Abstract

GONG, YAN. Immersed-Interface Finite-Element Methods for Elliptic and Elasticity Interface Problems. (Under the direction of Dr. Zhilin Li.)

The purpose of the research has been to develop a class of new finite-element methods, called *immersed-interface finite-element methods*, to solve elliptic and elasticity interface problems with homogeneous and non-homogeneous jump conditions. Simple non-body-fitted meshes are used. Single functions that satisfy the same non-homogeneous jump conditions are constructed using a level-set representation of the interface. With such functions, the discontinuities across the interface in the solution and flux are removed; and equivalent elliptic and elasticity interface problems with homogeneous jump conditions are formulated. Special finite-element basis functions are constructed for nodal points near the interface to satisfy the homogeneous jump conditions. Error analysis and numerical tests are presented to demonstrate that such methods have an optimal convergence rate. These methods are designed as an efficient component of the finite-element level-set methodology for fast simulation of interface dynamics that does not require re-meshing. Such simulation has been a powerful numerical approach in understanding material properties, biological processes, and many other important phenomena in science and engineering.

Immersed-Interface Finite-Element Methods for Elliptic and Elasticity Interface Problems

BY

Yan Gong

A DISSERTATION SUBMITTED TO THE GRADUATE FACULTY OF

NORTH CAROLINA STATE UNIVERSITY

IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

APPLIED MATHEMATICS

RALEIGH, NORTH CAROLINA

July 2007

APPROVED BY:

Zhilin Li CHAIR OF ADVISORY COMMITTEE	Sharon R. Lubkin
Xiao-biao Lin	Jason Osborne

Dedication

To: My father and mother, Naichang Gong and Zhiping Liu, and to my husband Jidong and my son Caleb.

Biography

Yan Gong was born on August 5, 1973 in Beijing, P. R. China. After attending elementary, junior high, and high school, Yan was admitted to Tsinghua University, the most prestigious university in China, without taking the National Entrance Exam. This was a big honor for her family and herself, because only 90 out of over half million students received this offer.

During her time at Tsinghua University, Yan earned her Bachelor of Engineering degree in the Department of Engineering Mechanics, Minor of Management degree and Master of Econometrics degree in the school of Economics and Management. As a reward, she won "The Graduate With Honor " award.

This was not the end of her academic journey. Thanks to Dr. Zhilin Li, Yan entered the Ph.D program in Applied Mathematics at North Carolina State University with financial supports.

Upon completion of her dissertation, Yan will begin one-year visiting Assistant Professor position in the Mathematics and Statistics department at University of North Carolina - Greensboro.

Acknowledgments

I am very grateful for my advisor, Dr. Zhilin Li, for his guidance, suggestion and encouragement.

I would like to thank my committee members, Dr. Sharon R. Lubkin, Dr. Xiaobiao Lin, Dr. Jason Osborne, for all of time and effort they have invested in me during my graduate school career, both in and out of class. Their helpful comments and suggestions regarding my dissertation have been greatly appreciated.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis concerns numerical solutions of elliptic and elasticity interface problems using finite element methods.

1.1 The Model Problems and Applications

1.1.1 Single Elliptic Interface Problems

The first model problem is a two-dimensional single elliptic interface problem

$$-\nabla \cdot \beta^{-} \nabla u = f \qquad \text{in } \Omega^{-}, \tag{1.1}$$

$$-\nabla \cdot \beta^{+} \nabla u = f \qquad \text{in } \Omega^{+}, \tag{1.2}$$

$$[u]_{\Gamma} = w, \tag{1.3}$$

$$\left[\beta \frac{\partial u}{\partial n}\right]_{\Gamma} = Q, \tag{1.4}$$

$$u = g \qquad \text{on } \partial\Omega. \tag{1.5}$$

Here, $\Omega \subset \mathbb{R}^2$ is a bounded domain with its boundary $\partial\Omega$. Both Ω^- and Ω^+ are sub-domains of Ω such that $\Omega^- \cap \Omega^+ = \emptyset$ and $\overline{\Omega^-} \cup \overline{\Omega^+} = \overline{\Omega}$, where an overline denotes the closure; see Figure 1.1 for an illustration. For simplicity, we assume that $\overline{\Omega^-} \cap \partial\Omega = \emptyset$. We denote $\Gamma = \overline{\Omega^-} \cap \overline{\Omega^+}$, and call it the *interface* separating Ω^- and Ω^+. We shall assume that Γ is smooth enough. We also denote by n the unit vector normal to Γ pointing from Ω^- to Ω^+, or the unit exterior normal to $\partial\Omega$; and by $\partial/\partial n$ or ∂_n the corresponding normal derivative.

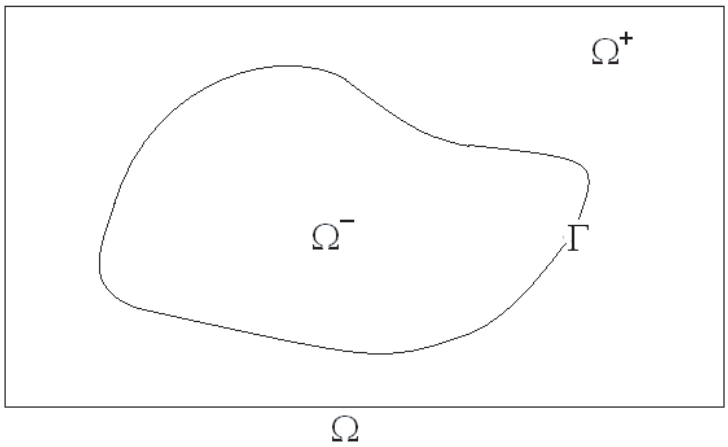

Figure 1.1: A diagram of the geometry of an elliptic interface problem.

All the functions β, $f : \Omega \to \mathbb{R}$, w, $Q : \Gamma \to \mathbb{R}$, and $g : \partial\Omega \to \mathbb{R}$ are given. As usual, for any function $\xi : \Omega \to \mathbb{R}$, we denote the restrictions of ξ on Ω^- and Ω^+ by

$$\xi^- = \xi|_{\Omega^-} \qquad \text{and} \qquad \xi^+ = \xi|_{\Omega^+},$$

respectively; and denote by

$$[\xi]_\Gamma(x) = \lim_{y\to x, y\in\Omega^+} \xi^+(y) - \lim_{y\to x, y\in\Omega^-} \xi^-(y),$$

the jump of ξ across the interface Γ at $x \in \Gamma$, when the unique limiting values on Γ of ξ from both sides of Γ exist. We assume that β^- and β^+ are smooth and bounded on Ω^- and Ω^+, respectively,

$$\beta(x) \geq \beta_0 \qquad \forall x \in \Omega \tag{1.6}$$

for some constant $\beta_0 > 0$, $f \in L^2(\Omega)$, and all w, Q, and g are smooth and bounded.

Note that the *interface problems* refer to a wide range of differential equations that involve discontinuous or non-smooth input functions (such as conductivities, densities, diffusions and the singular source terms) in partial differential equations across the interface between two or more subdomains.

Equations (1.3) and (1.4) are the non-homogeneous interface or jump conditions. The elliptic interface problems (1.1) – (1.5) with such jump conditions arise in many areas. For example, in a Burton-Cabrera-Frank type model for epitaxial growth of thin films, the adatom (adsorbed adom) density that solves diffusion equation on terraces, and the corresponding flux, can both have jumps across interfaces that represent atomic steps [5, 8, 9]. Another example is that the reaction potential of electrostatics of a solvation energy satisfies a non-homogeneous jump condition for the flux [20].

1.1.2 Elasticity Problems with Interfaces

The second model problem in this thesis is a two-dimensional elasticity problem with interfaces:

$$\nabla \cdot \sigma + F = 0 \qquad \text{in } \Omega^- \cup \Omega^+, \tag{1.7}$$

$$[u]_\Gamma = W, \tag{1.8}$$

$$[\sigma n]_\Gamma = Q, \tag{1.9}$$

$$u = G \qquad \text{on } \partial\Omega. \tag{1.10}$$

We assume that the elastic material occupies a bounded domain $\Omega \subset \mathbb{R}^2$ with its boundary $\partial\Omega$ .We define Ω^- ,Ω^+ ,Ω ,$\overline{\Omega^-}$,$\overline{\Omega^+}$,$\overline{\Omega}$,$\partial\Omega$,Γ ,$\partial/\partial n$ and ∂_n in the same way as in the single elliptic interface problem; see Figure 1.1 for an illustration.

In [7], the $u = (u_1, u_2)$ is called *displacement* and the *stress tensor* is $\sigma = (\sigma_{ij})$, σ is a 2×2 matrix and symmetric $\sigma_{ij} = \sigma_{ji}, i, j = 1, 2$. The vector $F = (f_1, f_2)$ is the *body force* which is known function. u_i is the displacement in the x_i-direction , σ_{ii} is the *normal stress* in the x_i-direction, and the $\sigma_{ij}, i \neq j$, are the *shear stresses*. The *deformation* (tensor) $\varepsilon(u)$ associated with the displacement u also is a 2×2 matrix, where the entries of $\varepsilon(u) = (\varepsilon_{ij}(u))$ are given by

$$\varepsilon_{ij}(u) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \qquad i, j = 1, 2. \tag{1.11}$$

If we assume that the material is a homogenous isotropic and linearly elastic and

that the displacements are small, the relation between stress tensor and deformation is given by

$$\sigma_{ij} = \lambda\,(\nabla \cdot u)\,\delta_{ij} + 2\,\mu\,\varepsilon_{ij}(u), \tag{1.12}$$

where λ and μ are positive constants,

$$\nabla \cdot u = \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2}\right), \tag{1.13}$$

and

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ \\ 0, & \text{if } i \neq j. \end{cases} \tag{1.14}$$

The equations (1.7) can be written as

$$-\left(\frac{\partial \sigma_{11}}{\partial x_1} + \frac{\partial \sigma_{12}}{\partial x_2}\right) = f_1, \tag{1.15}$$

$$-\left(\frac{\partial \sigma_{21}}{\partial x_1} + \frac{\partial \sigma_{22}}{\partial x_2}\right) = f_2, \tag{1.16}$$

or as

$$-\left\{(\lambda + 2\mu)\frac{\partial^2 u_1}{\partial x_1^2} + (\lambda + \mu)\frac{\partial^2 u_2}{\partial x_1 \partial x_2} + \mu\frac{\partial^2 u_1}{\partial x_2^2}\right\} = f_1, \tag{1.17}$$

$$-\left\{(\lambda + 2\mu)\frac{\partial^2 u_2}{\partial x_2^2} + (\lambda + \mu)\frac{\partial^2 u_1}{\partial x_1 \partial x_2} + \mu\frac{\partial^2 u_2}{\partial x_1^2}\right\} = f_2. \tag{1.18}$$

Similarly, the jump condition (1.9) can be rewritten as

$$\left[\lambda\left(\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}\right)n_1+2\mu\frac{\partial u_1}{\partial x_1}n_1+\mu\left(\frac{\partial u_1}{\partial x_2}+\frac{\partial u_2}{\partial x_1}\right)n_2\right]_\Gamma=q_1, \tag{1.19}$$

$$\left[\lambda\left(\frac{\partial u_1}{\partial x_1}+\frac{\partial u_2}{\partial x_2}\right)n_2+2\mu\frac{\partial u_2}{\partial x_2}n_2+\mu\left(\frac{\partial u_1}{\partial x_2}+\frac{\partial u_2}{\partial x_1}\right)n_1\right]_\Gamma=q_2, \tag{1.20}$$

where $Q=(q_1,q_2)$ and $n=(n_1,n_2)$ is the unit normal vector to the interface Γ, pointing from Ω^- to Ω^+. The jump condition (1.8) become

$$[u_1]_\Gamma=w_1, \tag{1.21}$$

$$[u_2]_\Gamma=w_2, \tag{1.22}$$

where $W=(w_1,w_2)$.

Remark The constants λ and μ in (1.17) and (1.18) can be expressed as

$$\mu=\frac{E}{2(1+\nu)},\qquad \lambda=\frac{E\nu}{(1+\nu)(1-2\nu)}, \tag{1.23}$$

where λ is called *Lamé's first parameter*, μ is the *shear modulus* or *Lamé's second parameter*. E is the ratio of a simple tension stress applied to a material to the resulting strain parallel to the tension, called the *modulus of elasticity (Young's modulus)* [34] and ν is the ratio of the transverse contracting strain to the elongation strain when a rod is stretched by forces which are applied at its ends and which are parallel to the rod's axis, called the *contraction ratio (Poisson's ratio)* of the elastic material [37]. ν is dimensionless and ranges from $0.2\sim0.49$, and is around 0.3 for most materials.

All the material parameters μ, λ, ν, and E are piecewise constants. The μ and ν are given by

$$\mu = \begin{cases} \mu^+, & \text{in } \Omega^+, \\ \mu^-, & \text{in } \Omega^-, \end{cases} \tag{1.24}$$

$$\nu = \begin{cases} \nu^+, & \text{in } \Omega^+, \\ \nu^-, & \text{in } \Omega^-, \end{cases} \tag{1.25}$$

where all $\mu^+, \mu^-, \nu^+, \nu^-$ are positive constants.

All the functions $f_1,\ f_2 : \Omega \to \mathbb{R}$, $w_1,\ w_2,\ q_1,\ q_2 : \Gamma \to \mathbb{R}$, and $g_1,\ g_2 : \partial\Omega \to \mathbb{R}$ are given, here $G = (g_1, g_2)$ and $f_1, f_2 \in L^2(\Omega)$, all $w_1,\ w_2,\ q_1,\ q_2,\ g_1$ and g_2 are smooth and bounded. $\square$

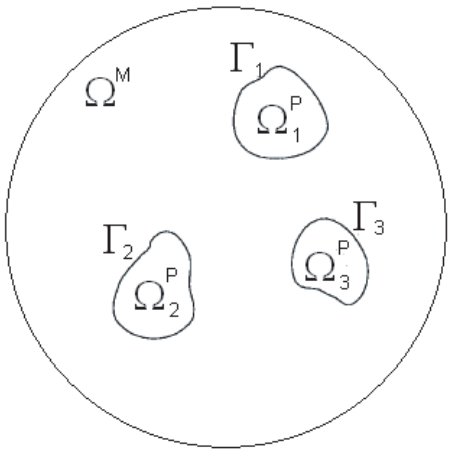

Figure 1.2: A two-phase domain with three precipitates (p = 3).

Elasticity interface problems (1.7)– (1.10) with such jump conditions often occur in material science. For example, in [22], they simulate the diffusional evolution of microstructure in stressed solids which caused by the solid state diffusional transformation.When a sudden lowering of temperature drives the thermodynamically stable mixture of solids away from equilibrium, the solid state diffusional transformation is produced. We use a two-phase domain with three precipitates as the example, the matrix and precipitate phases occupy the two-dimensional plane $\mathbb{R}^2$; see Figure 1.2 for an illustration. When the equilibrium is broken, the system intends to rebuild the balance. There are two distinct phases involved in the diffusional evolution – *growth* and *equilibration*. Growth occurs to satisfy a local mass balance relation at each precipitate interface. Equilibration produces a dynamic rearrangement of mass in the system in order to reach a global mass balance and minimize a global energy. The final result of the diffusional evolution is either it reaches a new equilibrium or it is stopped by quenching. Understanding this kind of physical process is very important to improve material stability properties.

1.2 The Essence of Our Methods

1.2.1 Finite Element Methods

In [42], the brief history of Finite Element Analysis is provided. Finite Element Analysis was first presented in 1943 by Richard Courant (1888 - 1972), who was a German-American mathematician. Richard Courant utilized the Ritz method of numerical analysis and variational calculus to get approximate solutions to vibration

systems. Shortly thereafter, a paper [41] published in 1956 by M. J. Turner, R. W. Clough, H. C. Martin, and L. J. Topp established a broader definition of numerical analysis. In 1964, IBM 7094, Univac 1107, etc., were used for American aircraft FE programs. The automotive industry began to use FEA in 1970. Since computers were very expensive and generally owned by the aeronautics, automotive, defense, and nuclear industries by the early 70's, FEA was limited. Present day supercomputers have incredible computing power and since the cost of computers has declined rapidly, FEA has been well developed. The benefits of FEA include increased accuracy, enhanced design and better insight into critical design parameters, virtual prototyping, fewer hardware prototypes, a faster and less expensive design cycle, increased productivity, and increased revenue [18].

Finite element analysis (FEA) is a computer simulation technique used by engineers, scientists, and mathematicians. It uses a numerical technique called the *finite element method (FEM)*. There are two steps: dividing the domain of the solution into a finite number of simple subdomains, called *finite elements* and using variational concepts to obtain an approximation of the solution to the differential equations that describe, or approximately describe a wide variety of physical (and non-physical) problems over the collection of finite elements [6]. The method can transfer a complicated domain to a series of smaller regions in which the differential equations are approximately solved. Elements are connected at specific points, called *nodes*. By assembling the set of linear equations for each element, the behavior over the entire domain is determined and the solution must to be continuous along common boundaries of adjacent elements. The *FEM* can be used to deal with many

physical problems: solid, fluid, soil mechanics and electromagnetism or dynamics.

The reasons we choose finite element methods are the following:

- *Immersed Interface Method (IIM)* is originally invented by R. J. LeVeque and Z. Li in [25, 27]. *IIM* is based on the *Finite Difference (FD)* and *Cartesian Grids*. *IIM* is well-developed for *scalar equations*, but FEM will generally give better conditioned linear systems of equations, see [46].
- Since*FEM* is based on the weak or variational formulation in the integral form, it reduces the requirement of the regularity of the solutions.
- The linear system obtained from *FEM* is better conditioned than from *FD*; it is symmetric for self-adjoint system.
- *FD* needs partial derivatives of the coefficient function $\beta(x)$ for the single elliptic interface problem.
- There are some difficulties for the theoretical analysis by using *FD*.
- We believe the research in this thesis also provides necessary preparations to get higher order accuracy, say fourth order, for the interface problems.

1.2.2 Cartesian Grids

In [45], the definition of the *Cartesian grid* is given. A *Cartesian grid* is a special case where the elements are unit squares or unit cubes, and the vertices are integer points, see Figure 1.3 for an illustration.

In this thesis, we only focus on two dimensional problems with the interface and all elements being triangles. We choose Cartesian grids to generate the triangulation over Ω. For an example, see Figure 1.4, where $\Omega = (-1, 1) \times (-1, 1)$ and the interface

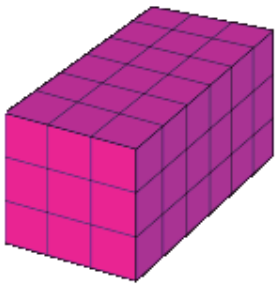

Figure 1.3: An example of Cartesian grid in 3–D.

Γ is the circle centered at point $(0, 0)$ with radius $R = 0.5$. Yellow triangles are *interface elements* that the interface cuts through; white triangles are *non-interface elements*.

There are several advantages using Cartesian grids:

- **No cost for grid generation**

  This is potentially significant when we deal with moving interface problems. In every single time step, since grid points are fixed, we do not need to generate mesh every time.

- **Taking advantages of black box solvers**

  There are many software packages based on the Cartesian grids. Z. Li, D. Calhoun, Z. Yang, R. J. LeVeque, A. Almgren and many others have applied fast Poisson solvers, the Clawpack, multigrid solvers, and level set methods in their research [4, 32, 24, 10, 47, 11].

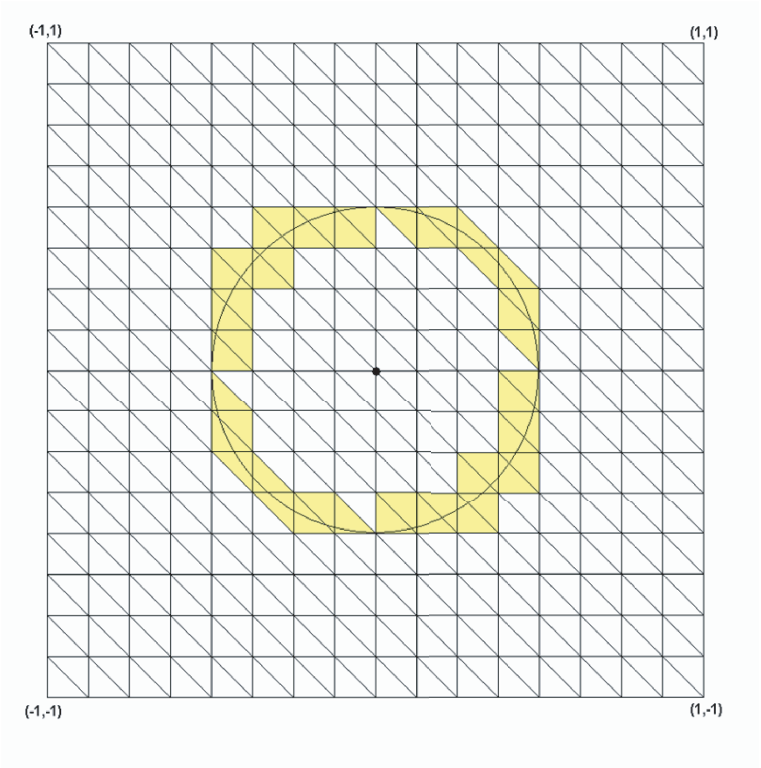

Figure 1.4: Cartesian grid to generate a triangulation over Ω.

