^{1,h}_0$, we have

$$\sum_{K\in T^h}\left(\int_{K^+}\beta\nabla U^h(u^h)\cdot\nabla T^h(\psi^h)+\int_{K^-}\beta\nabla U^h(u^h)\cdot\nabla T^h(\psi^h)\right)$$
$$=\sum_{K\in T^h}\left(\int_{K^+}fT^h(\psi^h)+\int_{K^-}fT^h(\psi^h)\right).$$

To implement the above method, we use the Gaussian quadrature rule for integrals. If T is separated into two pieces by the interface $\overline{u_{p_4}u_{p_5}}$, we connect u_{p_3} and u_{p_4}, then we get three triangles: $T_1=\triangle u_{p_1}u_{p_4}u_{p_5}$, $T_2=\triangle u_{p_2}u_{p_3}u_{p_4}$, $T_3=\triangle u_{p_3}u_{p_4}u_{p_5}$. For each triangle, label the center point p_{ij} of each edge $\overline{u_iu_j}$. In numerical computation, apply the average of three $f(p_{ij})$ in each triangle. Numerical results show the improvement over [19], where fewer sample points were used.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

For the case in [20], the right hand side of jump conditions will not affect the large sparse matrix. They will only affect the right hand side of the large linear system. However, for imperfect contact condition, since the jump in the solution is not a given function, rather, it is related to the gradient, unfortunately the local system will affect the large sparse matrix. The nice matrix property of positive definiteness in the previous case does not apply here. Despite of this, we observed numerically that the condition number of the large sparse matrix grows with about $\mathcal{O}(h^{-2})$.

3 Numerical experiments

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

$$\begin{aligned} u&=u^+(x,y) \quad \text{in } \Omega^+,\\ u&=u^-(x,y) \quad \text{in } \Omega^-, \end{aligned}$$

are given. Hence

$$\begin{aligned} f&=-\nabla\cdot(\beta\nabla u) && \text{in } \Omega,\\ \lambda(x)&=\frac{u^+-u^-}{(\beta^+\nabla u^+)\cdot\mathbf{n}} && \text{on } \Gamma. \end{aligned}$$

g is obtained as a proper Dirichlet boundary condition, since the solutions are given.

All errors of the solutions are measured in the L^∞ norm in the whole domain Ω. The grids used in all examples are $N\times N$, where $N=24i+1$, $i=1,2,\cdots,15$. Examples 3.1-3.3 are defined on $[-1,1]\times[-1,1]$, Examples 3.4 and 3.4 are defined on $[-2,2]\times[-2,2]$.

Example 3.1. The level-set function $\phi(x,y)$, the coefficients $\beta^{\pm}$ and the solutions $u^{\pm}$ are given as follows:

$$\begin{aligned} &\phi(x,y)=(x-1)^2+(y-1)^2-1.755,\\ &\beta^+(x,y)=\begin{pmatrix} \sin(xy)+2 & 0\\ 0 & \sin(xy)+2 \end{pmatrix},\\ &\beta^-(x,y)=\begin{pmatrix} (x+1)(\sin(xy)+2) & 0\\ 0 & (y+1)(\sin(xy)+2) \end{pmatrix},\\ &u^+(x,y)=x^2+y^2+2x+2y,\\ &u^-(x,y)=2(x+y). \end{aligned}$$

Fig. 3 shows the error on different grids using least square method. The numerical result shows 1.92th. order accuracy in the L^∞ norm for the solution.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