1.2.3 The Level Set Representation of Interfaces

The *level set method* is a numerical technique for tracking interfaces and shapes [44]. In Subsection 1.2.2, we explained the interface problems and fixed Cartesian grids. The interface Γ is a set of curves for 2–D problems or a set of surfaces for 3–D problems. The level set method simplifies numerical computation without having to parameterize these objects. Specially, for moving interface cases, for modeling time-varying objects, it is easy to follow shapes which change topology by the level set method.

We use the zero level set

$$\Gamma = \{(x_1, x_2) | \varphi(x_1, x_2) = 0\} \tag{1.26}$$

of a Lipschitz continuous function, in particular, a good approximation of the signed distance function, to represent the interface Γ. For such a level set function, the interface can only cut the interface once between two grid lines. The resolution of the interface Γ is determined by the level set function. We note that there are plenty of discussions in the literature about how to construct the level set function for an interface Γ, see, e.g., [35, 38]. If the interface is complicated in reference to a mesh, then it may not be resolved well by the level set function. Consequently, the finite element solution obtained from our method may not resolve all fine details of the solution. A finer mesh is then needed to resolve the geometry and the solution.

1.3 A Brief Overview of the Methods for Interface Problems

Many numerical methods have been developed for interface problems and it is impossible to explain them all here. In first part of this section, we list some methods either because they illustrate the historical development, or because they represent the current state-of-the-art for solving interface problems. The second part of this section, we focus on reviewing the *immersed finite element methods* related to this thesis.

We now give a list of some previous methods for interface problems without descriptions:

- The smoothing method for discontinuous coefficients;
- Harmonic averaging for discontinuous coefficients;
- The ghost fluid method;
- Peskin's Immersed Boundary (IB) method;
- Immersed Interface Methods (IIM);
- The finite element method with body fitted grids;
- Mortar finite element methods;
- The immersed finite element method(IFEM).

We refer the reader to [46] for a short literature review.

The *immersed finite element method* was first presented by Z. Li, T. Lin and X. Wu in [30] for solving the elliptic interface problems with homogeneous jump conditions, i.e., $w = 0$ and $Q = 0$ on Γ in (1.3) and (1.4). The standard Cartesian

grids and uniform triangulations are applied. The interface Γ is allowed to cut through elements. The non-conforming and conforming immersed finite element spaces were constructed using piecewise linear basis functions which satisfy the jump conditions across the interface at least approximately. The first order accurate in L^∞ norm is achieved using the non-conforming finite element space, second order accurate is showed numerically using the conforming immersed finite element method.

The immersed finite element methods incorporate the finite-element discretization into the framework of the immersed-interface method for interface problems; and we therefore call them *immersed-interface finite-element methods* in this thesis. The basic idea of an immersed-interface finite-element method is to incorporate the jump conditions in constructing basis functions. In a finite-difference immersed-interface method, the jump conditions are enforced through finite-difference equations on grid points near the interface. In immersed-interface finite-element methods, the jump conditions are enforced through the construction of special finite-element basis functions that satisfy the homogeneous interface conditions. Clearly, such basis functions depend on the interface location and the jump w. Some of the related work can be found in [12, 19, 46].

1.4 Outline and Major Contributions of the Thesis

Our methods developed in this thesis can deal with the elliptic and elasticity interface problems with homogeneous or non-homogeneous jump conditions across the interface. It is strongly motivated by the need of a fast and accurate solver of interface

problems that is required in each time step in a long-time level-set simulation of interface dynamics without re-meshing. Such simulation has been a powerful numerical approach in understanding material properties, biological processes, and many other important phenomena in science and engineering [35, 38].

- **Elliptic interface problems with homogeneous jump conditions**

  If the jump conditions are homogeneous, i.e., $w = 0$ and $Q = 0$ on Γ in (1.3) and (1.4), then the problem (1.1)–(1.5) is equivalent to that of finding $u \in H^1(\Omega)$ with $u = g$ on $\partial\Omega$ such that

  $$\int_\Omega \beta \nabla u \cdot \nabla v \, dx = \int_\Omega f v \, dx \qquad \forall v \in H_0^1(\Omega).$$

  In Chapter 2, we develop a finite-element method based on Cartesian meshes to solve the elliptic interface problem (1.1)–(1.5) with homogeneous jump conditions. Such a method has the following distinguished features:

  (a) A fixed finite-element mesh is used that allows the interface to cut through edges of elements.

  (b) Special finite-element basis functions for nodal points near the interface are constructed to satisfy the homogeneous jump conditions.

  (c) The resulting linear system of discretization is symmetric positive definite.

  (d) Optimal convergence rates, same as those for a body-fitted finite-element method, are achieved.

In [30], a conforming IFEM has been developed for homogeneous jump conditions using piecewise linear basis functions. Those elements in which the interface cuts through are divided into three pieces according to an angle condition. This makes the discussion and the algorithm a little bit complicated. In Chapter 2, we propose piecewise quadratic basis functions that may be simpler and may have other advantages.

The finite element spaces constructed in Chapter 2 are conforming ones that contain piecewise polynomials. The interpolation error is therefore second order accuracy. If the solution to the elliptic interface problem is piecewise smooth, or more precisely, $u^{\pm} \in C^2(\overline{\Omega^{\pm}})$, which is true for many applications, then the standard convergence results are true for our finite element methods. This means that our finite element solution is first order accurate in the H^1 norm, and second order accurate in the L^2 norm. In Chapter 2, while we only construct some two-dimensional, linear and quadratic, immersed interface finite elements, our framework described here can be used for high-order, immersed-interface finite elements and for more general elliptic interface problems in two and three space dimensions.

- **Elliptic interface problems with non-homogeneous jump conditions**

  In Chapter 3 of this thesis, we present a new immersed -interface finite-element method for the elliptic interface problems with a special treatment of non-homogeneous jump conditions: $w \neq 0$ and $Q \neq 0$ on Γ in (1.3) and

(1.4). In Chapter 2, the immersed interface finite element space we defined only approximately satisfy the homogeneous jump conditions, i.e., $w = 0$ and $Q = 0$ on Γ. In order to continue to apply the same immersed interface finite element space, we need to have a way to remove the source singularities.

The idea is to construct a function with the *same non-homogeneous jump conditions* by using a level set function [31]. We transform the interface problem to a new one with homogeneous jump conditions, then we apply the immersed interface finite element method. The modifications to the stiffness matrix and the load vector are only needed in those elements that the interface cuts through.

- **Elasticity interface problems with homogeneous jump conditions**

  In Chapter 4 , we apply the immersed interface finite element methods for elasticity interface problems with homogeneous jump conditions: $W = (w_1, w_2) = (0, 0)$ on Γ in (1.21) and (1.22), $Q = (q_1, q_2) = (0, 0)$ on Γ in (1.19) and (1.20).

  In [46], a non-conforming immersed finite element method has been developed for elasticity interface problems with homogeneous jump conditions. The convergence rate of this method is first order, but there are no error estimates and grid refinement analysis for all numerical examples provided in [46]. In Chapter 4, we construct the non-conforming and conforming immersed interface finite element spaces and obtain their corresponding interpolation errors. We also compare the interpolation errors, the global

errors of the finite element solutions between the non-conforming and conforming immersed interface finite element methods. Second order convergence rate is achieved by using the conforming immersed interface finite element method. The resulting linear system of discretization is still symmetric positive definite.

- **Elasticity interface problems with non-homogeneous jump conditions**

  In Chapter 5, we consider the elasticity interface problems with non-homogeneous jump conditions: $W = (w_1, w_2) \neq (0, 0)$ on Γ in (1.21) and (1.22), $Q = (q_1, q_2) \neq (0, 0)$ on Γ in (1.19) and (1.20). To construct a function with the *same non-homogeneous jump conditions* is more difficult than that for a single equation because four equations (1.21), (1.22), (1.19) and (1.20) are all coupled together.

  Again, we extend the jump conditions along the normal directions of the interface Γ, then use the immersed-interface finite-element methods discussed in previous chapters to solve the elasticity interface problems with non-homogeneous jump conditions without losing the convergence rate. We also present some numerical examples.

- **Conclusions**

  In Chapter 6, we draw conclusions.

Chapter 2

IIFEM for Elliptic Interface Problems with Homogeneous Jump Conditions

In this chapter, we develop a finite-element method based on Cartesian meshes to solve the elliptic interface problem (1.1)–(1.5) with homogeneous jump conditions, i.e., $w = 0$ and $Q = 0$ on Γ in (1.3) and (1.4).

The conforming, piecewise quadratic, immersed interface finite element spaces are constructed in this chapter. The interpolation is therefore second order accurate. If the solution to the elliptic interface problem is piecewise smooth, or more precisely, $u^{\pm} \in C^2(\overline{\Omega^{\pm}})$, which is true for many applications, then the standard convergence results are true for our finite element methods. This means that our finite element solution is first order accurate in the H^1 norm, and second order accurate in the L^2 norm.

Our framework described here can be used for high-order, immersed-interface finite elements and for more general elliptic interface problems in two and three space dimensions.

2.1 The Weak Formulation

By multiplying a *test function* $v \in H_0^1(\Omega)$ by the self-adjoint equation (1.1) and (1.2) and integrating both sides, we have

$$\int_\Omega (-\nabla \cdot (\beta u))v \, dx = \int_\Omega fv \, dx. \tag{2.1}$$

By using the formula of integration by parts (*Green's first identity*), we get

$$\int_\Omega (-\nabla \cdot (\beta u))v \, dx = \int_\Omega \beta \nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} \beta v \frac{\partial u}{\partial n} \, ds, \tag{2.2}$$

and put (2.2) into (2.1), the weak form of (1.1) and (1.2) is

$$\int_\Omega \beta \nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} \beta v \frac{\partial u}{\partial n} \, ds = \int_\Omega fv \, dx. \tag{2.3}$$

Since $f \in L^2(\Omega)$, we can get

$$\int_\Omega fv \, dx = \int_{\Omega^+} fv \, dx + \int_{\Omega^-} fv \, dx, \tag{2.4}$$

we split (2.3) into two equations

$$\int_{\Omega^+} \beta \nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} \beta v \frac{\partial u}{\partial n} \, ds - \int_\Gamma \beta^+ v^+ \frac{\partial u^+}{\partial n^+} \, ds = \int_{\Omega^+} fv \, dx, \tag{2.5}$$

$$\int_{\Omega^-} \beta \nabla u \cdot \nabla v \, dx - \int_\Gamma \beta^- v^- \frac{\partial u^-}{\partial n^-} \, ds = \int_{\Omega^-} fv \, dx, \tag{2.6}$$

where n^+ and n^- are unit normal direction of interface Γ pointing inward and outward. There exits a relation along the interface Γ :

$$-n^+ = n^- = n. \tag{2.7}$$

By using $v \in H_0^1(\Omega)$ condition to (2.5) and summarizing (2.5) and (2.6), we get

$$\int_\Omega \beta \nabla u \cdot \nabla v \, dx + \int_\Gamma \beta^+ v^+ \frac{\partial u^+}{\partial n} \, ds - \int_\Gamma \beta^- v^- \frac{\partial u^-}{\partial n} \, ds = \int_\Omega f v \, dx. \tag{2.8}$$

We combine (1.4) and (2.8) to get the weak form for the interface problem

$$\int_\Omega \beta \nabla u \cdot \nabla v \, dx = \int_\Omega f v \, dx - \int_\Gamma Q v \, ds \qquad \forall v \in H_0^1(\Omega). \tag{2.9}$$

In [6], for homogeneous jump conditions, (2.9) becomes

$$\int_\Omega \beta \nabla u \cdot \nabla v \, dx = \int_\Omega f v \, dx \qquad \forall v \in H_0^1(\Omega). \tag{2.10}$$

2.2 The Immersed-Interface Finite-Element Space

Let $\mathcal{T}_h$ be a finite-element mesh with mesh size h that covers $\overline{\Omega}$. We assume that the elements in $\mathcal{T}_h$ are all triangles. For simplicity, we shall assume that the boundary $\partial\Omega$ consists of line segments so that the mesh covers $\overline{\Omega}$ exactly. Standard finite-element techniques can be applied to treat a curved boundary without affecting our approximation properties. We remark that in practice the computational domain Ω can often be chosen as a rectangular domain with sides parallel to the coordinate

axes; and the finite-element mesh can be uniform.

We call an element $T \in \mathcal{T}_h$ an *interface element*, if $\Gamma \cap \operatorname{int} T \neq \emptyset$. Note that an element is a *non-interface element*, if one of its edges is part of the interface. We assume for any interface element $T \in \mathcal{T}_h$, the set $\Gamma \cap \partial T$ consists of exactly two points that are on different edges of T.

We define an immersed-interface finite-element space V_h with respect to the mesh $\mathcal{T}_h$ to be a finitely-dimensional subspace of $L^2(\Omega)$. To be more precise, we specially define a *conforming, immersed-interface finite-element space* V_h for the approximation of a second-order elliptic interface problem to be a subspace of $H_0^1(\Omega)$ in this chapter.

V_h consists of all the linear combinations of the corresponding basis functions $\phi_1, \ldots, \phi_N$ for some integer $N \geq 1$:

$$V_h = \operatorname{span}\{\phi_1, \ldots, \phi_N\}. \tag{2.11}$$

The basis functions $\phi_1, \ldots, \phi_N$ are the usual finite-element basis functions on a non-interface element, and are piecewise polynomials on each interface element that is determined by the element and the interface Γ. All the basis functions, and hence all the functions in V_h, satisfy the homogeneous jump conditions for both of the function and flux

$$[v_h]_{\Gamma_h} = 0 \quad \text{and} \quad \left[\beta \frac{\partial v_h}{\partial n}\right]_{\Gamma_h} = 0 \qquad \forall v_h \in V_h, \tag{2.12}$$

where Γ_h denotes the union of line segments approximating the interface Γ.

The *interpolation function* u_I and the *finite element solution* u_h are both in V_h.

In the mathematical subfield of numerical analysis, *interpolation* is a method of constructing new data points from a discrete set of known data points [43]. In this chapter, the interpolation function u_I is given as

$$u_I(x) = \sum_{i=1}^{N} u(x^{(i)})\phi_i(x), \tag{2.13}$$

where $u(x^{(i)})$ is the exact value of u at the i^{th} nodal point (N_i) and the finite element solution u_h can be written as

$$u_h(x) = \sum_{i=1}^{N} u_i\phi_i(x), \tag{2.14}$$

where u_i is the finite element solution at the i^{th} nodal point (N_i). u_h is the projection of u onto the *conforming, immersed-interface finite-element space* V_h in term of energy inner product. The u_h is best approximation to the exact solution u on V_h in the energy norm.

2.3 Construction of Basis Functions

This section includes three subsections: basis functions for non-interface elements, a non-conforming immersed interface finite element space and a conforming quadratic finite element space.

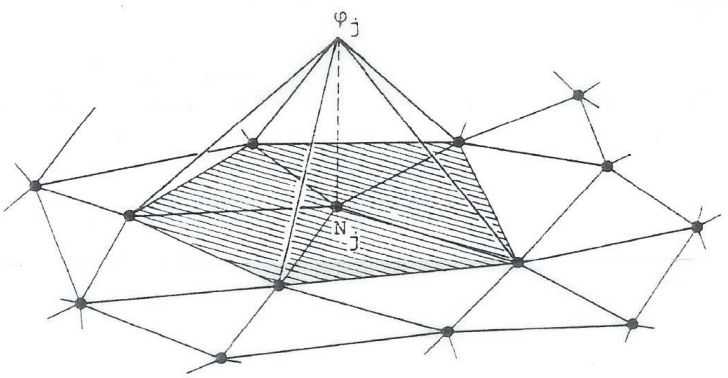

Figure 2.1: A global basis function ϕ_j for regular node N_j.

2.3.1 Basis Functions for Non-interface Elements

We now review general conforming, linear finite element basis function [6]. In this thesis, let $N_1, \ldots, N_N$ be all the vertices of triangular elements in the mesh $\mathcal{T}_h$. we call a vertex an *irregular vertex* or *irregular node*, if it is a vertex of an interface element; and *regular vertex* or *regular node*, otherwise. For a regular node N_i, we define ϕ_i to be the general conforming, linear finite element basis function associated with N_i with

$$\phi_i(N_j) = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j, \end{cases} \tag{2.15}$$

see Figure 2.1 for an illustration. The support of ϕ_i is the union of the triangles surrounding the node N_i.

In this thesis, we transform all triangles Ω_e: $(x_1^{(1)}, x_2^{(1)})$, $(x_1^{(2)}, x_2^{(2)})$, $(x_1^{(3)}, x_2^{(3)})$ to master element: (0,0),(1,0),(0,1); see Figure 2.2. There are only three non-zero basis functions over this master element. The non-zero local basis functions can be

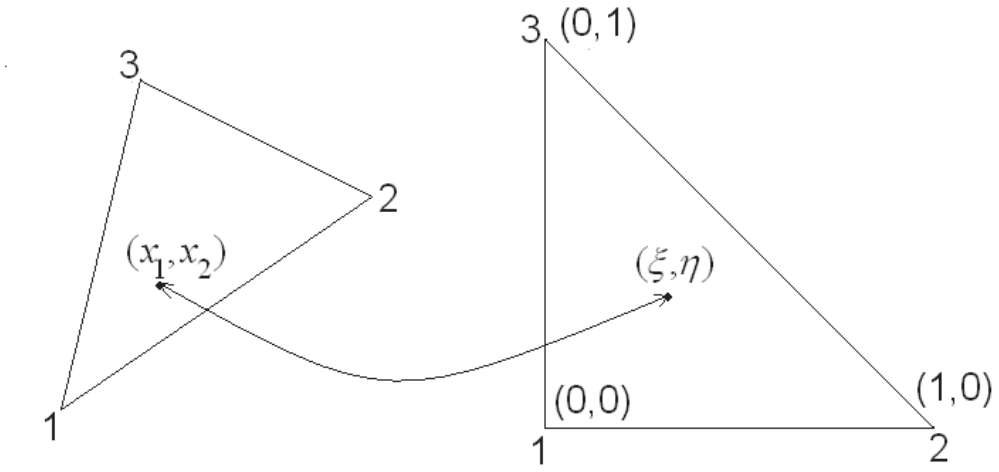

Figure 2.2: Linear transformation from triangle Ω_e to master element.

expressed as

$$\phi_1(\xi,\eta) = 1 - \xi - \eta, \tag{2.16}$$

$$\phi_2(\xi,\eta) = \xi, \tag{2.17}$$

$$\phi_3(\xi,\eta) = \eta. \tag{2.18}$$

The relation between (x_1, x_2) and (ξ, η) is

$$x_1 = x_1^{(1)}\phi_1(\xi,\eta) + x_1^{(2)}\phi_2(\xi,\eta) + x_1^{(3)}\phi_3(\xi,\eta), \tag{2.19}$$

$$x_2 = x_2^{(1)}\phi_1(\xi,\eta) + x_2^{(2)}\phi_2(\xi,\eta) + x_2^{(3)}\phi_3(\xi,\eta), \tag{2.20}$$

$$\xi = \frac{1}{2A_e}\{(x_2^{(3)} - x_2^{(1)})(x_1 - x_1^{(1)}) - (x_1^{(3)} - x_1^{(1)})(x_2 - x_2^{(1)})\}, \tag{2.21}$$

$$\eta = \frac{1}{2A_e}\{-(x_2^{(2)} - x_2^{(1)})(x_1 - x_1^{(1)}) + (x_1^{(2)} - x_1^{(1)})(x_2 - x_2^{(1)})\}, \tag{2.22}$$

where A_e is the area of Ω_e.

2.3.2 A Non-conforming Immersed Interface Finite Element Space

We now review some old nonconforming, immersed-interface finite-element basis functions that satisfy the homogeneous jump conditions which was first constructed in [30]; see also [19]. We briefly review it, since it will be used to construct our new linear and nearly linear elements.

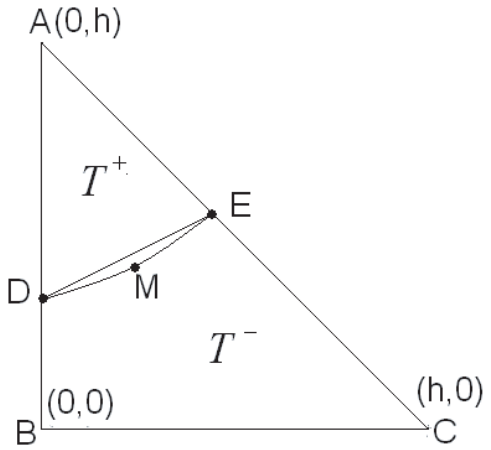

Figure 2.3: A typical interface element $T = \triangle ABC$. The arc DME is the part of the interface Γ in T. It is approximated by the line segment $\overline{DE}$. $T^+ = \triangle ADE$, $T^- = T - T^+$, and T_r is the region enclosed by the $\overline{DE}$ and the arc DME.

Fix an interface element $T \in \mathcal{T}_h$, cf. Figure 2.3. As in the common practice, we approximate the interface in T, $\Gamma \cap T$, by a line segment connecting the intersections of the interface and the edges of the triangle T. This line segment is $\overline{DE}$ in Figure 2.3. The line segment divides T into two parts T^+ and T^- : one triangular and the other

quadrilateral. Note that there is a small region T_r in T,

$$T_r = T - \Omega^+ \cap T^+ - \Omega^- \cap T^-,$$

whose area is of order $O(h^3)$.

We now consider the typical interface element T in which the curve between points D and E is part of the interface in Figure 2.3 as a reference interface element T and define local basis function for each vertex of T. The local basis function for a general interface element in the mesh $\mathcal{T}_h$ can be defined through the usual affine transformation. We denote

$$T = \{(x_1, x_2) : 0 \le x_1 \le h, \quad 0 \le x_2 \le h, \quad x_1 + x_2 \le h\},$$

and assume that the coordinates at A, B, C, D, and E are

$$(0, h), \quad (0, 0), \quad (h, 0), \quad (0, x_2^D), \quad (h - x_2^E, x_2^E),$$

respectively; cf. Figure 2.3. In the practice, we take $h = 1$ in this thesis.

Each of the three local basis functions corresponding to the nodes A, B, or C is piecewise linear function

$$\phi(x) = \begin{cases} \phi^+(x) = a_0 + a_1 x_1 + a_2(x_2 - h) & \text{if } x = (x_1, x_2) \in T^+, \\ \phi^-(x) = b_0 + b_1 x_1 + b_2 x_2 & \text{if } x = (x_1, x_2) \in T^-. \end{cases} \tag{2.23}$$

There are six unknown coefficients a_i and b_i $(i = 0, 1, 2)$. They are determined by the

conditions (cf. [30]):

$$\phi_i(N_j) = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j, \end{cases} \tag{2.24}$$

$$\phi^+(D) = \phi^-(D), \tag{2.25}$$

$$\phi^+(E) = \phi^-(E), \tag{2.26}$$

$$\beta^+ \frac{\partial \phi}{\partial n}^+ = \beta^- \frac{\partial \phi}{\partial n}^-, \tag{2.27}$$

where n is the unit normal direction of the line segment $\overline{DE}$. It is shown in [30] that this function ϕ exits and is uniquely determined. Note that basis functions defined in this way can be discontinuous across edges of interface elements; cf. Figure 2.4. So, this defines a nonconforming finite element.

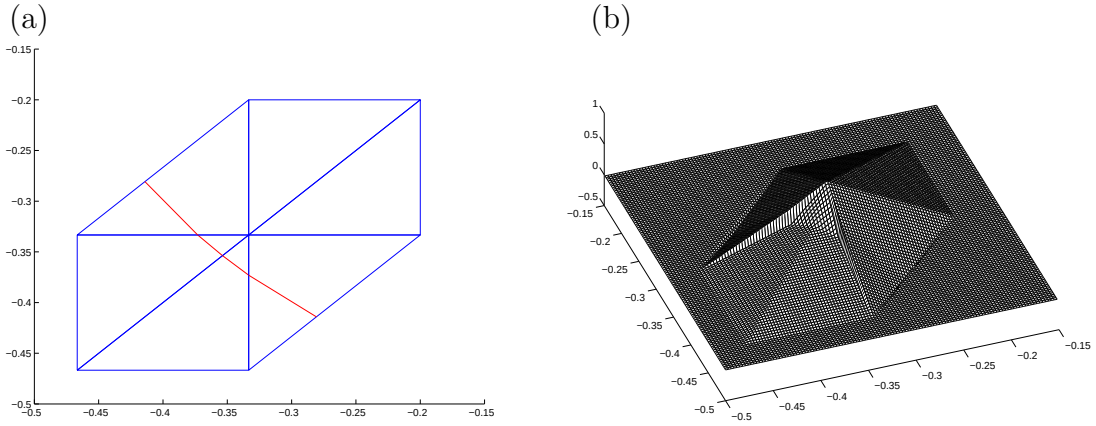

Figure 2.4: (a) A standard domain of six triangles with an interface cutting through. (b) A global basis function on its support in the non-conforming immersed interface finite element space. The basis function has small jump across some edges.

Once the basis functions are defined, the *Galerkin finite element method* then is applied to get the finite element solution. This method is called the *Non-conforming immersed interface finite element method*. This method is simple, easy to implement, and has an algebraic system similar to that of the Galerkin finite element method based on the standard finite element space.

2.3.3 A Conforming Quadratic Finite Element Space

We now construct a conforming, quadratic element. As before, the basis functions associated with regular nodes are the standard conforming linear finite-element basis functions. But the basis functions associated with irregular nodes are piecewise quadratic. All the basis functions are globally continuous. The non-conforming basis functions are continuous in each element, but may be discontinuous across edges of interface elements. To maintain continuity of the basis function, one strategy is to extend the basis functions along the interface to one more element, then average the values and the tangential derivatives along edges.

(a) (b)

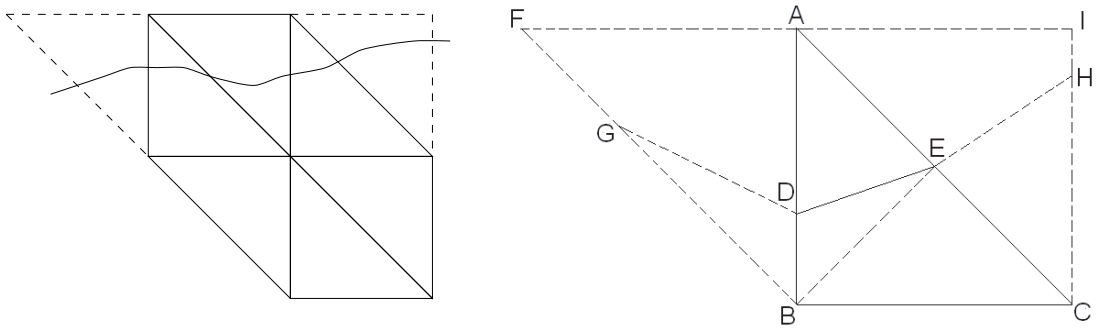

Figure 2.5: (*a*) The support of a local basis function. (*b*) A diagram for the construction of a local basis function on $\triangle ABC$.

Let $T = \Delta ABC$ be an interface element; cf. Figure 2.5 (b). As before, we assume that the interface meets edges of this element at D and E. To construct a local basis function that is globally continuous, we extend the previously defined basis function at the same node (vertex) to one more triangle along the interface (cf. Figure 2.5 (a)). We require the local basis functions in two adjacent interface elements, such as ΔABC and ΔAFB, take the same value at the interface point on their common edge, such as the point D. This will achieve the global continuity of a basis function associated with an irregular node.

The construction of the piecewise quadratic basis functions includes the following steps :

P1. Use (2.24), (2.25), (2.26) and (2.27) to construct the three nonconforming linear finite-element basis functions defined on the elements ΔABC, ΔAFB, and ΔACI, respectively;

P2. Set the value at D as the average of the values at D of the nonconforming linear finite-element basis functions defined ΔABC and ΔAFB constructed in **P1**. Assign the tangential derivative at D along AD to be the average of the values of the tangential derivatives (along AD) of the nonconforming linear finite-element basis functions on ΔABC and ΔAFB constructed in **P1**. Similarly, assign the tangential derivative at D along DB as the average of the values of the tangential derivatives (along DB) of the nonconforming finite-element basis functions on ΔABC and ΔAFB constructed in **P1**;

P3. Repeat to set the value at E as the average of values at E of the nonconforming linear finite-element basis functions defined on the elements ΔABC and ΔACI in **P1**. Also assign the tangential derivatives along AE and EC as the average of those from the nonconforming finite-element basis functions along the same edge;

P4. Assign the tangential derivatives along BC, DE and BE exactly the same (no average) as those from the nonconforming finite-element basis functions on ΔABC;

P5. Set the values of at the points A, B and C exactly the same as those from the nonconforming finite-element basis function on ΔABC;

P6. Partition the element ΔABC into three sub-triangles ΔADE, ΔDBE and ΔBCE by the line segment $\overline{BE}$ and $\overline{DE}$; see Figure 2.6.

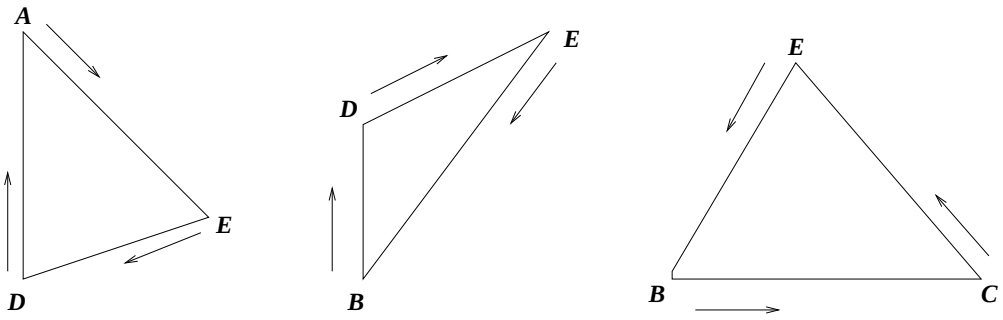

Figure 2.6: A diagram of three triangles on which piecewise continuous quadratic functions can be determined. The symbol "$\longrightarrow$" indicates a tangential derivative.

P7. Define the basis function ϕ_i to be the piecewise quadratic function in the

three sub-triangles

$$\phi(x) = \begin{cases} c_1 + c_2x_1 + c_3x_2 + c_4x_1^2 + c_5x_2^2 + c_6x_1x_2 & \text{if } (x_1, x_2) \in \Delta ADE, \\ c_7 + c_8x_1 + c_9x_2 + c_{10}x_1^2 + c_{11}x_2^2 + c_{12}x_1x_2 & \text{if } (x_1, x_2) \in \Delta DBE, \\ c_{13} + c_{14}x_1 + c_{15}x_2 + c_{16}x_1^2 + c_{17}x_2^2 + c_{18}x_1x_2 & \text{if } (x_1, x_2) \in \Delta BCE. \end{cases} \tag{2.28}$$

All 18 unknown coefficients are determined by the values at the points A, B, C, D, and E, respectively, and the tangential derivatives from each side of the triangles; see Figure 2.6 for an illustration.

We now show that a quadratic function on a triangle is uniquely determined by its values at the three vertices and its tangential derivatives along the three sides at some particular points.

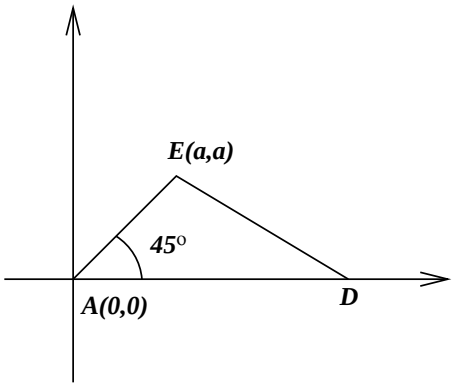

Figure 2.7: A diagram in determining of the quadratic interpolation function.

Lemma 2.1. *Referring to the geometry in Figure 2.7, there is a unique quadratic polynomial*

$$I_h v(x_1, x_2) = v_A + a_{10}x_1 + a_{01}x_2 + a_{20}x_1^2 + a_{11}x_1x_2 + a_{02}x_2^2, \tag{2.29}$$

that interpolates the values of v_A, v_D and v_E, and three tangential derivatives v'_{AE}, v'_{AD} and v'_{ED} defined at the three vertices A, E and D respectively.

Proof: Using the undetermined coefficient method, we find the unique expression of the coefficients

$$\begin{aligned} a_{10} &= \frac{\tilde{h}v'_{AD} + 2v_D - 2v_A}{\tilde{h}}, \\ a_{01} &= -\frac{\tilde{h}v'_{AD} - \sqrt{2}v'_{AE}\tilde{h} + 2v_D - 2v_A}{\tilde{h}}, \\ a_{20} &= -\frac{\tilde{h}v'_{AD} + v_D - v_A}{\tilde{h}^2}, \end{aligned}$$

$$\begin{aligned} a_{11} = -\frac{-2\tilde{h}\alpha v'_{AD} - 2\alpha v_D + 2\alpha v_A + \tilde{h}^2 v'_{AD} + 2\tilde{h}v_D}{\tilde{h}^2\alpha} \\ - \frac{\sqrt{\tilde{h}^2 - 2\tilde{h}\alpha + 2\alpha^2}v'_{ED}\tilde{h} + \tilde{h}\alpha\sqrt{2}v'_{AE} - 2\tilde{h}v_E}{\tilde{h}^2\alpha}, \end{aligned}$$

$$\begin{aligned} a_{02} = \frac{\tilde{h}^2\alpha v'_{AD} + 2\tilde{h}\alpha v_D - \tilde{h}^2\alpha\sqrt{2}v'_{AE} - \tilde{h}v'_{AD}\alpha^2 - \alpha^2 v_D + +\alpha^2 v_A}{\tilde{h}^2\alpha^2} \\ + \frac{v_E\tilde{h}^2 - v_A\tilde{h}^2 + \sqrt{\tilde{h}^2 - 2\tilde{h}\alpha + 2\alpha^2}v'_{ED}\tilde{h}\alpha + \tilde{h}\alpha^2\sqrt{2}v'_{AE} - 2v_E\tilde{h}\alpha}{\tilde{h}^2\alpha^2}, \end{aligned}$$

where $\tilde{h}$ is the distance from A to D, and the coordinates at E is (α, α) . **Q.E.D.**

By the above lemma, the all local piecewise quadratic basis functions are uniquely determined using the procedure **P1**–**P7**.

Because all the piecewise quadratic basis functions satisfy homogeneous jump conditions and are continuous and in H^1, this finite element space is conforming; see

(a) (b)

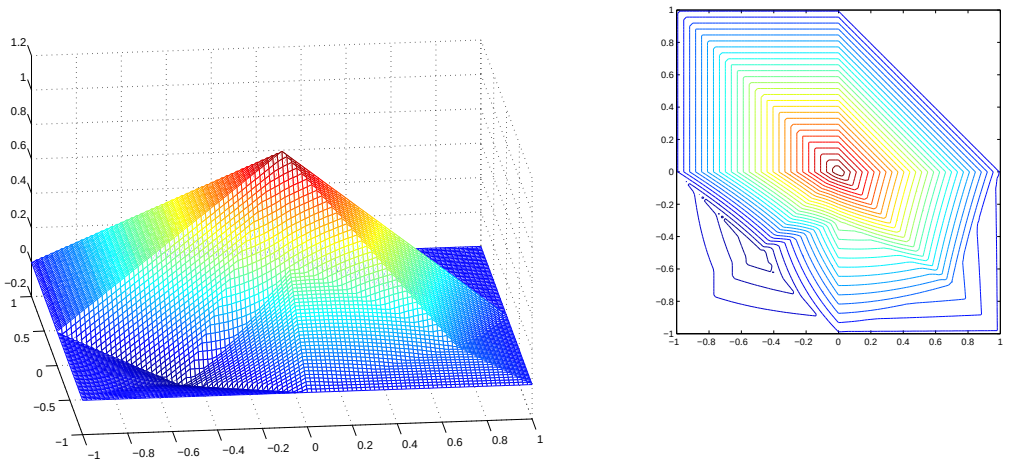

Figure 2.8: (*a*) A global piecewise quadratic basis function on its support in the conforming immersed interface finite element space. The basis function is continuous. (*b*) A contour plot of the basis function in (a).

Figure 2.8. Also, this conforming immersed interface finite element space has the same dimension as the non-conforming immersed interface finite element space and the standard linear finite element space defined on the partition $\mathcal{T}_h$.

Remark 1. Consider a uniform triangular mesh. For both non-conforming linear and conforming, quadratic elements, the basis functions ϕ_i associated with the i^{th} node (N_i) is non-zero only on the six surrounding triangular elements, if the interface does not cut through any of these triangles. Otherwise, the support of a basis function includes two more triangular elements along the direction of the interface, if the interface cuts through any of the surrounding triangle. A corresponding finite-difference scheme, however, has generally a non-standard nine point stencil.

Remark 2. If the coefficient β is continuous across the interface, i.e., $[\beta]_\Gamma = 0$, then both the linear and quadratic basis functions become the standard linear basis functions.

2.4 Assembling the Stiffness Matrix and Load Vector

In this chapter, we define $A = \{a_{ij}\}$ and $F = \{F_i\}$ as the *global stiffness matrix* and *global load vector*. A is a $N \times N$ matrix, F is a $N \times 1$ vector, here N is the total number of *nodal points* . Since the weak form for the single elliptic interface problems with homogeneous jump conditions is (2.10), the entries of the stiffness matrix and the load vector can be written as

$$a_{ij} = \int_{\Omega} \beta \nabla \phi_i \cdot \nabla \phi_j \, dx, \tag{2.30}$$

$$F_i = \int_{\Omega} f \phi_i \, dx. \tag{2.31}$$

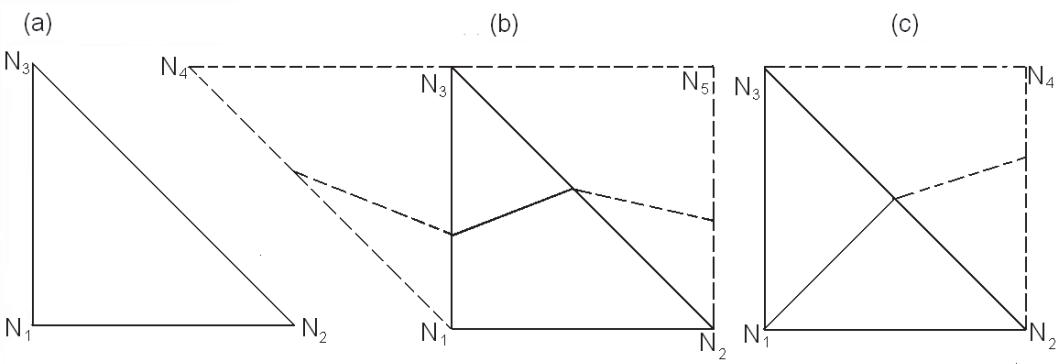

Figure 2.9: (a) A non-interface element, $\Omega_e = \Delta N_1 N_2 N_3$. (b) A interface element (Type 1), $\Omega_e = \Delta N_1 N_2 N_3$. (c) A interface element (Type 2), $\Omega_e = \Delta N_1 N_2 N_3$.

Let $\Omega_e = \Delta N_1 N_2 N_3$ be an element, on non-interface elements ; see Figure 2.9 (a), where the interface does not cut through, we can use the standard way in computing

the contribution to the stiffness matrix and the load vector. Since there are only three non-zero basis functions ϕ_1, ϕ_2, ϕ_3 which are associated with N_1, N_2, N_3 over this element $\Omega_e = \Delta N_1 N_2 N_3$, the *local stiffness matrix* A^{Ω_e} is a 3×3 matrix and the *local load vector* F^{Ω_e} is a 3×1 vector. A^{Ω_e} and F^{Ω_e} can be written as

$$A^{\Omega_e} = \begin{pmatrix} \int_{\Omega_e} \beta \nabla\phi_1 \cdot \nabla\phi_1 \, dx & \int_{\Omega_e} \beta \nabla\phi_1 \cdot \nabla\phi_2 \, dx & \int_{\Omega_e} \beta \nabla\phi_1 \cdot \nabla\phi_3 \, dx \\ \int_{\Omega_e} \beta \nabla\phi_2 \cdot \nabla\phi_1 \, dx & \int_{\Omega_e} \beta \nabla\phi_2 \cdot \nabla\phi_2 \, dx & \int_{\Omega_e} \beta \nabla\phi_2 \cdot \nabla\phi_3 \, dx \\ \int_{\Omega_e} \beta \nabla\phi_3 \cdot \nabla\phi_1 \, dx & \int_{\Omega_e} \beta \nabla\phi_3 \cdot \nabla\phi_2 \, dx & \int_{\Omega_e} \beta \nabla\phi_3 \cdot \nabla\phi_3 \, dx \end{pmatrix}, \tag{2.32}$$

$$F^{\Omega_e} = \begin{pmatrix} \int_{\Omega_e} f\phi_1 \, dx \\ \int_{\Omega_e} f\phi_2 \, dx \\ \int_{\Omega_e} f\phi_3 \, dx \end{pmatrix}. \tag{2.33}$$

If the interface cuts through two edges of the element, we define this kind of the interface elements as *type 1*, see Figure 2.9 (b). Since there are five non-zero basis functions $\phi_1, \phi_2, \phi_3, \phi_4, \phi_5$ which are associated with N_1, N_2, N_3, N_4, N_5 over this element $\Omega_e = \Delta N_1 N_2 N_3$, the *local stiffness matrix* A^{Ω_e} is a 5×5 matrix and the *local load vector* F^{Ω_e} is a 5×1 vector. $A^{\Omega_e} = \{a_{ij}^{\Omega_e}\}$ and $F^{\Omega_e} = \{F_i^{\Omega_e}\}$ are given by

$$a_{ij}^{\Omega_e} = \int_{\Omega_e} \beta \nabla\phi_i \cdot \nabla\phi_j \, dx, \qquad i, j = 1, 2, 3, 4, 5, \tag{2.34}$$

$$F_i^{\Omega_e} = \int_{\Omega_e} f\phi_i \, dx \qquad i = 1, 2, 3, 4, 5. \tag{2.35}$$

We call the interface element as *type 2 interface element* if the interface cuts through one edge and one vertex of the element, see Figure 2.9 (c). Since there are four non-zero basis functions $\phi_1, \phi_2, \phi_3, \phi_4$ which are associated with N_1, N_2, N_3, N_4

over this element $\Omega_e = \Delta N_1 N_2 N_3$, the *local stiffness matrix* A^{Ω_e} is a 4×4 matrix and the *local load vector* F^{Ω_e} is a 4×1 vector. $A^{\Omega_e} = \{a_{ij}^{\Omega_e}\}$ and $F^{\Omega_e} = \{F_i^{\Omega_e}\}$ are given by

$$a_{ij}^{\Omega_e} = \int_{\Omega_e} \beta \nabla \phi_i \cdot \nabla \phi_j \, dx, \qquad i, j = 1, 2, 3, 4, \tag{2.36}$$

$$F_i^{\Omega_e} = \int_{\Omega_e} f \phi_i \, dx \qquad i = 1, 2, 3, 4. \tag{2.37}$$

2.5 The Numerical Integration

In order to obtain the local stiffness matrix and the load vector, we need the numerical integration over the master triangle: $(0, 0)$, $(1, 0)$, $(0, 1)$; see Figure 2.2, and [3, 16, 33, 39].