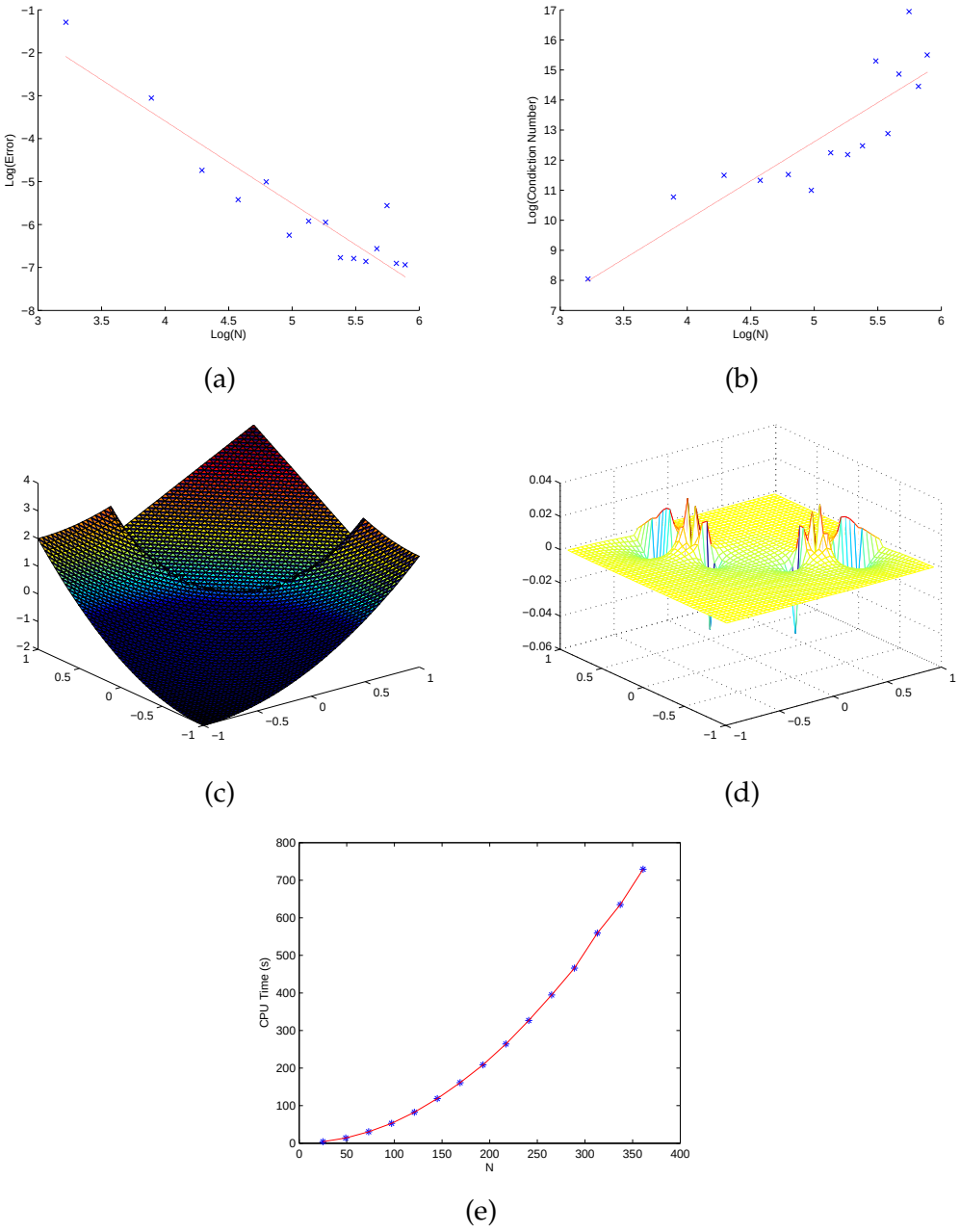


Figure 3: (a) Numerical result error on different grids, the order is 1.92. (b) Condition number of the linear system matrix on different grids, the order is -2.60. (c) Numerical result of the solution on grid 49×49. (d) Numerical error on grid 49×49. (e) CPU time of different grids.

Example 3.2. The level-set function $\phi(x,y)$, the coefficients $\beta^{\pm}$ and the solutions $u^{\pm}$ are given as follows:

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

$$\begin{aligned}
&\phi(x,y)=(x-1)^2+(y-1)^2-1.7755,\\
&\beta^+(x,y)=\begin{pmatrix}\cos(x+y)+1 & 0\\ 0 & \cos(x+y)+1\end{pmatrix},\\
&\beta^-(x,y)=\begin{pmatrix}x+1 & 0\\ 0 & y+1\end{pmatrix},\\
&u^+(x,y)=x^2+y^2+2x+2y+2,\\
&u^-(x,y)=2(\sin(x+y)+x+y)+1.
\end{aligned}$$

Fig. 4 shows the error on different grids using least square method. The numerical result shows 1.63th. order accuracy in the L^∞ norm for the solution.

Example 3.3. The level-set function $\phi(x,y)$, the coefficients $\beta^\pm$ and the solutions $u^\pm$ are given as follows:

$$\begin{aligned}
&\phi(x,y)=0.145879x-y+0.1111,\\
&\beta^+(x,y)=\begin{pmatrix}\sin(xy)+2 & 0\\ 0 & \sin(xy)+2\end{pmatrix},\\
&\beta^-(x,y)=\begin{pmatrix}(\sin(xy)+2)/2 & 0\\ 0 & (\sin(xy)+2)/2\end{pmatrix},\\
&u^+(x,y)=x^2+y^2,\\
&u^-(x,y)=2(x^2+y^2).
\end{aligned}$$

Fig. 5 shows the error on different grids using least square method. The numerical result shows 1.86th. order accuracy in the L^∞ norm for the solution.