The *numerical integration* is the process of using a set of approximate values of a function to calculate its integral to comparable accuracy [36, 40]. The basic problem we considered in this thesis is to compute an approximate solution to the definite integral: $\int_\Delta f(x_1, x_2) \, dx_1 dx_2$. We apply four points *Gaussian quadrature*:

$$\int_\Delta f(x_1, x_2) \, dx_1 dx_2 = \sum_{i=1}^{4} f(x_1^{(i)}, x_2^{(i)}) w_i, \tag{2.38}$$

where w_i's are called *weights*. All four points a, b, c and d are inner points; see Figure 2.10 for an illustration, the values of w_i's and the coordinates of four points: a, b, c and d are given in Table 2.1.

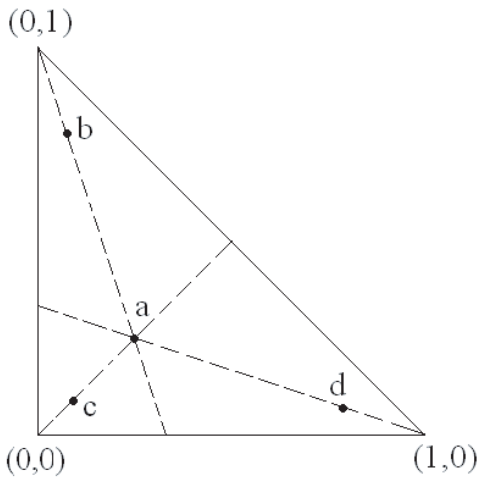

Figure 2.10: Gaussian points for integration over master triangle.

Table 2.1: Gaussian points $(x_1^{(i)}, x_2^{(i)})$ and weights w_i.

Point	i	$x_1^{(i)}$	$x_2^{(i)}$	w_i
a	1	1/3	1/3	-27/96
b	2	2/15	11/15	25/96
c	3	2/15	2/15	25/96
d	4	11/15	2/15	25/96

2.6 The Software Package: ITPACK 2C

After we obtain $A = \{a_{ij}\}$ and $F = \{F_i\}$, we can solve the linear equation system $AU = F$ to get the numerical solution of $U = \{u_i\}$ at all nodal points. In this thesis, we used the ITPACK 2C [23, 48] package to do it.

We have downloaded the ITPACK Software Package from [23]. It was developed at the Center for Numerical Analysis, the University of Texas at Austin. Below is a short description of the ITPACK 2C given in [23]:

> ITPACK is a collection of seven FORTRAN subroutines for solving large sparse linear systems by adaptive accelerated iterative algorithms. Basic iterative procedures, such as the Jacobi method, the Successive Overrelaxation method, the Symmetric Successive Overrelaxation method, and the RS method for the reduced system are combined, where possible, with acceleration procedures such as Chebyshev (Semi-Iteration) and Conjugate Gradient for rapid convergence. Automatic selection of the acceleration parameters and the use of accurate stopping criteria are major features of this software package. While the ITPACK routines can be called with any linear system containing positive diagonal elements, they are the most successful in solving systems with symmetric positive definite or mildly nonsymmetric coefficient matrices.

The stiffness matrix A from our $IIFEM$ is a sparse matrix. When storing sparse matrices, instead of storing all entries, the idea used by many linear solvers is to only store the non-zero entries. There exist some different data structures which avoid to waste the computer memory. In ITPACK, we use three vectors: $A(*)$, $IA(*)$ and

$JA(*)$ to store them. If the original coefficient matrix of the linear system is a $N \times N$ matrix, then IA is a $(N+1) \times 1$ vector. The size of $JA(*)$ is same as $A(*)$ determined by the total number of non-zero entries. We search the non-zero values row-wisely, store them in $A(*)$ one by one, and store their associated column numbers in $JA(*)$. For the symmetric matrix, we can save even more computer memory by storing only the nonzero entries in each row on and above the main diagonal. $JA(k)$ is the column number for the value $A(k)$ for either nonsymmetric or symmetric case. $IA(1)$ is equal to 1. $IA(i)$ indicates that the starting locations of i-th row of the original coefficient matrix of the linear system in $A(*)$. Clearly, the total non-zero entries in i-th row is given by $IA(i+1) - IA(i)$.

Next, we give an example to show the storage scheme described above. The coefficient matrix

$$\begin{pmatrix} 1 & 0 & 11 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 11 & 0 & 3 & 0 & 0 & 0 & 9 & 0 \\ 0 & 0 & 0 & 4 & 12 & 0 & 0 & 0 \\ 0 & 0 & 0 & 12 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6 & 0 & 10 \\ 0 & 0 & 9 & 0 & 0 & 0 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 & 8 \end{pmatrix}$$

would be represented in nonsymmetric sparse storage as

$$\begin{aligned}
\mathbf{A}(*) &= (1, 11, 2, 11, 3, 9, 4, 12, 12, 5, 6, 10, 9, 7, 10, 8)^T, \\
\mathbf{JA}(*) &= (1, 3, 2, 1, 3, 7, 4, 5, 4, 5, 6, 8, 3, 7, 6, 8)^T, \\
\mathbf{IA}(*) &= (1, 3, 4, 7, 9, 11, 13, 15, 17)^T,
\end{aligned}$$

and in symmetric sparse storage as

$$\begin{aligned}
\mathbf{A}(*) &= (1, 11, 2, 3, 9, 4, 12, 5, 6, 10, 7, 8)^T, \\
\mathbf{JA}(*) &= (1, 3, 2, 3, 7, 4, 5, 5, 6, 8, 7, 8)^T, \\
\mathbf{IA}(*) &= (1, 3, 4, 6, 8, 9, 11, 12, 13)^T.
\end{aligned}$$

2.7 Numerical Results

We report one example of numerical calculations using our conforming, quadratic, immersed-interface finite-element method.

Example 2.1. *Elliptic interface problem with homogeneous jump conditions.*

This example is from [30]. We consider the problem (1.1)–(1.5) with $\Omega = (-1, 1) \times (-1, 1)$, Γ being the circle centered at point $(0, 0)$ with radius $R = 0.5$, and $\beta^- = 1$ and $\beta^+ = 100$. The source term $f(x_1, x_2)$ and the Dirichlet boundary data $g(x_1, x_2)$

Table 2.2: A grid refinement analysis for Example 2.1. The first column is the number of grid lines in x_1 andx_2 directions. The second column is the finite-element solution errors in L^∞ . The third column is the convergence rate calculated from (2.39).

m	$\|u - u_h\|_\infty$	Order
32	1.3142×10^{-3}	
64	4.2148×10^{-4}	1.64
128	1.0075×10^{-4}	2.06
256	2.7291×10^{-5}	1.88
512	7.6973×10^{-6}	2.03

are calculated from the exact solution $u(x_1, x_2)$:

$$u(x_1, x_2) = \begin{cases} \dfrac{r^3}{\beta^-} & \text{if } r \le R, \\ \dfrac{r^3}{\beta^+} + \left(\dfrac{1}{\beta^-} - \dfrac{1}{\beta^+}\right) R^3 & \text{otherwise,} \end{cases}$$

where $r = \sqrt{x_1^2 + x_2^2}$. The exact solution satisfies the homogeneous jump conditions.

In Table 2.2, we show a grid refinement analysis of our calculations. The first column $E_N = \|u - u_h\|_\infty$ is the error of the finite-element solution measured in the L^∞-norm. The second column is the estimate of the order of accuracy using the formula

$$\text{order} = \frac{\log\left(\|E_N\|_\infty / \|E_{2N}\|_\infty\right)}{\log 2}. \tag{2.39}$$

We see clearly the second order accuracy. Figure 2.11 shows two plots : the finite element solution u_h with $m = 64$ and the error $(u - u_h)$ with $m = 64$.

In Table 2.3, The second to the last columns are the interpolation errors. We see that the interpolation errors are second order and derivatives are of first order. These

(a) (b)

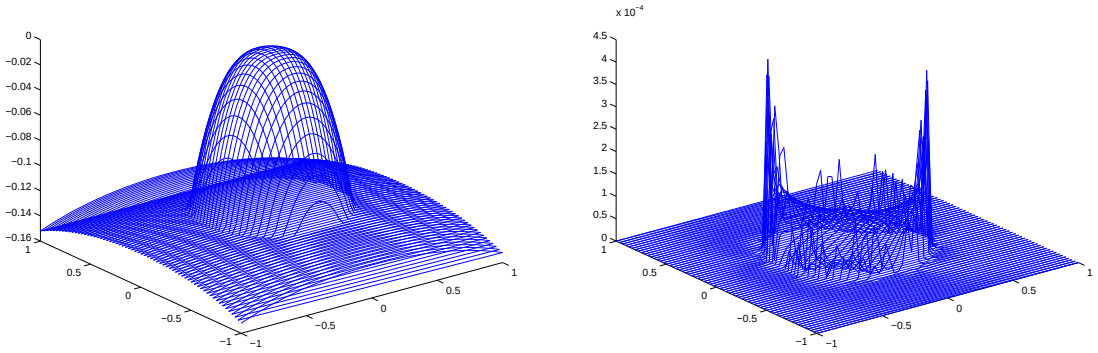

Figure 2.11: (a) The plot of u_h for Example 2.1 ($m = 64$). (b) The plot of $(u - u_h)$ for Example 2.1 ($m = 64$).

Table 2.3: A grid refinement analysis of the interpolation errors for Example 2.1.

m	$\|u - u_I\|_\infty$	Order	$\|\frac{\partial u}{\partial x_1} - \frac{\partial u_I}{\partial x_1}\|_{\Omega',\infty}$	Order	$\|\frac{\partial u}{\partial x_2} - \frac{\partial u_I}{\partial x_2}\|_{\Omega',\infty}$	Order
32	2.5895×10^{-3}		1.0559×10^{-1}		1.0427×10^{-1}	
64	6.7985×10^{-4}	1.93	5.3514×10^{-2}	0.98	5.3514×10^{-2}	0.96
128	1.7803×10^{-4}	1.93	2.7544×10^{-2}	0.96	2.7544×10^{-2}	0.96
256	4.5013×10^{-5}	1.99	1.3912×10^{-2}	0.99	1.3928×10^{-2}	0.99
512	1.1379×10^{-5}	1.99	6.9986×10^{-3}	0.99	6.9986×10^{-3}	0.99

are optimal.

In Table 2.4, we show the results and comparison of the finite-element solution in L^∞, L^2, and H^1 norms. We see second order convergence in L^∞, L^2 norms, and first order in H^1 norm as expected.

Table 2.4: A grid refinement analysis of the finite-element solution errors in L^∞, L^2, and H^1 norms for Example 2.1, where $\Omega' = \Omega \setminus \sum T_r$.

m	$\|u - u_h\|_\infty$	Order	$\|u - u_h\|_{L^2}$	Order	$\|u - u_h\|_{H^1}$	Order
32	1.3142×10^{-3}		6.4996×10^{-4}		5.7773×10^{-2}	
64	4.2148×10^{-4}	1.64	1.5968×10^{-4}	2.03	2.6610×10^{-2}	1.12
128	1.0075×10^{-4}	2.06	4.0006×10^{-5}	2.00	1.3453×10^{-2}	0.99
256	2.7291×10^{-5}	1.88	9.8992×10^{-6}	2.01	6.5930×10^{-3}	1.03
512	6.6973×10^{-6}	2.03	2.4888×10^{-6}	1.99	3.2891×10^{-3}	1.00

Chapter 3

IIFEM for Elliptic Interface Problems with Non-homogeneous Jump Conditions

In this chapter, we present the new immersed-interface finite-element method for the elliptic interface problems with a special treatment of non-homogeneous jump conditions: $w \neq 0$ and $Q \neq 0$ on Γ in (1.3) and (1.4).

The idea is to construct a function with the *same non-homogeneous jump conditions* by using a level set method [31]. We transform the interface problem to a new one with homogeneous jump conditions, then we apply the conforming, quadratic, immersed interface finite element space constructed in Chapter 2. The modification is only needed in those elements that the interface cuts through. This method maintains second order accuracy.

3.1 The Signed Distance Function

In this section, We first review the level-set representation of the interface Γ which is assumed to be smooth. Let $\varphi : \Omega \to \mathbb{R}$ be a function that satisfies

$$\varphi(x) \begin{cases} < 0 & \text{if } x \in \Omega^-, \\ = 0 & \text{if } x \in \Gamma, \\ > 0 & \text{if } x \in \Omega^+. \end{cases} \tag{3.1}$$

We call such a function $\varphi : \Omega \to \mathbb{R}$ a *level-set function* that represents the interface Γ. For $\rho > 0$, we denote the ρ-neighborhood of Γ in Ω by

$$N(\Gamma, \rho) = \{x \in \Omega : \text{dist}\,(x, \Gamma) < \rho\}, \tag{3.2}$$

where $\text{dist}\,(x, \Gamma)$ is the distance from x to Γ. We assume that there exists $\rho_0 > 0$ such that $\overline{N(\Gamma, \rho_0)} \subset \Omega$, and

$$\varphi \text{ is smooth and } |\nabla\varphi| > 0 \text{ in } N(\Gamma, \rho_0). \tag{3.3}$$

We note that the unit normal n to Γ, pointing from Ω^- to Ω^+, is given by

$$n = \frac{\nabla\varphi}{|\nabla\varphi|}. \tag{3.4}$$

The signed distance function

$$\varphi(x) = \begin{cases} -\operatorname{dist}(x,\Gamma) & \text{if } x \in \Omega^-, \\ 0 & \text{if } x \in \Gamma, \\ +\operatorname{dist}(x,\Gamma) & \text{if } x \in \Omega^+, \end{cases} \tag{3.5}$$

is a typical level-set function that satisfies our assumptions.

We now turn to the description of our treatment of the elliptic interface problem with non-homogeneous jump conditions. We first need the following lemma, which is extracted from [14]:

Lemma 3.1. *Let $\rho > 0$ be small enough. Then, for any $x \in N(\Gamma, \rho)$, there exists a unique $x^* \in \Gamma$ such that*

$$|x - x^*| = dist(x, \Gamma). \tag{3.6}$$

Moreover,

$$\frac{x - x^*}{|x - x^*|} = \begin{cases} -n(x^*) & \text{if } x \in \Omega^-, \\ +n(x^*) & \text{if } x \in \Omega^+, \end{cases} \tag{3.7}$$

where $n(x^*)$ is the unit normal to Γ at x^*, pointing from Ω^- to Ω^+.

Proof. Without loss of generality, let us assume that, in a local Cartesian coordinate system, $x = (0, 0)$ is the origin, and the interface nearby is a graph of a smooth function $\eta = \eta(s)$ that is nonzero for any s in a certain range. The distance from the

origin to any point $(s, \eta(s))$ on the interface is then given by $\sqrt{I(s)}$ with

$$I(s) = s^2 + [\eta(s)]^2.$$

Setting $I'(s) = 0$, we get $\eta'(s) = -s/\eta(s)$. This implies (3.7), since $(-1, \eta'(s))$ is parallel to the normal at $(s, \eta(s))$. Moreover,

$$I''(s) = 2 + 2[\eta'(s)]^2 + 2\eta(s)\eta''(s)$$

is positive, since $I(s)$, and hence $\eta(s)$, is small enough. Thus, there exists a unique s that minimize $I(s)$. This implies the existence and uniqueness of x^* that satisfies (3.6). **Q.E.D.**

In what follows, we fix $\rho \in \mathbb{R}$ as in the above lemma and assume that $0 < \rho < \rho_0$, where ρ_0 is the same as in (3.3).

3.2 The Numerical Extensions of Jumps w and Q

In this section, we give details about the numerical extension of the jumps w and Q. We define $w^e : \Omega \to \mathbb{R}$ and $Q^e : \Omega \to \mathbb{R}$ to be the extension of $w : \Gamma \to \mathbb{R}$ and $Q : \Gamma \to \mathbb{R}$, respectively, that satisfy the following:

E1. Both w^e and Q^e are smooth on $\overline{\Omega}$;

E2. $w^e(x) = w(x^*)$ and $Q^e(x) = Q(x^*)$ for any $x \in N(\Gamma, \rho)$, where x^* is defined as in Lemma 3.1;

E3. $w^e(x) = g(x)$ for any $x \in \partial\Omega$, and $Q^e(x) = 0$ for any $x \in \Omega \setminus N(\Gamma, \rho)$.

To see the existence of w^e, we first apply Lemma 3.1 to define a function, say w_1, by $w_1(x) = w(x^*)$ for all $x \in N(\Gamma, 2\rho)$ for $\rho > 0$ small enough, and $w_1 = 0$ elsewhere in Ω. Using the lifting operator [2, 17], there exists a smooth function, say w_2, on Ω such that the restriction of w_2 on $\partial\Omega$ is g. (The smoothness depends on that of $\partial\Omega$.) Now, assume that $N(\Gamma, 2\rho)$ and $A := N(\partial\Omega, \varepsilon) = \{x \in \Omega : \text{dist}\,(x, \partial\Omega) < \varepsilon\}$ are disjoint for some small $\varepsilon > 0$. Then by applying mollifiers to $w_1 + \chi_A w_2$, we obtain the desire w^e. The existence of Q^e is seen similarly. By **E2**, both w^e and Q^e are constant along the normal to Γ in a small neighborhood of Γ.

We use a Gaussian quadrature in each triangle like $\triangle ADE$, $\triangle DEB$, and $\triangle BEC$ in Figure 2.5. Let x be such a Gaussian point which is close to the interface. We need to extend the jumps w and Q to this Gaussian point by the definition of w^e and Q^e given in **E2** and **E3**. This is done in two steps. The first step is to find an approximation of the orthogonal projection x^* of x on the interface Γ, see Lemma 3.1. The second step is to define the extensions of w and Q at x as $w(x^*)$ and $Q(x^*)$, respectively. The orthogonal projection can be approximated by

$$x^\star = x + \alpha p,$$

where

$$p = \begin{pmatrix} \dfrac{\partial\varphi}{\partial x_1} \\ \dfrac{\partial\varphi}{\partial x_2} \end{pmatrix}.$$

The scalar α is determined from the following quadratic equation:

$$\varphi(x) + (\nabla\varphi(x) \cdot p)\,\alpha + \frac{1}{2}\left(p^T \text{He}(\varphi(x))\, p\right)\,\alpha^2 = 0,$$

where

$$p^T \mathrm{He}(\varphi)\, p = \left(\frac{\partial \varphi}{\partial x_1}\right)^2 \frac{\partial^2 \varphi}{\partial x_1^2} + 2\, \frac{\partial \varphi}{\partial x_1}\, \frac{\partial \varphi}{\partial x_2}\, \frac{\partial^2 \varphi}{\partial x_1 \partial x_2} + \left(\frac{\partial \varphi}{\partial x_2}\right)^2 \frac{\partial^2 \varphi}{\partial x_2^2},$$

and $\mathrm{He}(\varphi)$ is called *Hessian matrix* given by

$$\mathrm{He}(\varphi) = \begin{pmatrix} \dfrac{\partial^2 \varphi}{\partial x_1^2} & \dfrac{\partial^2 \varphi}{\partial x_1\, \partial x_2} \\ \dfrac{\partial^2 \varphi}{\partial x_2\, \partial x_1} & \dfrac{\partial^2 \varphi}{\partial x_2^2} \end{pmatrix}.$$

The sign of α is chosen to be opposite to that of $\varphi(x)$. If the underlying mesh is uniform, formulated from a Cartesian grid, then, the partial derivatives $\nabla\varphi(x)$, the Hessian matrix $\mathrm{He}(\varphi)$ can be computed at x using the standard centered 5-point finite difference formula. Using the method above, the computed projections have third order of accuracy.

Note that in our implementation, we use the orthogonal projections $x^* \in \Gamma$. It is also possible to use the orthogonal projections $x^* \in \Gamma_h$. The difference in the finite element solution using the two different implementations is small since the area of the mis-matched region ΣT_r is small. If the level set function is the signed distance function, or a good approximation of the signed distance function, then the error in approximating x^* would be $O(h^3)$, see [31]. We also refer the reader to [13, 15] for some of the recent discussion on computing orthogonal projections.

In practice, we need only to extend both jumps w and Q to $N(\Gamma, 2h)$, the $2h$ neighborhood of the interface Γ.

3.3 Constructing the Function $\hat{u}$

In order to continue to apply the conforming immersed interface finite element space presented in Chapter 2 to solve the elliptic interface problems with non-homogeneous jump conditions: $w \neq 0$ and $Q \neq 0$ on Γ in (1.3) and (1.4), we need to have a way of removing the source singularities. The idea is to construct the function $\hat{u}$ with the *same non-homogeneous jump conditions* by using a level set method. We transform the interface problem to a new one with homogeneous jump conditions.

We define $\hat{u} : \Omega \to \mathbb{R}$ by

$$\hat{u}(x) = \chi_{\Omega^+}(x) \left(w^e(x) + \frac{Q^e(x)}{\beta^+(x)} \frac{\varphi(x)}{|\nabla \varphi(x)|} \right) \qquad \forall x \in \Omega, \tag{3.8}$$

where χ_{Ω^+} is the *characteristic function* of Ω^+ and can be written as

$$\chi_{\Omega^+}(x) = \begin{cases} 1, & \text{if } x \in \Omega^+, \\ 0, & \text{otherwise.} \end{cases} \tag{3.9}$$

Note by **E3** in Section 3.2 that $Q^e(x) = 0$ outside $N(\Gamma, \rho_0)$ in which φ may not be smooth. Thus, implicitly, the second term in $\hat{u}$ is defined to be 0 outside $N(\Gamma, \rho_0)$.

The following statement summarizes some useful properties of $\hat{u} : \Omega \to \mathbb{R}$:

Lemma 3.2. *Both $\hat{u}^-$ and $\hat{u}^+$ are smooth on Ω^- and Ω^+, respectively. Moreover, $\hat{u} = g$ on $\partial\Omega$, and*

$$[\hat{u}]_\Gamma = w, \qquad \left[\beta \frac{\partial \hat{u}}{\partial n} \right]_\Gamma = Q.$$

Proof. Since β^+ is smooth and bounded on Ω^+, it follows from (1.6), **E1** in Section

3.2, and (3.3) that $\hat{u}^+$ is smooth on Ω^+. Obviously, $\hat{u}^- = 0$ is smooth on Ω^-. By **E3** in Section 3.2, $\hat{u} = g$ on $\partial\Omega$. By (3.1) and **E2** in Section 3.2,

$$[\hat{u}]_\Gamma = \hat{u}^+\big|_\Gamma - \hat{u}^-\big|_\Gamma = w.$$

Now, fix $x \in \Gamma$. It follows from Lemma 3.1 and **E2** in Section 3.2 that $\partial_n w^e = 0$. Moreover, $\varphi(x) = 0$ by (3.1). Therefore, for any $x \in \Gamma$, we obtain by **E2**, (3.8), (3.4), and the fact that $\varphi(x) = 0$ that

$$\begin{aligned}
\left[\beta(x)\frac{\partial \hat{u}(x)}{\partial n}\right]_\Gamma &= \beta^+(x)\frac{\partial \hat{u}^+(x)}{\partial n} \\
&= \beta^+(x)\frac{\partial w^e(x)}{\partial n} + \beta^+(x)\frac{\partial}{\partial n}\left(\frac{Q^e(x)}{\beta^+(x)|\nabla\varphi(x)|}\right)\varphi(x) + \frac{Q^e(x)}{|\nabla\varphi(x)|}\frac{\partial\varphi(x)}{\partial n} \\
&= Q(x)\frac{n(x)\cdot\nabla\varphi(x)}{|\nabla\varphi(x)|} \\
&= Q(x).
\end{aligned}$$

The proof is completed. **Q.E.D.**

3.4 Removing Source Singularities

Since $\hat{u}$ constructed in Section 3.3 has the same non-homogeneous jump conditions w and Q as unknown function u, we obtain

Theorem 3.1. *There exists a unique $q \in H_0^1(\Omega)$ such that*

$$\int_\Omega \beta \nabla q \cdot \nabla v \, dx = \int_\Omega f v \, dx - \int_\Gamma Q v \, ds - \int_{\Omega^+} \beta^+ \nabla \hat{u} \cdot \nabla v \, dx \qquad \forall v \in H_0^1(\Omega). \tag{3.10}$$

This is equivalent to

$$\int_\Omega \beta \nabla q \cdot \nabla v \, dx = \int_\Omega f v \, dx + \int_{\Omega^+} (\nabla \cdot \beta^+ \nabla \hat{u}) \, v \, dx \qquad \forall v \in H_0^1(\Omega). \tag{3.11}$$

Moreover, $u = q + \hat{u}$ solves the problem (1.1)–(1.5). *In particular, $q \in H_0^1(\Omega)$ satisfies the homogeneous jump conditions*

$$[q]_\Gamma = 0 \qquad \textit{and} \qquad \left[\beta \frac{\partial q}{\partial n}\right]_\Gamma = 0. \tag{3.12}$$

Proof. Note that the line integral in (3.10) defines a continuous, linear functional on $H_0^1(\Omega)$. The existence and uniqueness of $q \in H_0^1(\Omega)$ that satisfies (3.10) follows therefore from the Lax-Milgram Theorem. By integration by parts, (3.8), and Lemma 3.2, we imply that (3.10) and (3.11) are equivalent.

Choosing $v \in H_0^1(\Omega)$ with $\operatorname{supp} v \subset \Omega^-$ and $\operatorname{supp} v \subset \Omega^+$, respectively, applying the regularity theory of elliptic problems, we obtain from (3.10) that $u = q + \hat{u}$ satisfies (1.1) and (1.2). The jump condition (1.3) follows from Lemma 3.2. Now, integrating by parts, we obtain by (1.1), (1.2), (3.8), and (3.10) that

$$\int_\Gamma \left(\left[\beta \frac{\partial u}{\partial n}\right]_\Gamma - Q \right) v \, ds = 0 \qquad \forall v \in H_0^1(\Omega).$$

This leads to (1.4). Clearly, the function $u = q + \hat{u}$ satisfies (1.5).

Finally, since $q = u - \hat{u}$, we have

$$[q]|_\Gamma = [u]|_\Gamma - [\hat{u}]|_\Gamma = w - w = 0$$

from (1.3), (1.4) and Lemma 3.2. We also have

$$\left[\beta \frac{\partial q}{\partial n}\right]_\Gamma = \left[\beta \frac{\partial u}{\partial n}\right]_\Gamma - \left[\beta \frac{\partial \hat{u}}{\partial n}\right]_\Gamma = Q - Q = 0$$

Q.E.D.

3.5 Modifying the Load Vector

On non-interface elements where the interface does not cut through, we can use the standard way in computing the contribution to the stiffness matrix and the load vector. On interface elements where the interface cuts through, we need to modify the load vector (the right-hand side); but the computation for the stiffness matrix remains to be the same as in the case of homogeneous jump conditions.

The function $q = u - \hat{u}$ constructed in Section 3.4 satisfies the homogeneous jump condition (3.12), we can use the conforming immersed interface finite element method to solve it and the resulting linear system is $AU^q = F^q$. Since the weak form of q is (3.11), the entries of the stiffness matrix $A = \{a_{ij}\}$ and the load vector $F^q = \{F_i^q\}$ can be written as

$$a_{ij} = \int_\Omega \beta \nabla \phi_i \cdot \nabla \phi_j \, dx, \tag{3.13}$$

$$F_i^q = \int_\Omega f \phi_i \, dx + \int_{\Omega^+} (\nabla \cdot \beta^+ \nabla \hat{u}) \, \phi_i \, dx. \tag{3.14}$$

Let $U = \{u_i\}$ and $U^q = \{q_i\}$ are the finite element solution of u and q at the i^{th} nodal point N_i, $i = 1, \ldots, N$ and $\hat{U} = \{\hat{u}_i\}$ is the exact value of the function $\hat{u}$ at the i^{th} nodal point N_i, $i = 1, \ldots, N$, then we have $U = U^q + \hat{U}$, therefore we have

$$AU = A(U^q + \hat{U}) = AU^q + A\hat{U} = F^q + A\hat{U} = F \tag{3.15}$$

where F is the load vector of u.

To be more precise, let us focus on the quadratic, conforming element (cf. Subsection 2.3.3) and combine (3.13), (3.14) and (3.15) , we can write the finite-element equation in terms of $u_h = q_h + \hat{u}$:

$$\int_\Omega \beta \nabla u_h \cdot \nabla v_h \, dx = \int_\Omega f \, v_h \, dx + \int_\Omega \beta \nabla \hat{u} \cdot \nabla v_h \, dx + \int_\Omega H(\varphi) \, \nabla \cdot (\beta \nabla \tilde{u}) \, v_h \, dx \quad \forall v_h \in V_h. \tag{3.16}$$

where $H(\cdot)$ is the *Heaviside function*, defined as

$$H(\varphi) = \begin{cases} 0 & \text{if } \varphi(x) < 0, \\ \dfrac{1}{2} & \text{if } \varphi(x) = 0, \\ 1 & \text{if } \varphi(x) > 0, \end{cases} \tag{3.17}$$

and $\tilde{u}$ is given by

$$\tilde{u} = w^e(x) + \frac{Q^e(x)}{\beta^+(x)} \frac{\varphi(x)}{|\nabla \varphi(x)|} \qquad \forall x \in \Omega. \tag{3.18}$$

Note that $\tilde{u}$ is a smooth function in a neighborhood of Γ and that $\hat{u} = \chi_{\Omega^+} \tilde{u}$ and its corresponding flux have non-homogeneous jumps across the interface. Clearly, the

left-hand side is exactly the same as in the case of the homogeneous jump condition. The entries of the stiffness matrix are (3.13) or (2.30). Only the right-hand side of the system of equation needs to be modified for certain triangles near the interface.

At a non-interface triangle that is entirely in Ω^-, the last two terms of integration over the triangle are zero, since $H(\varphi(x)) = 0$ and $\hat{u} = 0$. The situation is a little more complicated for triangles in Ω^+. If all the non-zero basis functions over a triangle in Ω^+ have no support from interface triangles, then the last two terms in (3.16) are canceled out. To see this, let T_k be such a triangle, and ϕ_l be such a basis function with Ω_l being its support. We have

$$\begin{aligned}
\int_{\Omega_l} \beta \nabla \hat{u} \cdot \nabla \phi_l \, dx \quad &+ \quad \int_{\Omega_l} H(\varphi) \, \nabla \cdot (\beta \nabla \tilde{u}) \, \phi_l \, dx = \\
&= \int_{\Omega_l} \beta \nabla \hat{u} \cdot \nabla \phi_l \, dx + \int_{\Omega_l} \nabla \cdot (\beta \nabla \tilde{u}) \, \phi_l \, dx \\
&= \int_{\Omega_l} \beta \nabla \hat{u} \cdot \nabla \phi_l \, dx + \int_{\partial \Omega_l} \frac{\partial \tilde{u}}{\partial n} \phi_l \, ds - \int_{\Omega_l} \beta \nabla \tilde{u} \cdot \nabla \phi_l \, dx = 0,
\end{aligned}$$

since $\tilde{u} = \hat{u}$ and $\phi_l = 0$ along $\partial \Omega_l$. In other words, the total contribution of the line integral along the boundary of each triangle summed up to be zero. The right hand side of the load vector can be summarized as the following,

$$F_i = \begin{cases} \displaystyle\int_{\Omega_i} f \phi_i \, dx & \text{if } \Omega^s(\phi_i) \cap \Gamma = \emptyset, \\ \displaystyle\int_{\Omega_i} f \phi_i \, dx + \int_{\Omega^s} H(\varphi) \nabla \cdot (\beta \nabla \tilde{u}) \, \phi_i \, dx & \\ \qquad + \displaystyle\sum_j \hat{u}(x_j) \int_{\Omega_i} \beta \nabla \phi_j \cdot \nabla \phi_i \, dx & \text{otherwise,} \end{cases} \tag{3.19}$$

where $\Omega^s(\phi_i)$ is the non-zero support region of ϕ_i.

Take Figure 2.3 as an example. There are two ways to evaluate the integral. One way is to use

$$\int_{T_j} H(\varphi)\nabla\cdot(\beta\nabla\tilde{u})\,\phi_i\,dx = \int_{\partial\triangle ADE}\beta\,\frac{\partial\tilde{u}}{\partial n}\,\phi_i\,ds - \int_{\triangle ADE}\nabla\tilde{u}\cdot\nabla\phi_i\,dx \tag{3.20}$$

The line integral is evaluated using a Gaussian quadrature formula or some other numerical quadrature.

The second approach is to evaluate the double integral directly over the triangle using a quadrature formula, say, the four-point formula [21]. The coefficient β is approximated by a constant in the triangle. In order to evaluate the values of the integrand at a given point x, we first find a square $(x_i, x_{i+1})\times(y_j, y_{j+1})$ that contains the point x. The Laplacian at the vertices $(x_{i\pm1}, y_{j\pm1})$ is computed using the standard three-point central finite difference formula. Finally, the Laplacian at the point x is interpolated using the bi-linear interpolation, see [29].

3.6 Numerical Results

Example 3.1. *A complicated interface and non-homogeneous jump conditions.*

We consider the problem (1.1)–(1.5) with $\Omega = (-1,1)\times(-1,1)$ and the interface Γ being the zero level set of the function

$$\varphi(x_1,x_2) = \sqrt{x_1^2+x_2^2} - 0.1\sin(5\theta-\pi/5) - 0.5,$$

where $\tan\theta = x_1/x_2$ and $0\le\theta\le 2\pi$; see Figure 3.1 for the geometry. The function

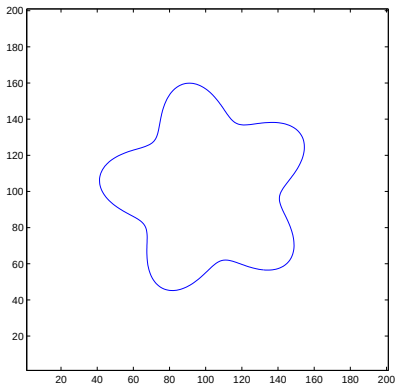

Figure 3.1: The domain and the interface for Example 3.1.

β is defined by

$$\beta(x_1, x_2) = \begin{cases} x_1^2 + x_2^2 + 1 & \text{if } (x_1, x_2) \in \Omega^-, \\ b & \text{otherwise,} \end{cases}$$

where $b > 0$ is a parameter. The function f is defined by

$$f(x_1, x_2) = \begin{cases} -4\left(2x_1^2 + 2x_2^2 + 1\right) & \text{if } (x_1, x_2) \in \Omega^-, \\ 2\sin x_1 \cos x_2 & \text{otherwise.} \end{cases}$$

Both β and f have nonzero jumps across the interface Γ. The exact solution is

$$u(x_1, x_2) = \begin{cases} x_1^2 + x_2^2, & \text{if}(x_1, x_2) \in \Omega^-, \\ \dfrac{1}{b}\left(\sin x_1 \cos x_2 + \log\sqrt{x_1^2 + x_2^2}\right) & \text{otherwise.} \end{cases}$$

Notice that both the solution u and the normal flux $\beta\partial_n u$ have nonzero jumps across the interface Γ.

We remark that the solution behavior depends on the magnitude of the parameter b. If b is large, then the solution is close to a piecewise quadratic function. If b is small, then the jumps of the solution and its normal flux across the interface are very large. Numerically, this gives rise to difficulties to achieve optimal convergence properties [1, 28]. We test our method for various b and analyze the computed results.

In Table 3.1, we show a grid refinement analysis for $b = 100$. We see clearly the second-order accuracy in L^∞ and L^2 norms, and first-order accuracy in H^1 norm.

Table 3.1: A grid refinement analysis for Example 3.1 with $b = 100$, where p_i are the approximated convergence order and the norms that involve the partial derivatives. Second-order accuracy in L^∞ norm is observed.

N	$\|u-u_h\|_\infty/\|u\|_\infty$	p_1	$\|u-u_h\|_{L^2}$	p_2	$\|u-u_h\|_{H^1}$	p_3
32	1.1995×10^{-1}		1.6705×10^{-2}		3.9175×10^{-1}	
64	2.4397×10^{-2}	2.30	1.8542×10^{-3}	3.17	1.9551×10^{-1}	1.00
128	5.3913×10^{-3}	2.18	3.2668×10^{-4}	2.51	9.8144×10^{-2}	0.99
256	1.1218×10^{-3}	2.27	5.1452×10^{-5}	2.66	4.9894×10^{-2}	0.94
512	2.7480×10^{-4}	2.03	9.4668×10^{-6}	2.44	2.5310×10^{-2}	0.98

Where b gets smaller, the jumps in the solution and flux get larger. For interface problems, the errors obtained from non-body fitted meshes usually do not decrease monotonically as we refine the mesh; see for example [26]. For small b, it is thus more realistic to find the asymptotic convergence rate as the slope of the line fitting of the experimental data $(\log(h_i), \log(E_i))$.

In Figure 3.2, we show the linear regression analysis for $b = 1$ and $b = 0.1$ for the computed finite element solution. For these two cases, the convergence orders are 2.8122; and 2.4061. As the mesh gets finer, the linear regression analysis done (by deleting the results from coarse meshes) are getting closer to number two indicating

a second-order accuracy. In Figure 3.3, we also show the linear regression analysis in L^2 and H^1 norm. The convergence orders are 1.9906; and 0.9135 respectively.

Figure 3.4 shows the result for $b = 0.01$ that is quite small, and hence there is large ratio in the coefficient from both sides of the interface. The convergence order from the sample meshes ranging from 40 to 500 with 10 increment is 1.8875, see Figure 3.4 (a). But as the mesh gets finer, the linear regression analysis done by cutting the results from coarse meshes are getting closer to number two again indicating a second-order accuracy. Figure 3.4 (b) shows the convergence order to be 1.9811.

(a). (b).

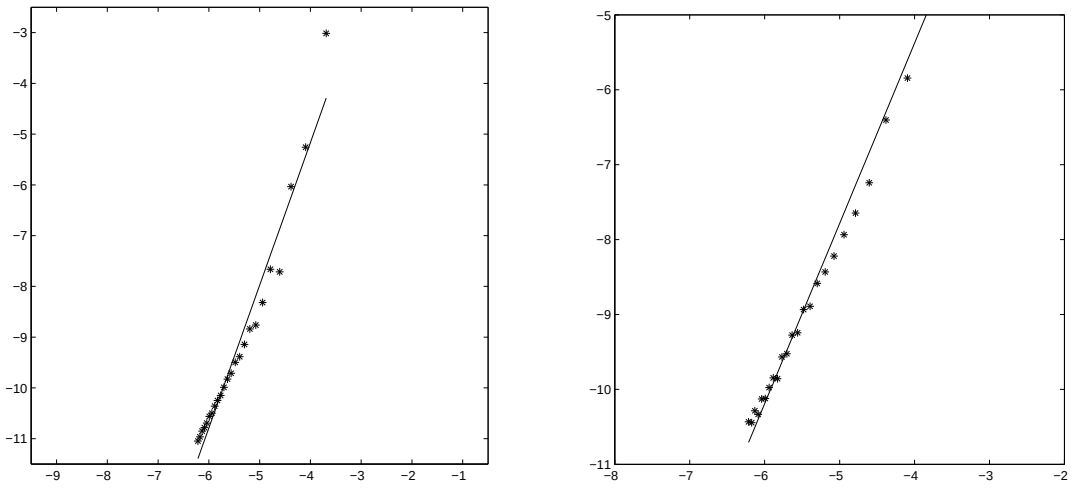

Figure 3.2: The linear regression analysis in the L^∞ norm in log-log scale with the mesh varying according to $N = 40 + 20k$, $k = 0, 1, \ldots, 23$. (a). $b = 1$, the slope (convergence order) is 2.8122; (b). $b = 0.1$, the slope is 2.4061.