Example 3.4. The level-set function $\phi(x,y)$, the coefficients $\beta^\pm$ and the solutions $u^\pm$ are given as follows:

$$\begin{aligned}
&\phi(x,y)=(x-2)^2+(y-2)^2-1.755,\\
&\beta^+(x,y)=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix},\\
&\beta^-(x,y)=\begin{pmatrix}2 & 0\\ 0 & 2\end{pmatrix},\\
&u^+(x,y)=2(x^2+y^2),\\
&u^-(x,y)=x^2+y^2+0.5.
\end{aligned}$$

Fig. 6 shows the error on different grids using least square method. The numerical result shows 1.78th. order accuracy in the L^∞ norm for the solution.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

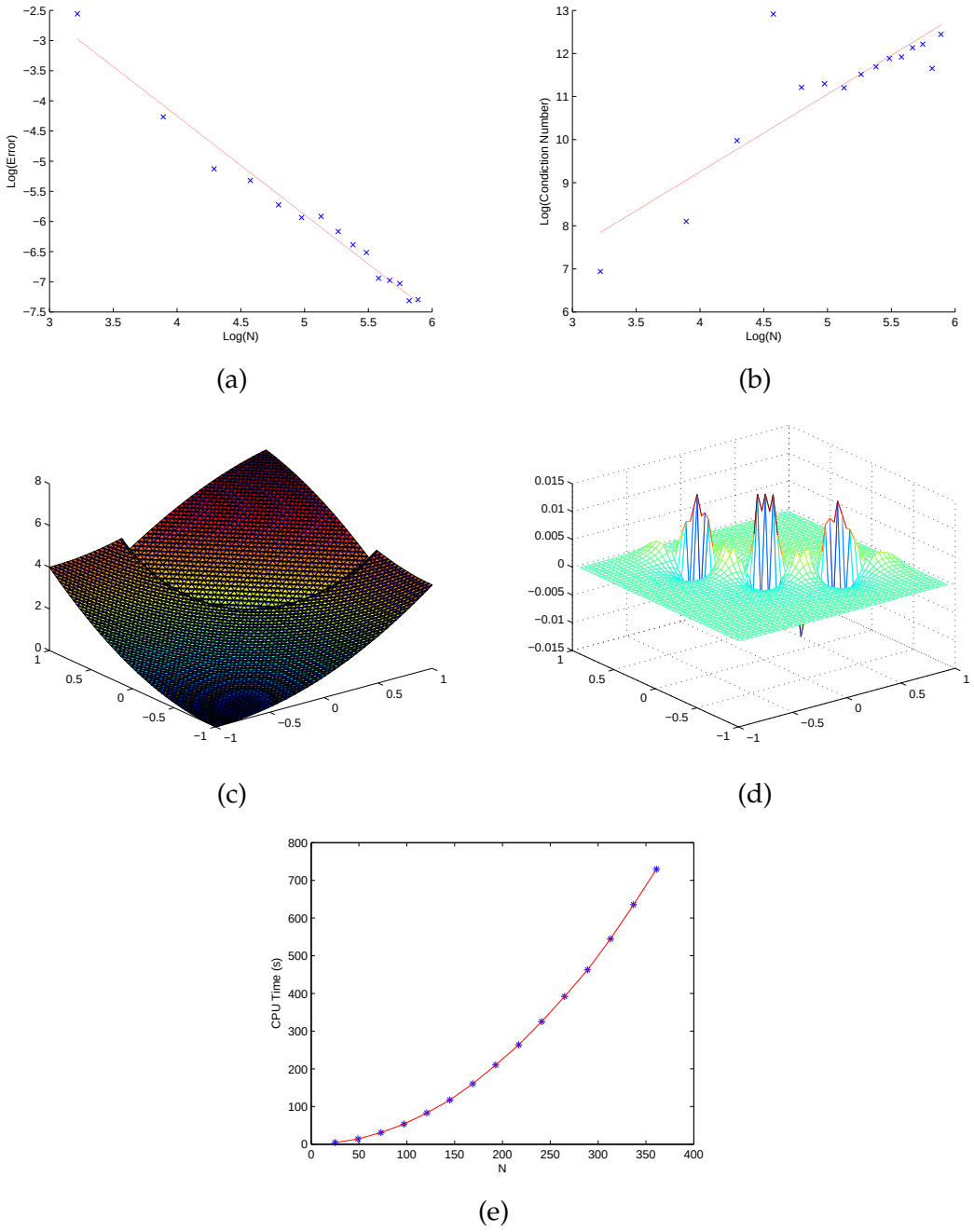
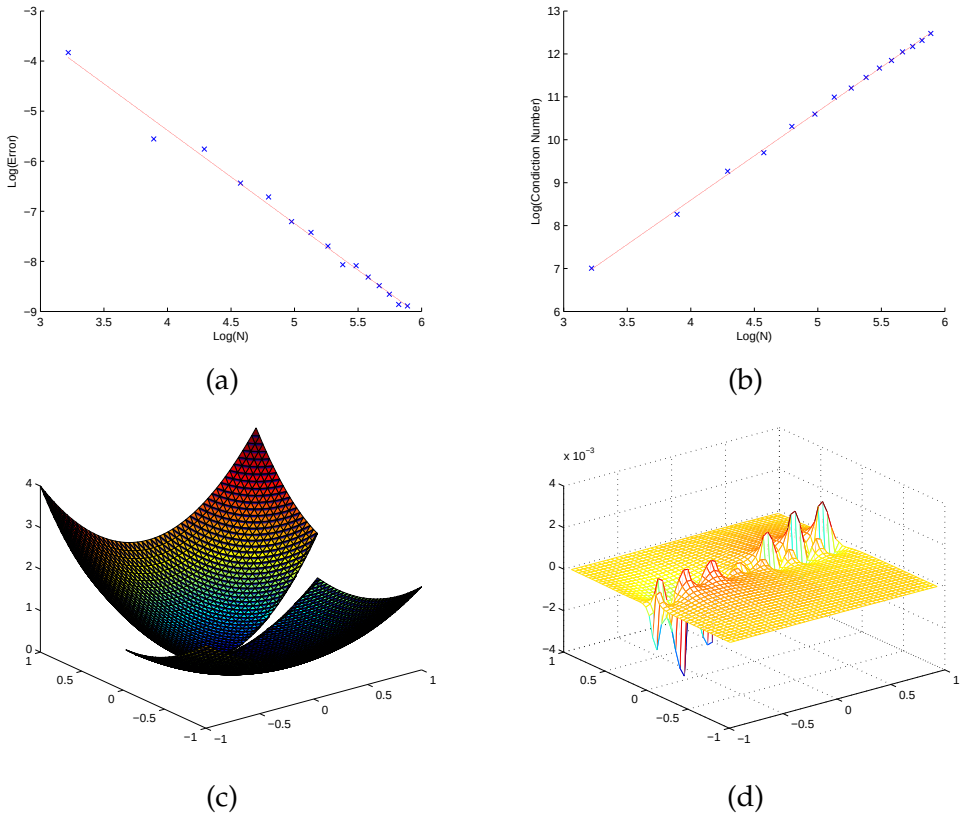