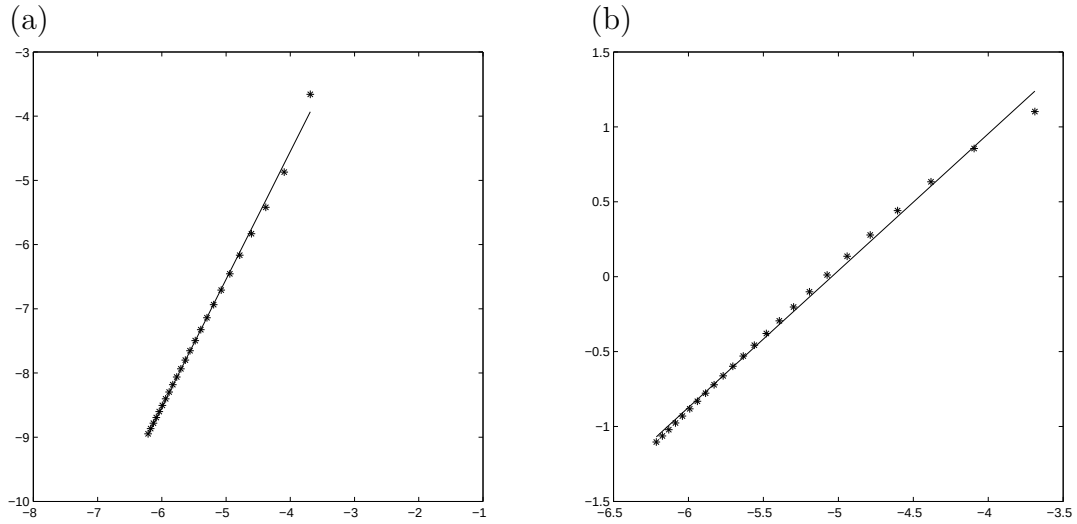

Figure 3.3: The linear regression analysis in the L^2 norm (a), and in H^1 norm (b), in log-log scale with the mesh varying according to $N = 40 + 20k$, $k = 0, 1, \ldots, 23$, $b = 1$. The slope (convergence order) is 1.9906 and 0.9135, respectively.

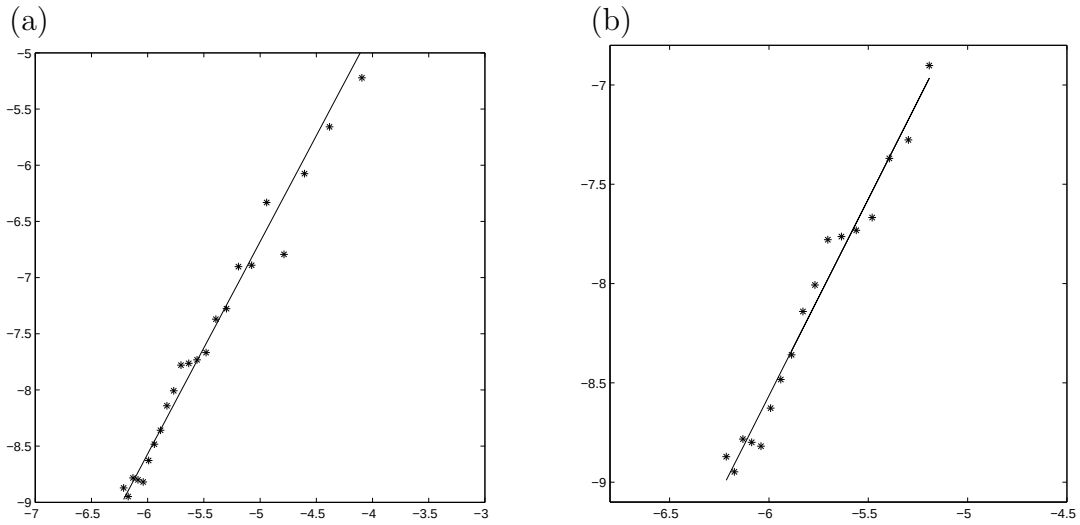

Figure 3.4: The linear regression analysis in the L^∞ norm in log-log scale for $b = 0.01$. (a). $N = 40 + 20k$, $k = 0, 1, \ldots, 23$, the slope (convergence order) is 1.8875; (b). $N = 40 + 20k$, $k = 1, 15, \ldots, 23$, the slope is 1.9811.

Chapter 4

IIFEM for Elasticity Interface Problems with Homogeneous Jump Conditions

In this chapter, we apply the immersed interface finite element methods for elasticity interface problems with homogeneous jump conditions: $W = (w_1, w_2) = (0, 0)$ on Γ in (1.21) and (1.22), $Q = (q_1, q_2) = (0, 0)$ on Γ in (1.19) and (1.20).

In [46], a non-conforming immersed finite element method has been developed for elasticity interface problems with homogeneous jump conditions. The convergence rate of this method is first order, but there are no error estimates and grid refinement analysis for all numerical examples provided in [46].

In this chapter, we construct the non-conforming and conforming immersed interface finite element spaces and obtain their corresponding interpolation errors. We also compare the interpolation errors and the errors of the finite element solutions using the non-conforming and conforming immersed interface finite element methods and carefully complete the grid refinement analysis for all of them. Second order

of the convergence rate is achieved by using the conforming immersed interface finite element method. The resulting linear system of discretization is still symmetric positive definite.

4.1 The Weak Formulation

In [7], the weak formulation of the elasticity problem (1.7)–(1.10) is given as:

$$a(u,v) = L(v) \qquad \forall v \in \{H_0^1(\Omega)\}^2, \tag{4.1}$$

where $v = (v_1, v_2)$ and

$$a(u,v) = \int_\Omega \left\{ \lambda \nabla \cdot u \nabla \cdot v + 2\,\mu \sum_{i,j=1}^{2} \varepsilon_{ij}(u)\,\varepsilon_{ij}(v) \right\} dx, \tag{4.2}$$

$$L(v) = \int_\Omega (f_1 v_1 + f_2 v_2)\, dx. \tag{4.3}$$

We continue the derivation and put (1.11) and (1.13) into (4.2), we have

$$\begin{aligned} a(u,v) &= \int_\Omega \left\{ \lambda \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) \left(\frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} \right) \right. \\ &+ \left. 2\mu \left(\frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2}\frac{\partial v_2}{\partial x_2} \right) + \mu \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) \left(\frac{\partial v_1}{\partial x_2} + \frac{\partial v_2}{\partial x_1} \right) \right\} dx, \end{aligned}$$

$$\begin{aligned} a(u,v) = \int_\Omega \Big\{ &(\lambda + 2\mu)\frac{\partial u_1}{\partial x_1}\frac{\partial v_1}{\partial x_1} + (\lambda + 2\mu)\frac{\partial u_2}{\partial x_2}\frac{\partial v_2}{\partial x_2} \\ &+ \mu\frac{\partial u_1}{\partial x_2}\frac{\partial v_1}{\partial x_2} + \mu\frac{\partial u_2}{\partial x_1}\frac{\partial v_2}{\partial x_1} + \lambda\frac{\partial u_1}{\partial x_1}\frac{\partial v_2}{\partial x_2} + \lambda\frac{\partial u_2}{\partial x_2}\frac{\partial v_1}{\partial x_1} + \mu\frac{\partial u_1}{\partial x_2}\frac{\partial v_2}{\partial x_1} + \mu\frac{\partial u_2}{\partial x_1}\frac{\partial v_1}{\partial x_2} \Big\}\, dx. \end{aligned}$$

4.2 Construction of the Basis Functions

This section includes three subsections: basis functions for non-interface elements, a non-conforming immersed interface finite element space, and a conforming linear finite element space.

4.2.1 Basis Functions for Non-interface Elements

Since $u = (u_1, u_2)$ is a vector function, we need to construct approximation of each component of u. For a typical regular element Ω_e such as that shown in Figure 2.2, the set of basis functions given in [6] is

$$\phi^e = \begin{pmatrix} \phi_1 & 0 & \phi_2 & 0 & \phi_3 & 0 \\ 0 & \phi_1 & 0 & \phi_2 & 0 & \phi_3 \end{pmatrix}, \tag{4.4}$$

where ϕ_1, ϕ_2 and ϕ_3 are defined in (2.16), (2.17), (2.18) and are no different from those we used in the single elliptic problems.

The interpolation function on a no-interface element Ω_e can be expressed as

$$u^I = \begin{pmatrix} u_1^I \\ u_2^I \end{pmatrix} = \begin{pmatrix} \phi_1 & 0 & \phi_2 & 0 & \phi_3 & 0 \\ 0 & \phi_1 & 0 & \phi_2 & 0 & \phi_3 \end{pmatrix} \begin{pmatrix} u_1^{(1)} \\ u_2^{(1)} \\ u_1^{(2)} \\ u_2^{(2)} \\ u_1^{(3)} \\ u_2^{(3)} \end{pmatrix}, \tag{4.5}$$

where $u_1^{(1)}$, $u_1^{(2)}$ and $u_1^{(3)}$ are values of the function u_1 on three vertices of Ω_e, $u_2^{(1)}$, $u_2^{(2)}$ and $u_2^{(3)}$ are values of the function u_2 on three vertices of Ω_e.

In equation (4.5), it clearly shows that the interpolation function u_1^I is obtained from the exact values of u_1, the interpolation function u_2^I is obtained from the exact values of u_2.

4.2.2 A Non-conforming Immersed Interface Finite Element Space

In this section, we briefly review the non-conforming, immersed interface finite element space for the elasticity interface problems which was first developed in [46] for the following considerations:

- We will apply it for some numerical examples and complete error estimates and the grid refinement analysis;
- We will use it to construct a new conforming, linear, immersed interface finite element space for the elasticity interface problems with the homogeneous jump conditions.

In the matrix (4.4), there are only six non-zero basis functions. Let $T = \Delta ABC \in \mathcal{T}_h$, be an interface element; cf. Figure 4.1. We assume that the interface meets edges of this element at D and E. The set of basis functions looks like

$$\phi^e = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 \\ \psi_1 & \psi_2 & \psi_3 & \psi_4 & \psi_5 & \psi_6 \end{pmatrix}, \tag{4.6}$$

where ϕ_i, ψ_i , $i = 1, 2, \ldots, 6$, are all non-zero piecewise linear functions. As before, the basis functions associated with regular nodes are the standard conforming linear finite element basis functions; see (4.4).

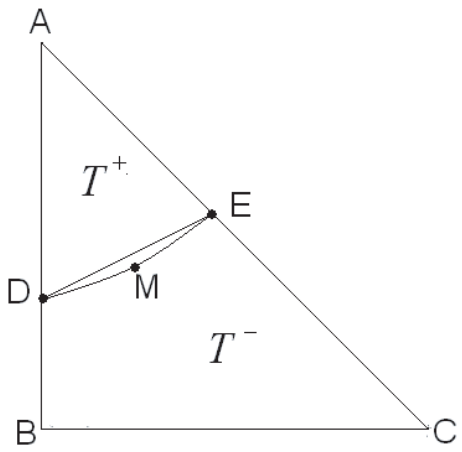

Figure 4.1: A typical interface element $T = \triangle ABC$. The arc $\overset{\frown}{DME}$ is the part of the interface Γ in T. It is approximated by the line segment $\overline{DE}$. $T^+ = \triangle ADE$, $T^- = T - T^+$, and T_r is the region enclosed by the $\overline{DE}$ and the arc $\overset{\frown}{DME}$.

The interpolation function on an interface element Ω_e such as $T = \triangle ABC$ can be expressed as

$$u^I = \begin{pmatrix} u_1^I \\ u_2^I \end{pmatrix} = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 \\ \psi_1 & \psi_2 & \psi_3 & \psi_4 & \psi_5 & \psi_6 \end{pmatrix} \begin{pmatrix} u_1^B \\ u_2^B \\ u_1^C \\ u_2^C \\ u_1^A \\ u_2^A \end{pmatrix}. \tag{4.7}$$

The interpolation function u_1^I is not obtained just from the exact values of u_1 at three vertices of $T = \triangle ABC$, it is also determined from the exact values of u_2 at three vertices of $T = \triangle ABC$. Similarly, u_2^I can be found by u_1^A, u_2^A, u_1^B, u_2^B, u_1^C and u_2^C.

Next problem is how to determine the following twelve piecewise linear functions ϕ_i and ψ_i :

$$\phi_1(x) = \begin{cases} c_{\phi_1}^{(1)} + c_{\phi_1}^{(2)} x_1 + c_{\phi_1}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\phi_1}^{(4)} + c_{\phi_1}^{(5)} x_1 + c_{\phi_1}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.8}$$

$$\phi_2(x) = \begin{cases} c_{\phi_2}^{(1)} + c_{\phi_2}^{(2)} x_1 + c_{\phi_2}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\phi_2}^{(4)} + c_{\phi_2}^{(5)} x_1 + c_{\phi_2}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.9}$$

$$\phi_3(x) = \begin{cases} c_{\phi_3}^{(1)} + c_{\phi_3}^{(2)} x_1 + c_{\phi_3}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\phi_3}^{(4)} + c_{\phi_3}^{(5)} x_1 + c_{\phi_3}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.10}$$

$$\phi_4(x) = \begin{cases} c_{\phi_4}^{(1)} + c_{\phi_4}^{(2)} x_1 + c_{\phi_4}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\phi_4}^{(4)} + c_{\phi_4}^{(5)} x_1 + c_{\phi_4}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.11}$$

$$\phi_5(x) = \begin{cases} c_{\phi_5}^{(1)} + c_{\phi_5}^{(2)} x_1 + c_{\phi_5}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\phi_5}^{(4)} + c_{\phi_5}^{(5)} x_1 + c_{\phi_5}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.12}$$

$$\phi_6(x) = \begin{cases} c_{\phi_6}^{(1)} + c_{\phi_6}^{(2)} x_1 + c_{\phi_6}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\phi_6}^{(4)} + c_{\phi_6}^{(5)} x_1 + c_{\phi_6}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.13}$$

$$\psi_1(x) = \begin{cases} c_{\psi_1}^{(1)} + c_{\psi_1}^{(2)} x_1 + c_{\psi_1}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\psi_1}^{(4)} + c_{\psi_1}^{(5)} x_1 + c_{\psi_1}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.14}$$

$$\psi_2(x) = \begin{cases} c_{\psi_2}^{(1)} + c_{\psi_2}^{(2)} x_1 + c_{\psi_2}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\psi_2}^{(4)} + c_{\psi_2}^{(5)} x_1 + c_{\psi_2}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.15}$$

$$\psi_3(x) = \begin{cases} c_{\psi_3}^{(1)} + c_{\psi_3}^{(2)} x_1 + c_{\psi_3}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\psi_3}^{(4)} + c_{\psi_3}^{(5)} x_1 + c_{\psi_3}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.16}$$

$$\psi_4(x) = \begin{cases} c_{\psi_4}^{(1)} + c_{\psi_4}^{(2)} x_1 + c_{\psi_4}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\psi_4}^{(4)} + c_{\psi_4}^{(5)} x_1 + c_{\psi_4}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.17}$$

$$\psi_5(x) = \begin{cases} c_{\psi_5}^{(1)} + c_{\psi_5}^{(2)} x_1 + c_{\psi_5}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\psi_5}^{(4)} + c_{\psi_5}^{(5)} x_1 + c_{\psi_5}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-, \end{cases} \tag{4.18}$$

$$\psi_6(x) = \begin{cases} c_{\psi_6}^{(1)} + c_{\psi_6}^{(2)} x_1 + c_{\psi_6}^{(3)} x_2, & \text{if } x = (x_1, x_2) \in T^+, \\ c_{\psi_6}^{(4)} + c_{\psi_6}^{(5)} x_1 + c_{\psi_6}^{(6)} x_2, & \text{if } x = (x_1, x_2) \in T^-. \end{cases} \tag{4.19}$$

For convenience, we use $(\phi_1, \psi_1)^T$ as an example to introduce the way to construct the basis functions. Let all coefficients of ϕ_1 and ψ_1 form a vector x_c and

$$x_c = (c_{\phi_1}^{(1)}, c_{\phi_1}^{(2)}, c_{\phi_1}^{(3)}, c_{\phi_1}^{(4)}, c_{\phi_1}^{(5)}, c_{\phi_1}^{(6)}, c_{\psi_1}^{(1)}, c_{\psi_1}^{(2)}, c_{\psi_1}^{(3)}, c_{\psi_1}^{(4)}, c_{\psi_1}^{(5)}, c_{\psi_1}^{(6)})^T, \tag{4.20}$$

where $(*)^T$ indicates the *transpose of a matrix*, which found by exchanging rows for columns. There are twelve unknown coefficients, they are solved by 12 linear equations. The u_1, u_2 at three vertices of ΔABC, see Figure 4.1 for an illustration, are given and we assume the coordinates at A, B, C, D and E are

$$(x_1^A, x_2^A), \quad (x_1^B, x_2^B), \quad (x_1^C, x_2^C), \quad (x_1^D, x_2^D), \quad (x_1^E, x_2^E),$$

therefore we have

$$
\begin{aligned}
c_{\phi_1}^{(1)} + c_{\phi_1}^{(2)} x_1^A + c_{\phi_1}^{(3)} x_2^A &= u_1^A, && (4.21)\\
c_{\psi_1}^{(1)} + c_{\psi_1}^{(2)} x_1^A + c_{\psi_1}^{(3)} x_2^A &= u_2^A, && (4.22)\\
c_{\phi_1}^{(4)} + c_{\phi_1}^{(5)} x_1^B + c_{\phi_1}^{(6)} x_2^B &= u_1^B, && (4.23)\\
c_{\psi_1}^{(4)} + c_{\psi_1}^{(5)} x_1^B + c_{\psi_1}^{(6)} x_2^B &= u_2^B, && (4.24)\\
c_{\phi_1}^{(4)} + c_{\phi_1}^{(5)} x_1^C + c_{\phi_1}^{(6)} x_2^C &= u_1^C, && (4.25)\\
c_{\psi_1}^{(4)} + c_{\psi_1}^{(5)} x_1^C + c_{\psi_1}^{(6)} x_2^C &= u_2^C. && (4.26)
\end{aligned}
$$

By the homogeneous jump condition $W = (w_1, w_2) = (0, 0)$ on Γ in (1.21) and (1.22), we have

$$
\begin{aligned}
c_{\phi_1}^{(1)} + c_{\phi_1}^{(2)} x_1^D + c_{\phi_1}^{(3)} x_2^D &= c_{\phi_1}^{(4)} + c_{\phi_1}^{(5)} x_1^D + c_{\phi_1}^{(6)} x_2^D, && (4.27)\\
c_{\phi_1}^{(1)} + c_{\phi_1}^{(2)} x_1^E + c_{\phi_1}^{(3)} x_2^E &= c_{\phi_1}^{(4)} + c_{\phi_1}^{(5)} x_1^E + c_{\phi_1}^{(6)} x_2^E, && (4.28)\\
c_{\psi_1}^{(1)} + c_{\psi_1}^{(2)} x_1^D + c_{\psi_1}^{(3)} x_2^D &= c_{\psi_1}^{(4)} + c_{\psi_1}^{(5)} x_1^D + c_{\psi_1}^{(6)} x_2^D, && (4.29)\\
c_{\psi_1}^{(1)} + c_{\psi_1}^{(2)} x_1^E + c_{\psi_1}^{(3)} x_2^E &= c_{\psi_1}^{(4)} + c_{\psi_1}^{(5)} x_1^E + c_{\psi_1}^{(6)} x_2^E. && (4.30)
\end{aligned}
$$

Last two equations are from

$$
\left[\lambda \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) n_1 + 2\mu \frac{\partial u_1}{\partial x_1} n_1 + \mu \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) n_2 \right]_\Gamma = 0, \tag{4.31}
$$

$$
\left[\lambda \left(\frac{\partial u_1}{\partial x_1} + \frac{\partial u_2}{\partial x_2} \right) n_2 + 2\mu \frac{\partial u_2}{\partial x_2} n_2 + \mu \left(\frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} \right) n_1 \right]_\Gamma = 0, \tag{4.32}
$$

we have

$$(\lambda^+ + 2\mu^+)\, n_1\, c_{\phi_1}^{(2)} + \mu^+\, n_2\, c_{\phi_1}^{(3)} + \mu^+\, n_2\, c_{\psi_1}^{(2)} + \lambda^+\, n_1\, c_{\psi_1}^{(3)} =$$
$$(\lambda^- + 2\mu^-)\, n_1\, c_{\phi_1}^{(5)} + \mu^-\, n_2\, c_{\phi_1}^{(6)} + \mu^-\, n_2\, c_{\psi_1}^{(5)} + \lambda^-\, n_1\, c_{\psi_1}^{(6)}, \quad (4.33)$$

$$\lambda^+\, n_2\, c_{\phi_1}^{(2)} + \mu^+\, n_1\, c_{\phi_1}^{(3)} + \mu^+\, n_1\, c_{\psi_1}^{(2)} + (\lambda^+ + 2\mu^+)\, n_2\, c_{\psi_1}^{(3)} =$$
$$\lambda^-\, n_2\, c_{\phi_1}^{(5)} + \mu^-\, n_1\, c_{\phi_1}^{(6)} + \mu^-\, n_1\, c_{\psi_1}^{(5)} + (\lambda^- + 2\mu^-)\, n_2\, c_{\psi_1}^{(6)}, \quad (4.34)$$

where

$$\lambda = \begin{cases} \lambda^+ & \text{if } x = (x_1, x_2) \in T^+, \\ \lambda^- & \text{if } x = (x_1, x_2) \in T^-, \end{cases}$$

$$\mu = \begin{cases} \mu^+ & \text{if } x = (x_1, x_2) \in T^+, \\ \mu^- & \text{if } x = (x_1, x_2) \in T^-, \end{cases}$$

and n is the unit normal direction of the line segment $\overline{DE}$.

We can rewrite the linear equation system as

$$A_c x_c = b_c.$$

Let $\theta^+ = \lambda^+ + 2\mu^+$ and $\theta^- = \lambda^- + 2\mu^-$, A_c is a 12×12 matrix:

$$\begin{pmatrix}
1 & x_1^A & x_2^A & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & x_1^A & x_2^A & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & x_1^B & x_2^B & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & x_1^B & x_2^B \\
0 & 0 & 0 & 1 & x_1^C & x_2^C & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & x_1^C & x_2^C \\
1 & x_1^D & x_2^D & -1 & -x_1^D & -x_2^D & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & x_1^D & x_2^D & -1 & -x_1^D & -x_2^D \\
1 & x_1^E & x_2^E & -1 & -x_1^E & -x_2^E & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & x_1^E & x_2^E & -1 & -x_1^E & -x_2^E \\
0 & \theta^+ n_1 & \mu^+ n_2 & 0 & -\theta^- n_1 & -\mu^- n_2 & 0 & \mu^+ n_2 & \lambda^+ n_1 & 0 & -\mu^- n_2 & -\lambda^- n_1 \\
0 & \lambda^+ n_2 & \mu^+ n_1 & 0 & -\lambda^- n_2 & -\mu^- n_1 & 0 & \mu^+ n_1 & \theta^+ n_2 & 0 & -\mu^- n_1 & -\theta^- n2
\end{pmatrix}. \tag{4.35}$$

The right hand side b_c is a 12×1 vector and

$$b_c = (u_1^A, u_2^A, u_1^B, u_2^B, u_1^C, u_2^C, 0, 0, 0, 0, 0, 0)^T. \tag{4.36}$$

In order to get all the coefficients of the basis function ϕ_1 and ψ_1, we set

$$b_c = (0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T,$$

then

$$\begin{aligned} x_c &= (c_{\phi_1}^{(1)}, c_{\phi_1}^{(2)}, c_{\phi_1}^{(3)}, c_{\phi_1}^{(4)}, c_{\phi_1}^{(5)}, c_{\phi_1}^{(6)}, c_{\psi_1}^{(1)}, c_{\psi_1}^{(2)}, c_{\psi_1}^{(3)}, c_{\psi_1}^{(4)}, c_{\psi_1}^{(5)}, c_{\psi_1}^{(6)})^T \\ &= A_c^{-1} b_c = A_c^{-1}(0,0,1,0,0,0,0,0,0,0,0,0)^T, \end{aligned}$$

where $(*)^{-1}$ indicates the *inverse of a matrix*.

Similarly, we can determine all the coefficients from (4.8) to (4.19):

$$\begin{aligned} x_c &= (c_{\phi_2}^{(1)}, c_{\phi_2}^{(2)}, c_{\phi_2}^{(3)}, c_{\phi_2}^{(4)}, c_{\phi_2}^{(5)}, c_{\phi_2}^{(6)}, c_{\psi_2}^{(1)}, c_{\psi_2}^{(2)}, c_{\psi_2}^{(3)}, c_{\psi_2}^{(4)}, c_{\psi_2}^{(5)}, c_{\psi_2}^{(6)})^T \\ &= A_c^{-1} b_c = A_c^{-1}(0,0,0,1,0,0,0,0,0,0,0,0)^T, \end{aligned}$$

$$\begin{aligned} x_c &= (c_{\phi_3}^{(1)}, c_{\phi_3}^{(2)}, c_{\phi_3}^{(3)}, c_{\phi_3}^{(4)}, c_{\phi_3}^{(5)}, c_{\phi_3}^{(6)}, c_{\psi_3}^{(1)}, c_{\psi_3}^{(2)}, c_{\psi_3}^{(3)}, c_{\psi_3}^{(4)}, c_{\psi_3}^{(5)}, c_{\psi_3}^{(6)})^T \\ &= A_c^{-1} b_c = A_c^{-1}(0,0,0,0,1,0,0,0,0,0,0,0)^T, \end{aligned}$$

$$\begin{aligned} x_c &= (c_{\phi_4}^{(1)}, c_{\phi_4}^{(2)}, c_{\phi_4}^{(3)}, c_{\phi_4}^{(4)}, c_{\phi_4}^{(5)}, c_{\phi_4}^{(6)}, c_{\psi_4}^{(1)}, c_{\psi_4}^{(2)}, c_{\psi_4}^{(3)}, c_{\psi_4}^{(4)}, c_{\psi_4}^{(5)}, c_{\psi_4}^{(6)})^T \\ &= A_c^{-1} b_c = A_c^{-1}(0,0,0,0,0,1,0,0,0,0,0,0)^T, \end{aligned}$$

$$\begin{aligned} x_c &= (c_{\phi_5}^{(1)}, c_{\phi_5}^{(2)}, c_{\phi_5}^{(3)}, c_{\phi_5}^{(4)}, c_{\phi_5}^{(5)}, c_{\phi_5}^{(6)}, c_{\psi_5}^{(1)}, c_{\psi_5}^{(2)}, c_{\psi_5}^{(3)}, c_{\psi_5}^{(4)}, c_{\psi_5}^{(5)}, c_{\psi_5}^{(6)})^T \\ &= A_c^{-1} b_c = A_c^{-1}(1,0,0,0,0,0,0,0,0,0,0,0)^T, \end{aligned}$$

$$
\begin{aligned}
x_c &= (c_{\phi_6}^{(1)}, c_{\phi_6}^{(2)}, c_{\phi_6}^{(3)}, c_{\phi_6}^{(4)}, c_{\phi_6}^{(5)}, c_{\phi_6}^{(6)}, c_{\psi_6}^{(1)}, c_{\psi_6}^{(2)}, c_{\psi_6}^{(3)}, c_{\psi_6}^{(4)}, c_{\psi_6}^{(5)}, c_{\psi_6}^{(6)})^T \\
&= A_c^{-1} b_c = A_c^{-1}(0,1,0,0,0,0,0,0,0,0,0,0)^T.
\end{aligned}
$$

Note that the basis functions defined in this way can be discontinuous across edges of interface elements, so when we use it to get the finite element solutions, the method is a non-conforming immersed interface finite element method.

4.2.3 A Conforming Linear Finite Element Space

In this section, we construct a conforming, linear finite element space, the basis functions associated with irregular nodes are still piecewise linear, but all the basis functions are globally continuous.

In order to maintain continuity, as in Subsection 2.3.3, we need to extend the previously defined basis function (4.6) at the same node (vertex) to one more triangle along the interface (cf. Figure 2.5 (b)). We require the local basis functions in two adjacent interface elements, such as ΔABC and ΔAFB, take the same value at the interface point on their common edge, such as the point D. This will achieve the global continuity of a basis function associated with an irregular node.

For an interface element $T = \Delta ABC \in \mathcal{T}_h$, see Figure 2.5, the set of basis functions becomes

$$
\phi^e = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 & \phi_7 & \phi_8 & \phi_9 & \phi_{10} \\ \psi_1 & \psi_2 & \psi_3 & \psi_4 & \psi_5 & \psi_6 & \psi_7 & \psi_8 & \psi_9 & \psi_{10} \end{pmatrix}. \tag{4.37}
$$

The interpolation function on an interface element Ω_e such as $T = \triangle ABC$ is

$$u^I = \begin{pmatrix} u_1^{IP} \\ u_2^{IP} \end{pmatrix} = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 & \phi_7 & \phi_8 & \phi_9 & \phi_{10} \\ \psi_1 & \psi_2 & \psi_3 & \psi_4 & \psi_5 & \psi_6 & \psi_7 & \psi_8 & \psi_9 & \psi_{10} \end{pmatrix} \begin{pmatrix} u_1^B \\ u_2^B \\ u_1^C \\ u_2^C \\ u_1^A \\ u_2^A \\ u_1^F \\ u_2^F \\ u_1^I \\ u_2^I \end{pmatrix}, \tag{4.38}$$

where $u^I = (u_1^{IP}, u_2^{IP})$ indicates the interpolation function in order to avoid the confusion with (u_1^I, u_2^I) which is the value at the vertex I in (4.38) (cf. Figure 2.5 (b)).

We construct a local basis function ϕ_i and ψ_i , $i = 1, 2, \ldots, 10$, by assigning their values at the vertices A, B, C, F, and I, respectively. This construction consists of the following five steps:

P1. Use the values at the nodes A, B, C, F, and I to construct the three sets of nonconforming finite-element basis functions defined as in Subsection 4.2.2 on the elements ΔABC, ΔAFB, and ΔACI, respectively;

P2. Set the value at D as the average of the values at D of the nonconforming, piecewise linear, basis functions defined on ΔABC and ΔAFB constructed in **P1**;

P3. Similarly, set the value at E as the average of values at E of the nonconforming, piecewise linear, basis functions defined on the elements ΔABC and ΔACI constructed in **P1**;

P4. Set the values of at the points A, B and C exactly the same as those from the nonconforming finite-element basis function on ΔABC:

$$\phi^e(B) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\phi^e(C) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\phi^e(A) = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix};$$

P5. Partition the element ΔABC into three sub-triangles by an auxiliary line, say line segment $\overline{BE}$, or $\overline{DC}$. We choose the auxiliary line in such a way that at least one of angles (or complimentary angle if the angle is more than $\pi/4$) is bigger than or equal to $\pi/2$, see [30] for more explanations;

P6. Define the basis functions ϕ_i and ψ_i, $i = 1, 2, \ldots, 10$, to be the piecewise linear function in the three sub-triangles

$$\phi_i(x) = \begin{cases} c_{\phi_i}^{(1)} + c_{\phi_i}^{(2)} x_1 + c_{\phi_i}^{(3)} x_2 & \text{if } x = (x_1, x_2) \in \Delta ADE, \\ c_{\phi_i}^{(4)} + c_{\phi_i}^{(5)} x_1 + c_{\phi_i}^{(6)} x_2 & \text{if } x = (x_1, x_2) \in \Delta DBE, \\ c_{\phi_i}^{(7)} + c_{\phi_i}^{(8)} x_1 + c_{\phi_i}^{(9)} x_2 & \text{if } x = (x_1, x_2) \in \Delta BCE, \end{cases} \tag{4.39}$$

$$\psi_i(x) = \begin{cases} c_{\psi_i}^{(1)} + c_{\psi_i}^{(2)} x_1 + c_{\psi_i}^{(3)} x_2 & \text{if } x = (x_1, x_2) \in \Delta ADE, \\ c_{\psi_i}^{(4)} + c_{\psi_i}^{(5)} x_1 + c_{\psi_i}^{(6)} x_2 & \text{if } x = (x_1, x_2) \in \Delta DBE, \\ c_{\psi_i}^{(7)} + c_{\psi_i}^{(8)} x_1 + c_{\psi_i}^{(9)} x_2 & \text{if } x = (x_1, x_2) \in \Delta BCE, \end{cases} \tag{4.40}$$

or

$$\phi_i(x) = \begin{cases} c_{\phi_i}^{(1)} + c_{\phi_i}^{(2)} x_1 + c_{\phi_i}^{(3)} x_2 & \text{if } x = (x_1, x_2) \in \Delta ADE, \\ c_{\phi_i}^{(4)} + c_{\phi_i}^{(5)} x_1 + c_{\phi_i}^{(6)} x_2 & \text{if } x = (x_1, x_2) \in \Delta EDC, \\ c_{\phi_i}^{(7)} + c_{\phi_i}^{(8)} x_1 + c_{\phi_i}^{(9)} x_2 & \text{if } x = (x_1, x_2) \in \Delta DBC, \end{cases} \tag{4.41}$$

$$\psi_i(x) = \begin{cases} c_{\psi_i}^{(1)} + c_{\psi_i}^{(2)} x_1 + c_{\psi_i}^{(3)} x_2 & \text{if } x = (x_1, x_2) \in \Delta ADE, \\ c_{\psi_i}^{(4)} + c_{\psi_i}^{(5)} x_1 + c_{\psi_i}^{(6)} x_2 & \text{if } x = (x_1, x_2) \in \Delta EDC, \\ c_{\psi_i}^{(7)} + c_{\psi_i}^{(8)} x_1 + c_{\psi_i}^{(9)} x_2 & \text{if } x = (x_1, x_2) \in \Delta DBC. \end{cases} \tag{4.42}$$

All unknown coefficients $c_{\phi_i}^{(j)}$ and $c_{\psi_i}^{(j)}$, $i = 1, 2, \ldots, 10$, $j = 1, 2, \ldots, 9$ are determined by the values at the points A, B, C, D and E.

4.3 Assembling the Stiffness Matrix and Load Vector

In this section, we keep using $A = \{a_{ij}\}$ and $F = \{F_i\}$ as notations of the global stiffness matrix and the global load vector. Since $u = (u_1, u_2)$ is a two dimensional vector function, A is a $2N \times 2N$ matrix, F is a $2N \times 1$ vector, here N is the total

number of nodal points.

We define an immersed-interface finite-element space V_h with respect to the mesh $\mathcal{T}_h$ to be a finitely-dimensional subspace of $\{L^2(\Omega)\}^2$. To be more precise, we specially define a conforming, linear immersed-interface finite-element space V_h for the approximation of a elasticity interface problem to be a subspace of $\{H_0^1(\Omega)\}^2$. V_h consists of all the linear combinations of the corresponding basis functions $(\phi_1, \psi_1)^T, \ldots, (\phi_{2N}, \psi_{2N})^T$ for some integer $N \geq 1$:

$$V_h = \text{span} \left\{ \begin{pmatrix} \phi_1 \\ \psi_1 \end{pmatrix}, \ldots, \begin{pmatrix} \phi_{2N} \\ \psi_{2N} \end{pmatrix} \right\}. \tag{4.43}$$

The interpolation function $u^I = (u_1^I, u_2^I)$ on the domain Ω is

$$u_1^I(x) = \sum_{i=1}^{N} \left\{ u_1(x^{(i)})\, \phi_{2i-1}(x) + u_2(x^{(i)})\, \phi_{2i}(x) \right\}, \tag{4.44}$$

$$u_2^I(x) = \sum_{i=1}^{N} \left\{ u_1(x^{(i)})\, \psi_{2i-1}(x) + u_2(x^{(i)})\, \psi_{2i}(x) \right\}, \tag{4.45}$$

where $u_1(x^{(i)})$ and $u_2(x^{(i)})$ are the exact values of u_1, u_2 at the i^{th} node (N_i). The finite element solution $u^h = (u_1^h, u_2^h)$ is given by

$$u_1^h(x) = \sum_{i=1}^{N} \left\{ u_1^{(i)}\, \phi_{2i-1}(x) + u_2^{(i)}\, \phi_{2i}(x) \right\}, \tag{4.46}$$

$$u_2^h(x) = \sum_{i=1}^{N} \left\{ u_1^{(i)}\, \psi_{2i-1}(x) + u_2^{(i)}\, \psi_{2i}(x) \right\}, \tag{4.47}$$

where $u_1^{(i)}$ and $u_2^{(i)}$ are the finite element solutions of u_1, u_2 at the i^{th} node (N_i).

We put the testing function $v = (v_1, v_2)^T = (\phi_i, \psi_i)^T$, $i = 1, \ldots, 2N$, and (4.46), (4.47) into the weak form (4.2) and (4.3), we have all entries of the stiffness matrix and the load vector for the elasticity problems:

$$\begin{aligned} a_{ij} &= \int_\Omega \left\{ (\lambda+2\mu)\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_1} + (\lambda+2\mu)\frac{\partial\psi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_2} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_2} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_1} \right. \\ &+ \left. \lambda\frac{\partial\phi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_2} + \lambda\frac{\partial\psi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_1} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_1} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_2} \right\} dx, \end{aligned}$$

$$F_i = \int_\Omega (f_1\phi_i + f_2\psi_i)\, dx.$$

For a *type 1* interface element $\Omega_e = \Delta N_1 N_2 N_3$, see Figure 2.9 (b) for an illustration, the local stiffness matrix A^{Ω_e} is a 10×10 matrix and symmetric, the local load vector F^{Ω_e} is a 10×1 vector. They can be expressed as

$$\begin{aligned} a_{ij}^{\Omega_e} &= \int_{\Omega_e} \left\{ (\lambda+2\mu)\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_1} + (\lambda+2\mu)\frac{\partial\psi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_2} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_2} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_1} \right. \\ &+ \left. \lambda\frac{\partial\phi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_2} + \lambda\frac{\partial\psi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_1} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_1} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_2} \right\} dx, \quad i,j = 1, \ldots, 10, \end{aligned}$$

$$F_i^{\Omega_e} = \int_{\Omega_e} (f_1\phi_i + f_2\psi_i)\, dx, \quad i,j = 1, \ldots, 10.$$

For a *type 2* interface element $\Omega_e = \Delta N_1 N_2 N_3$, see Figure 2.9 (c) for an illustration, the local stiffness matrix A^{Ω_e} is an 8×8 matrix and symmetric, the local load

vector F^{Ω_e} is an 8×1 vector. They can be expressed as

$$\begin{aligned} a_{ij}^{\Omega_e} &= \int_{\Omega_e} \left\{ (\lambda+2\mu)\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_1} + (\lambda+2\mu)\frac{\partial\psi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_2} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_2} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_1} \right. \\ &+ \left. \lambda\frac{\partial\phi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_2} + \lambda\frac{\partial\psi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_1} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_1} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_2} \right\} dx, \quad i,j = 1,\ldots,8, \end{aligned}$$

$$F_i^{\Omega_e} = \int_{\Omega_e} (f_1\phi_i + f_2\psi_i)\, dx, \quad i,j = 1,\ldots,8.$$

For a non-interface elements $\Omega_e = \Delta N_1 N_2 N_3$, see Figure 2.9 (a), where the interface does not cut through, the local stiffness matrix A^{Ω_e} is a 6×6 matrix and the local load vector F^{Ω_e} is a 6×1 vector. There are only six non-zero basis functions in (4.4) and ϕ_1, ϕ_2, ϕ_3 are defined in (2.16), (2.17) and (2.18). The expressions of A^{Ω_e} and F^{Ω_e} are different from above:

$$a_{2i-1,2j-1}^{\Omega_e} = \int_{\Omega_e} \left\{ (\lambda+2\mu)\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_1} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_2} \right\} dx, \qquad i,j = 1,2,3,$$

$$a_{2i-1,2j}^{\Omega_e} = \int_{\Omega_e} \left\{ \lambda\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_1} + \mu\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_2} \right\} dx, \qquad i,j = 1,2,3,$$

$$a_{2i,2j-1}^{\Omega_e} = \int_{\Omega_e} \left\{ \lambda\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_2} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_1} \right\} dx, \qquad i,j = 1,2,3,$$

$$a_{2i,2j}^{\Omega_e} = \int_{\Omega_e} \left\{ (\lambda+2\mu)\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_2} + \mu\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_1} \right\} dx, \qquad i,j = 1,2,3,$$

$$F_{2i-1}^{\Omega_e} = \int_{\Omega_e} f_1 \phi_i \, dx, \qquad i,j = 1,2,3,$$

$$F_{2i}^{\Omega_e} = \int_{\Omega_e} f_2 \phi_i \, dx, \qquad i,j = 1,2,3.$$

4.4 Numerical Results

In this section, we give one example of numerical calculations using our non-conforming and conforming, linear, immersed-interface finite-element method.

Example 4.1. *Elasticity interface problem with homogeneous jump conditions.*

We consider the problem (1.7)–(1.10) with $\Omega = (-1,1)\times(-1,1)$, Γ being the circle centered at point $(0,0)$ with radius $R = 0.5$, and $\lambda^- = \mu^- = 1$ and $\lambda^+ = \mu^+ = 100$. The body force term $F = (f_1, f_2)$ and the Dirichlet boundary condition $G = (g_1, g_2)$ are calculated from the exact solution $u = (u_1, u_2)$:

$$u_1(x_1,x_2) = u_2(x_1,x_2) = \begin{cases} (x_1^2 + x_2^2) & \text{if } r \le R, \\ \\ \dfrac{(x_1^2+x_2^2)}{100} + \left(1 - \dfrac{1}{100}\right) R^2 & \text{otherwise,} \end{cases}$$

where $r = \sqrt{x_1^2 + x_2^2}$. The exact solution satisfies the homogeneous jump conditions.