Figure 4: (a) Numerical result error on different grids, the order is 1.63. (b) Condition number of the linear system matrix on different grids, the order is -1.80. (c) Numerical result of the solution on grid 49×49. (d) Numerical error on grid 49×49. (e) CPU time of different grids.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press



(a) (b)

(c) (d)

Figure 5: (a) Numerical result error on different grids, the order is 1.86. (b) Condition number of the linear system matrix on different grids, the order is -2.06. (c) Numerical result of the solution on grid 49×49. (d) Numerical error on grid 49×49.

Example 3.5. The level-set function $\phi(x,y)$, the coefficients $\beta^{\pm}$ and the solutions $u^{\pm}$ are given as follows:

$$\begin{aligned}
&\phi(x,y)=x^2+y^2-1,\\
&\beta^+(x,y)=\begin{pmatrix}1&0\\0&1\end{pmatrix},\\
&\beta^-(x,y)=\begin{pmatrix}2&0\\0&2\end{pmatrix},\\
&u^+(x,y)=2(x^2+y^2),\\
&u^-(x,y)=x^2+y^2+1.
\end{aligned}$$

Fig. 7 shows the error on different grids using least square method. The numerical result shows 2.00th. order accuracy in the L^∞ norm for the solution.

Examples 3.1-3.5 show that our method works for various kinds of interface problems

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

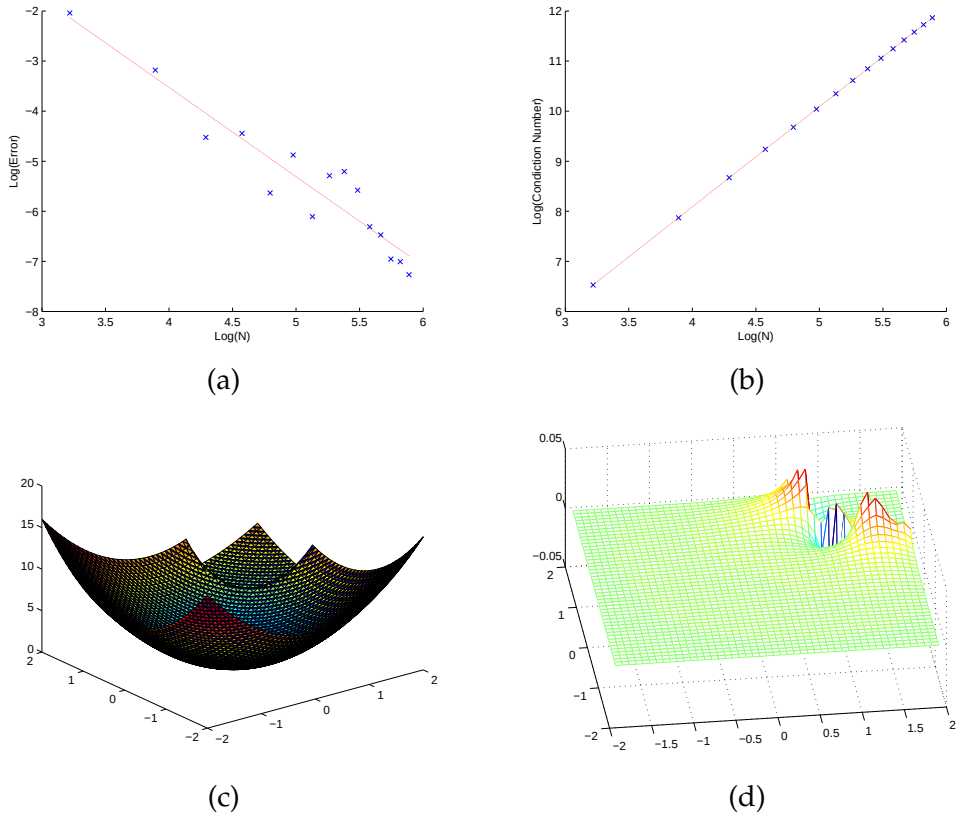
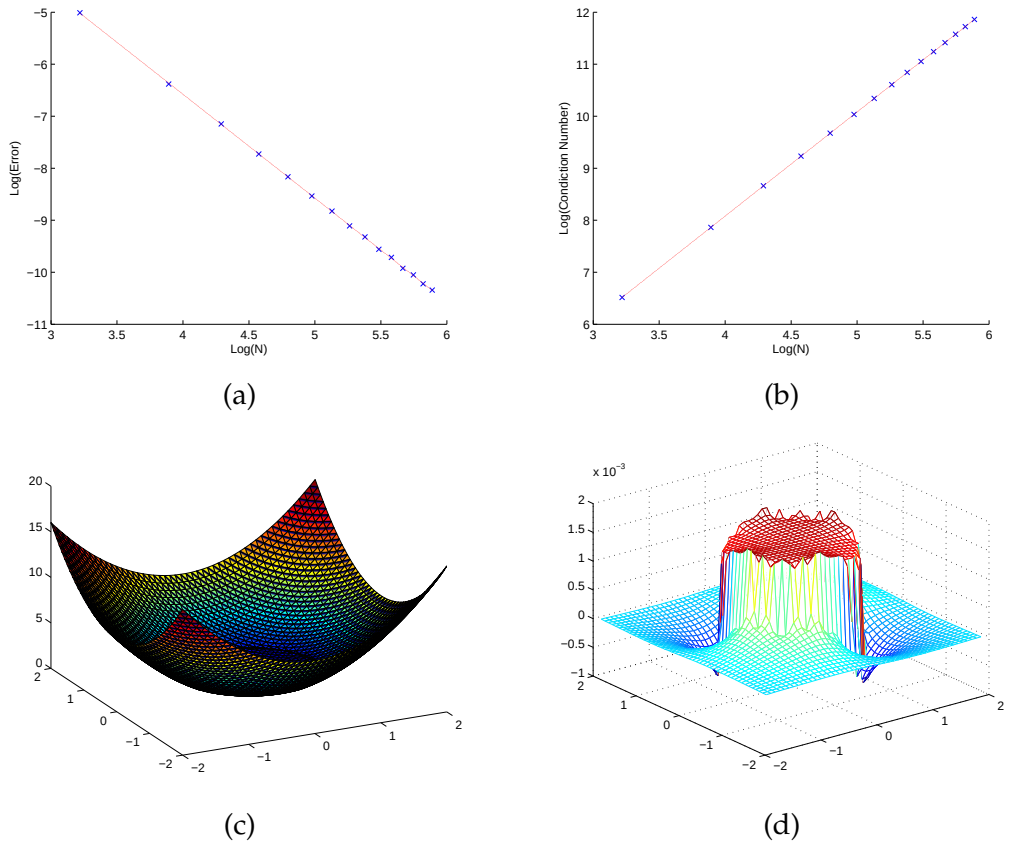


Figure 6: (a) Numerical result error on different grids, the order is 1.78. (b) Condition number of the linear system matrix on different grids, the order is -2.00. (c) Numerical result of the solution on grid 49×49. (d) Numerical error on grid 49×49.

with imperfect contact condition, including the following classes: (a) problems with complicated coefficients $\beta^{\pm}$, see Examples 3.1 and 3.2; (b) problems with constant coefficients $\beta^{\pm}$, see Examples 3.4 and 3.5; (c) problems with curve interface, see Examples 3.1, 3.2, 3.4 and 3.5; (d) problems with simple interface, see Example 3.3. We can see that the numerical results almost goes to second order accuracy in the L^{∞} norm, the largest errors locate at the grid points with interface coming across, because a triangle with poor property will exist when the interface cuts a grid point, and the real value of the gradient of this triangle is hard to calculate. This is the new challenge for imperfect contact condition that did not exist in the previous case in [20] etc. The order of the condition number is close to second order. The condition number growth in Example 3.1 is slightly faster because of the complicated coefficients $\beta^{\pm}$. The above numerical results demonstrate that simple coefficients $\beta^{\pm}$ get better numerical result, even when the function of the solution is more complicated. Examples 3.4 and 3.5 have much better condition number growth pattern because the coefficients $\beta^{\pm}$ are all constant.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press



(a) (b)

(c) (d)

Figure 7: (a)Numerical result error on different grids, the order is 2.00. (b)Condition number of the linear system matrix on different grids, the order is -2.00. (c)Numerical result of the solution on grid 49×49. (d)Numerical error on grid 49×49.

Finally, we use an example to further discuss the matrix property of imperfect contact problem compared with the standard jump condition problem.

Example 3.6. We use the same setup as in Example 3.1. Figs. 8(a) and (b) shows the signs of the k-by-k northwest submatrix determinants of the large sparse matrix A and the signs of the inverse of the large sparse matrix A for the imperfect contact condition problem as in Example 3.1. In (b) there are two colors: most points have the color representing "positive", meaning that most elements of the inverse of the large sparse matrix A are positive, with a few negative elements represented by a different color. Figs. 8(c) and (d) shows those for the standard jump conditions. Clearly, for standard jump conditions, the matrix is positive definite (though non-symmetric) and inverse-positive, meaning the inverse having all elements non-negative and therefore all points having the same color in (d). However, for our challenging problem, the condition would ruin the nice matrix properties. Still the 7-diagonal property is preserved.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

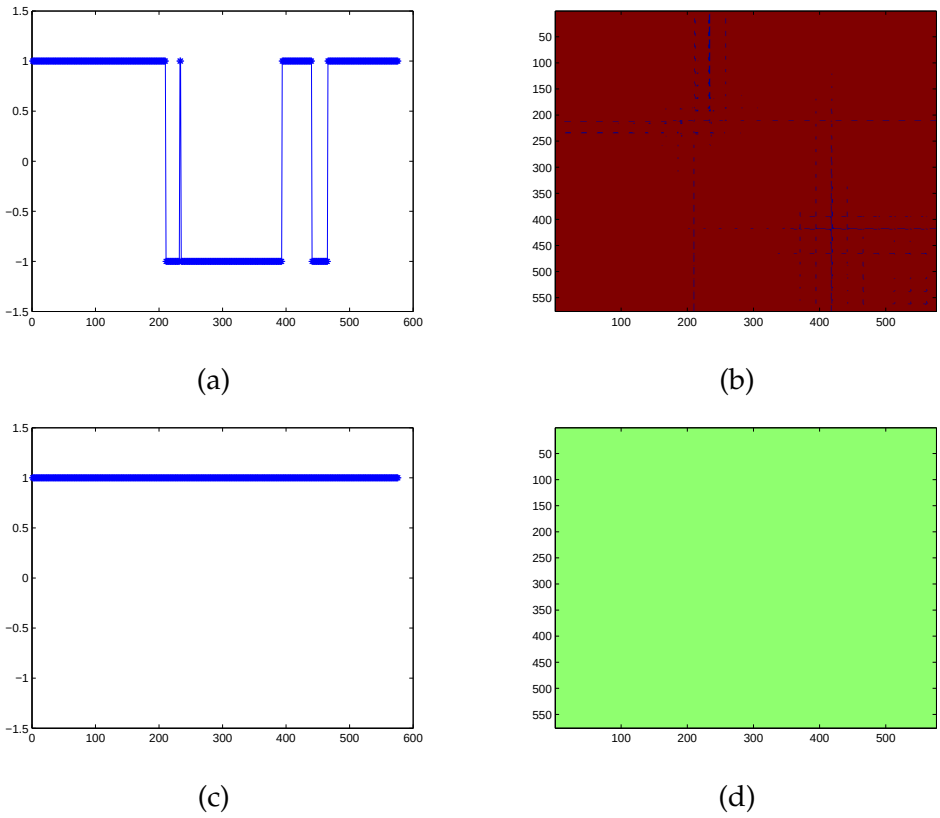


Figure 8: (a) The signs of determinants for submatrices of A for imperfect contact condition. (b) The signs of the elements in the inverse of A for imperfect contact condition. (c) The signs of determinants for submatrices of A for standard jump condition. (d) The signs of the elements in the inverse of A for standard jump condition.

4 Conclusions

We proposed a non-traditional finite element method for solving the elliptic interface problem with imperfect contact condition. We used several examples to show that our method averages slightly less than second order accuracy in L^∞ norm, and the condition number of the coefficient matrix grows with order $\mathcal{O}(h^{-2})$, which is the same as the case without interface. Unlike the case with jump conditions in [20] etc., the interface cut location would affect the convergence order, as illustrated by figures in numerical examples. The difficulty is originated from the fact that the jump in the solution and jump in the flux are coupled in the same jump condition, unlike the previous case. The inaccuracy of computing the flux would ultimately affect the convergence order. However, in average we still obtained less than but close to second order. The theoretical error estimates for standard finite element method do not apply to our proposed method, however, due to the complexity of the problem, in the literature only one dimensional problem or a very

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

special case of two dimensional problem has been discussed. Our method serves as a simple and practical method using non-body-fitted mesh to solve the elliptic interface problem with imperfect contact.

Acknowledgments

L. Shi's research is supported by the Science Foundations of China University of Political Science and Law (Nos. 1000-10816106 and 1000-10816330). S. Hou's research is supported by the NSF grant DMS-1317994 and Dr. Walter Koss Professorship. L. Wang's research is supported by the Science Foundations of China University of Petroleum-Beijing (Nos. 2462015BJB05, 2462015YQ0604 and 2462015QZDX02).

References

[1] J. WEISZ, *On an iterative method for the solution of discretized elliptic problems with imperfect contact condition,* J. Comput. Appl. Math., 72 (1996), pp. 319–333.
[2] J. R. BARBER AND R. ZHANG, *Transient behaviour and stability for the thermoelastic contact of the rods of dissimilar materials,* Int. J. Mech. Sci., 30 (1988), pp. 691–704.
[3] L. WANG AND L. SHI, *Numerical method for solving matrix coefficient elliptic equation on irregular domains with sharp-edged boundaries,* Int. J. PDE, Article ID 476873, (2013).
[4] C. PESKIN, *Numerical analysis of blood flow in the heart,* J. Comput. Phys., 25 (1977), pp. 220–252.
[5] C. PESKIN AND B. PRINTZ, *Improved volume conservation in the computation of flows with immersed elastic boundaries,* J. Comput. Phys., 105 (1993), pp. 33–46.
[6] M. SUSSMAN, P. SMEREKA AND S. OSHER, *A level set approach for computing solutions to incompressible two-phase flow,* J. Comput. Phys., 114 (1994), pp. 146–154.
[7] R. J. LEVEQUE AND Z. LI, *The immersed interface method for elliptic equations with discontinuous coefficients and singular sources,* SIAM J. Numer. Anal., 31 (1994), pp. 1019–1044.
[8] Z. LI AND K. ITO, The Immersed Interface Method: Numerical Solutions of PDES Involving Interfaces and Irregular Domains, SIAM, Philadelphia, 2006.
[9] Z. LI, *A fast iterative algorithm for elliptic interface problems,* SIAM J. Numer. Anal., 35 (1998), pp. 230–254.
[10] X. LIU, R. P. FEDKIW AND M. KANG, *A boundary condition capturing method for Poisson's equation on irregular domains,* J. Comput. Phys., 160 (2000), pp. 151–178.
[11] R. FEDKIW, T. ASLAM, B. MERRIMAN AND S. OSHER, *A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method),* J. Comput. Phys., 152 (1999), pp. 457–492.
[12] J. W. L. WAN AND X. LIU, *A boundary condition capturing multigrid approach to irregular boundary problems,* SIAM J. Sci. Comput., 25 (2004), pp. 1982–2003.
[13] X. LIU AND T. SIDERIS, *Convergence of the ghost fluid method for elliptic equations with interfaces,* Math. Comput., 72 (2003), pp. 1731–1746.
[14] P. MACKLIN AND J. S. LOWENGRUB, *A new ghost cell/ level set method for moving boundary problems: application to tumor growth,* J. Sci. Comput., 35 (2008), pp. 266–299.
[15] Z. LI, T. LIN, Y. LIN AND R. ROGERS, *An immersed finite element space and its approximation capability,* Numer. Meth. PDE, 20 (2004), pp. 338–367.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press

[16] Y. GONG, B. LI AND Z. LI, *Immersed-interface finite-element methods for elliptic interface problems with nonhomogeneous jump conditions,* SIAM J. Numer. Anal., 46 (2008), pp. 472–495.
[17] X. HE, T. LIN AND Y. LIN, *Immersed finite element methods for elliptic interface problems with non-homogeneous jump conditions,* Int. J. Numer. Anal. Model, 8 (2011), pp. 284–301.
[18] T. LIN, Y. LIN AND X. ZHANG, *Partially penalized immersed finite element methods for elliptic interface problems,* SIAM J. Numer. Anal., 53 (2015), pp. 1121–1144.
[19] S. HOU AND X. LIU, *A numerical method for solving variable coefficient elliptic equations with interfaces,* J. Comput. Phys., 202 (2005), pp. 411–445.
[20] S. HOU, W. WANG AND L. WANG, *Numerical method for solving matrix coefficient elliptic equation with sharp-edged interfaces,* J. Comput. Phys., 229 (2010), pp. 7162–7179.
[21] S. HOU, Z. LI, L. WANG AND W. WANG, *A numerical method for solving elasticity equations with interfaces,* Commun. Comput. Phys., 12 (2012), pp. 595–612.
[22] S. HOU, L. WANG AND W. WANG, *A numerical method for solving the elliptic interface problems with multi-domains and triple junction points,* J. Comput. Math, 30 (2012), pp. 504–516.
[23] S. HOU, P. SONG, L. WANG AND H. ZHAO, *A weak formulation for solving elliptic interface problems without body fitted grid,* J. Comput. Phys., 249 (2013), pp. 80–95.
[24] L. WANG, S. HOU AND L. SHI, *A numerical method for solving 3D elasticity equations with sharp-edged interfaces,* Int. J. PDE, Article ID 476873, (2013).
[25] Y. ZHOU, S. ZHAO, M. FEIG AND G. W. WEI, *High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources,* J. Comput. Phys., 213 (2006), pp. 1–30.
[26] 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 (2007), pp. 729–756.
[27] P. COLELLA AND H. JOHANSEN, *A Cartesian grid embedded boundary method for Poisson's equation on irregular domains,* J. Comput. Phys., 60 (1998), pp. 85–147.
[28] M. OEVERMANN AND R. KLEIN, *A Cartesian grid finite volume method for elliptic equations with variable coefficients and embedded interfaces,* J. Comput. Phys., 219 (2006), pp. 749–769.
[29] M. OEVERMANN, C. SCHARFENBERG AND R. KLEIN, *A sharp interface finite volume method for elliptic equations on Cartesian grids,* J. Comput. Phys., 228 (2009), pp. 5184–5206.
[30] T. CHERNOGOROVA, R. E. EWING, O. ILIEV AND R. LAZAROV, *On the discretization of interface problems with perfect and imperfect contact,* Lecture Notes in Physics, 552 (2000), pp. 93–103, New York, Springer-Verlag.
[31] Z. LI, T. LIN AND X. WU, *New cartesian grid methods for interface problems using the finite element formulation,* Numerische Mathematik, 96 (2003), pp. 61–98.

https://doi.org/10.4208/aamm.2015.m1236 Published online by Cambridge University Press