In Figure 4.2, we show a grid refinement analysis for the interpolation error $\|u - u^I\|_\infty$ by using the non-conforming, linear, immersed interface finite element space constructed in Subsection 4.2.2, here $u = (u_1, u_2)$ and $u^I = (u_1^I, u_2^I)$. We see clearly

the second order accuracy in Figure 4.2 (b) when the mesh is fine enough. The convergence order is 2.0691 with the mesh varying according to $N = 120 + 20k$, $k = 0, 1, \ldots, 19$.

(a). (b).

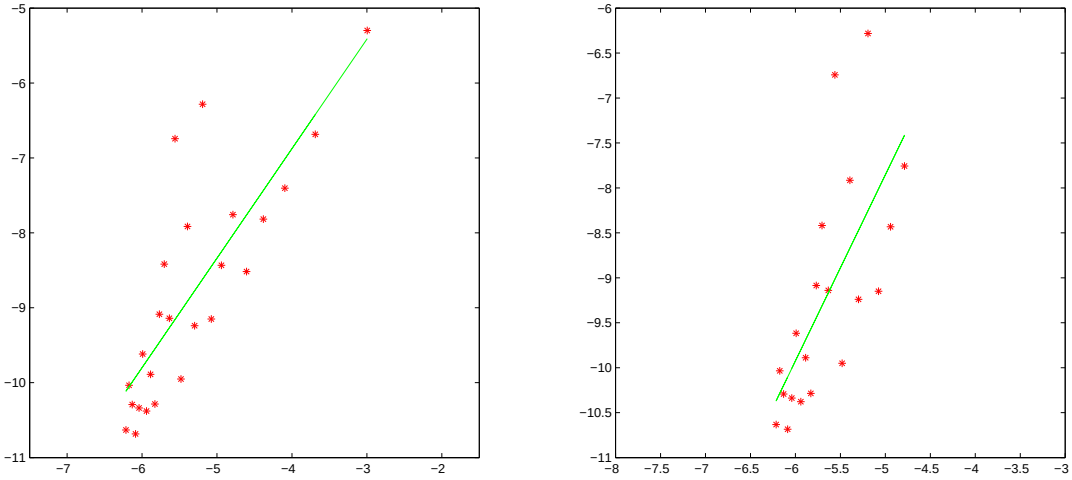

Figure 4.2: The linear regression analysis of $(u - u^I)$ in the L^∞ norm in log-log scale by using the non-conforming, linear, immersed interface finite element space. (a). the mesh is varying according to $N = 20 + 20k$, $k = 0, 1, \ldots, 24$, the slope (convergence order) is 1.4616; (b). $N = 120 + 20k$, $k = 0, 1, \ldots, 19$, the slope is 2.0691.

In Figure 4.3, we show the results of the errors of the finite element solution $(u - u^h)$ by using the non-conforming, linear, immersed interface finite element method in L^∞ norm, here $u^h = (u_1^h, u_2^h)$. Since the finite element space is non-conforming, we only have the first order accuracy. The convergence order is 0.9205 with the mesh varying according to $N = 20 + 20k$, $k = 0, 1, \ldots, 24$.

We now apply the conforming, linear, immersed interface finite element space obtained in Subsection 4.2.3 to solve the same example above. The interpolation errors are showed in Figure 4.4. We see that the interpolation errors are second order and the convergence order is 2.1479 with the mesh varying according to $N = 120+20k$,

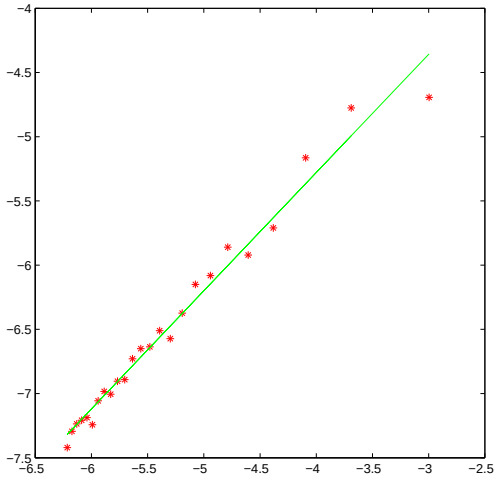

Figure 4.3: The linear regression analysis of $(u-u^h)$ in the L^∞ norm in log-log scale by using the non-conforming, linear, immersed interface finite element space. The mesh is varying according to $N = 20 + 20k$, $k = 0, 1, \ldots, 24$, the slope (convergence order) is 0.9205.

$k = 0, 1, \ldots, 19$.

We compare the interpolation errors using the non-conforming and conforming, linear, immersed interface finite element spaces, see Figure 4.2 (a) versus Figure 4.4 (a) and Figure 4.2 (b) versus Figure 4.4 (b), the comparison shows that the convergence orders using the conforming linear finite element space are higher than using the non-conforming linear finite element space.

We provide some interpolation errors obtained by using the non-conforming and conforming, linear, immersed interface finite element spaces in Table 4.1, the interpolation errors using the conforming linear finite element space are smaller than using the non-conforming linear finite element space.

In Figure 4.5, we show the results of the errors of the finite element solution $(u-u^h)$ by using the conforming, linear, immersed interface finite element method in L^∞

(a). (b).

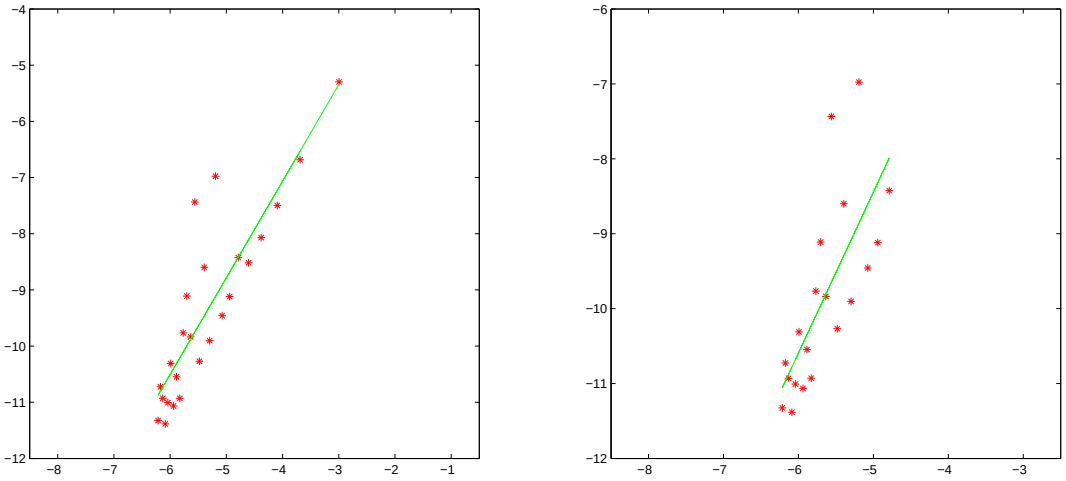

Figure 4.4: The linear regression analysis of $(u - u^I)$ in the L^∞ norm in log-log scale by using the conforming, linear, immersed interface finite element space. (a). the mesh is varying according to $N = 20 + 20k$, $k = 0, 1, \ldots, 24$, the slope (convergence order) is 1.7192; (b). $N = 120 + 20k$, $k = 0, 1, \ldots, 19$, the slope is 2.1479.

Table 4.1: The interpolation errors in L^∞ for Example 4.1.

Mesh	Non-conforming space	Conforming space
400	6.6574×10^{-5}	3.3281×10^{-5}
420	3.2381×10^{-5}	1.6581×10^{-5}
440	2.2896×10^{-5}	1.1380×10^{-5}
460	3.3878×10^{-5}	1.7915×10^{-5}
480	4.3826×10^{-5}	2.1976×10^{-5}
500	2.4131×10^{-5}	1.2075×10^{-5}

norm. Since the finite element space is conforming, we have the second order accuracy. The convergence order is 1.9443 with the mesh varying according to $N = 120 + 20k$, $k = 0, 1, \ldots, 19$.

(a). (b).

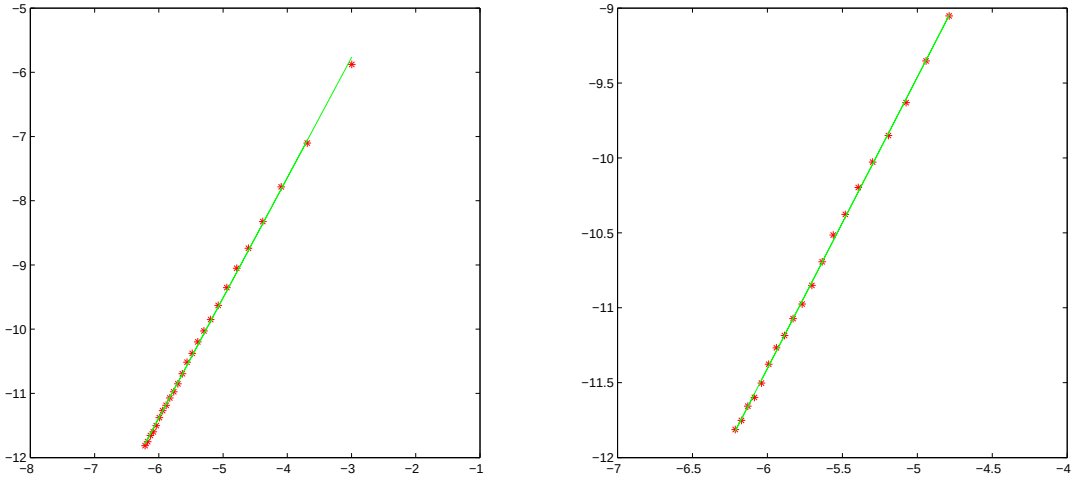

Figure 4.5: The linear regression analysis of $(u - u^h)$ in the L^∞ norm in log-log scale by using the conforming, linear, immersed interface finite element space. (a). the mesh is varying according to $N = 20 + 20k$, $k = 0, 1, \ldots, 24$, the slope (convergence order) is 1.8706; (b). $N = 120 + 20k$, $k = 0, 1, \ldots, 19$, the slope is 1.9443.

Chapter 5

IIFEM for Elasticity Interface Problems with Non-homogeneous Jump Conditions

In this chapter, we consider the elasticity interface problems with non-homogeneous jump conditions: $W = (w_1, w_2) \neq (0, 0)$ on Γ in (1.21) and (1.22), $Q = (q_1, q_2) \neq (0, 0)$ on Γ in (1.19) and (1.20). To construct a function with the *same non-homogeneous jump conditions* is more difficult than that for a single equation because four equations (1.21), (1.22), (1.19) and (1.20) are all coupled together.

Again, we extend the jump conditions along the normal directions of the interface Γ, then use the immersed-interface finite-element methods discussed in previous chapters to solve the elasticity interface problems with non-homogeneous jump conditions without losing the convergence rate. We also present some numerical examples.

5.1 Constructing the Function $\hat{u}$

In this section, we will use the same technique and notations as in Section 3.1, such as the signed distance function $\varphi(x)$, the unit normal n and the ρ-neighborhood of Γ in Ω $N(\Gamma, \rho)$, respectively, given in (3.5), (3.4) and (3.2). The Lemma 3.1 is also applicable here.

The functions $Q = (q_1, q_2)$ in (1.19), (1.20) and $W = (w_1, w_2)$ in (1.21), (1.22) are only defined on Γ. We need to extend $w_1 : \Gamma \to \mathbb{R}$, $w_2 : \Gamma \to \mathbb{R}$, $q_1 : \Gamma \to \mathbb{R}$ and $q_2 : \Gamma \to \mathbb{R}$ to $w_1^e : \Omega \to \mathbb{R}$, $w_2^e : \Omega \to \mathbb{R}$, $q_1^e : \Omega \to \mathbb{R}$ and $q_2^e : \Omega \to \mathbb{R}$, respectively, that satisfy the following:

E1. w_1^e, w_2^e, q_1^e and q_2^e are smooth on $\overline{\Omega}$;

E2. $w_1^e(x) = w_1(x^*)$, $w_2^e(x) = w_2(x^*)$, $q_1^e(x) = q_1(x^*)$ and $q_2^e(x) = q_2(x^*)$ for any $x \in N(\Gamma, \rho)$, where x^* is defined as in Lemma 3.1;

E3. $w_1^e(x) = g_1(x)$, $w_2^e(x) = g_2(x)$ for any $x \in \partial\Omega$, and $q_1^e(x) = 0$, $q_2^e(x) = 0$ for any $x \in \Omega \setminus N(\Gamma, \rho)$.

The existence of w_1^e, w_2^e, q_1^e and q_2^e are given in Section 3.2, so are the extension functions in detail.

We use the same idea in Section 3.3 to remove the source singularities in order to solve the elasticity interface problems with non-homogeneous jump conditions using the conforming, linear, immersed interface finite element space, but the way to construct the function $\hat{u} = (\hat{u}_1, \hat{u}_2)$ with the same non-homogeneous jump conditions for the elasticity interface problems is more complicated than that for the single elliptic interface problems in Section 3.3.

We first define $\tilde{u} = (\tilde{u}_1, \tilde{u}_2)$ by

$$\tilde{u}_1 = w_1^e(x) + J_1^e(x)\frac{\varphi(x)}{|\nabla\varphi(x)|} \qquad \forall x \in \Omega, \tag{5.1}$$

$$\tilde{u}_2 = w_2^e(x) + J_2^e(x)\frac{\varphi(x)}{|\nabla\varphi(x)|} \qquad \forall x \in \Omega, \tag{5.2}$$

where $J_1^e : \Omega \to \mathbb{R}$, $J_2^e : \Omega \to \mathbb{R}$ are the extensions of $J_1 : \Gamma \to \mathbb{R}$, $J_2 : \Gamma \to \mathbb{R}$.

Note that $J_1 : \Gamma \to \mathbb{R}$, $J_2 : \Gamma \to \mathbb{R}$, $J_1^e : \Omega \to \mathbb{R}$ and $J_2^e : \Omega \to \mathbb{R}$ are unknown functions and will be solved in this section. They satisfy the following:

A1. J_1^e and J_2^e are smooth on $\overline{\Omega}$, J_1 and J_2 are smooth on Γ;

A2. $J_1^e(x) = J_1(x^*)$ and $J_2^e(x) = J_2(x^*)$ for any $x \in N(\Gamma, \rho)$, where x^* is defined as in Lemma 3.1;

A3. $J_1^e(x) = 0$, $J_2^e(x) = 0$ for any $x \in \Omega \setminus N(\Gamma, \rho)$.

We also define $\hat{u} = (\hat{u}_1, \hat{u}_2)$ by adding the characteristic function χ_{Ω^+}:

$$\hat{u}_1 = \chi_{\Omega^+}(x)\tilde{u}_1(x) \qquad \forall x \in \Omega, \tag{5.3}$$

$$\hat{u}_2 = \chi_{\Omega^+}(x)\tilde{u}_2(x) \qquad \forall x \in \Omega. \tag{5.4}$$

We expect $\hat{u} = (\hat{u}_1, \hat{u}_2)$ has the following useful properties:

B1. $\hat{u}_1^+$, $\hat{u}_1^-$, $\hat{u}_2^+$ and $\hat{u}_2^-$ are smooth on Ω^- and Ω^+, respectively;

B2. $\hat{u}_1 = g_1$ and $\hat{u}_2 = g_2$ on $\partial\Omega$;

B3. $\hat{u} = (\hat{u}_1, \hat{u}_2)$ has same non-homogeneous jump conditions across the interface Γ as $u = (u_1, u_2)$:

$$[\hat{u}_1]_\Gamma = w_1, \tag{5.5}$$

$$[\hat{u}_2]_\Gamma = w_2, \tag{5.6}$$

$$\left[\lambda\left(\frac{\partial\hat{u}_1}{\partial x_1}+\frac{\partial\hat{u}_2}{\partial x_2}\right)n_1+2\mu\frac{\partial\hat{u}_1}{\partial x_1}n_1+\mu\left(\frac{\partial\hat{u}_1}{\partial x_2}+\frac{\partial\hat{u}_2}{\partial x_1}\right)n_2\right]_\Gamma = q_1, \tag{5.7}$$

$$\left[\lambda\left(\frac{\partial\hat{u}_1}{\partial x_1}+\frac{\partial\hat{u}_2}{\partial x_2}\right)n_2+2\mu\frac{\partial\hat{u}_2}{\partial x_2}n_2+\mu\left(\frac{\partial\hat{u}_1}{\partial x_2}+\frac{\partial\hat{u}_2}{\partial x_1}\right)n_1\right]_\Gamma = q_2. \tag{5.8}$$

We now solve the functions J_1 and J_2 by the following steps:

S1. Find the partial derivatives of $\tilde{u}_1$ and $\tilde{u}_2$. Note that $\varphi(x) = 0$ on the interface Γ, we have

$$\frac{\partial J_1^e}{\partial x_1}\left(\frac{\varphi}{|\nabla\varphi|}\right) = \frac{\partial J_1^e}{\partial x_2}\left(\frac{\varphi}{|\nabla\varphi|}\right) = \frac{\partial J_2^e}{\partial x_1}\left(\frac{\varphi}{|\nabla\varphi|}\right) = \frac{\partial J_2^e}{\partial x_2}\left(\frac{\varphi}{|\nabla\varphi|}\right) = 0,$$

then

$$\left.\frac{\partial\tilde{u}_1}{\partial x_1}\right|_\Gamma = \frac{\partial w_1^e}{\partial x_1}+J_1\frac{\partial}{\partial x_1}\left(\frac{\varphi(x)}{|\nabla\varphi(x)|}\right), \tag{5.9}$$

$$\left.\frac{\partial\tilde{u}_1}{\partial x_2}\right|_\Gamma = \frac{\partial w_1^e}{\partial x_2}+J_1\frac{\partial}{\partial x_2}\left(\frac{\varphi(x)}{|\nabla\varphi(x)|}\right), \tag{5.10}$$

$$\left.\frac{\partial\tilde{u}_2}{\partial x_1}\right|_\Gamma = \frac{\partial w_2^e}{\partial x_1}+J_2\frac{\partial}{\partial x_1}\left(\frac{\varphi(x)}{|\nabla\varphi(x)|}\right), \tag{5.11}$$

$$\left.\frac{\partial\tilde{u}_2}{\partial x_2}\right|_\Gamma = \frac{\partial w_2^e}{\partial x_2}+J_2\frac{\partial}{\partial x_2}\left(\frac{\varphi(x)}{|\nabla\varphi(x)|}\right); \tag{5.12}$$

S2. Obtain the following eqtions after we put (5.3), (5.4), (5.7) and (5.8) together:

$$\lambda^+ n_1\left(\frac{\partial\tilde{u}_1}{\partial x_1}+\frac{\partial\tilde{u}_2}{\partial x_2}\right)\Big|_\Gamma + 2\mu^+ n_1\left.\frac{\partial\tilde{u}_1}{\partial x_1}\right|_\Gamma + \mu^+ n_2\left(\frac{\partial\tilde{u}_1}{\partial x_2}+\frac{\partial\tilde{u}_2}{\partial x_1}\right)\Big|_\Gamma = q_1, \tag{5.13}$$

$$\lambda^+ n_2 \left(\frac{\partial \tilde{u}_1}{\partial x_1} + \frac{\partial \tilde{u}_2}{\partial x_2} \right) \Big|_\Gamma + 2\mu^+ n_2 \frac{\partial \tilde{u}_2}{\partial x_2} \Big|_\Gamma + \mu^+ n_1 \left(\frac{\partial \tilde{u}_1}{\partial x_2} + \frac{\partial \tilde{u}_2}{\partial x_1} \right) \Big|_\Gamma = q_2; \tag{5.14}$$

S3. Set equations for J_1, J_2 after we combine (5.9), (5.10), (5.11), (5.12), (5.13) and (5.14) together and rearrange some items:

$$a_{11}^J J_1 + a_{12}^J J_2 = b_1^J, \tag{5.15}$$

$$a_{21}^J J_1 + a_{22}^J J_2 = b_2^J, \tag{5.16}$$

where

$$\begin{aligned}
a_{11}^J &= (\lambda^+ + 2\mu^+) n_1 \frac{\partial}{\partial x_1} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right) + \mu^+ n_2 \frac{\partial}{\partial x_2} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right), \\
a_{12}^J &= \lambda^+ n_1 \frac{\partial}{\partial x_2} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right) + \mu^+ n_2 \frac{\partial}{\partial x_1} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right), \\
a_{21}^J &= \mu^+ n_1 \frac{\partial}{\partial x_2} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right) + \lambda^+ n_2 \frac{\partial}{\partial x_1} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right), \\
a_{22}^J &= \mu^+ n_1 \frac{\partial}{\partial x_1} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right) + (\lambda^+ + 2\mu^+) n_2 \frac{\partial}{\partial x_2} \left(\frac{\varphi(x)}{|\nabla \varphi(x)|} \right),
\end{aligned}$$

and

$$\begin{aligned}
b_1^J &= q_1 - (\lambda^+ + 2\mu^+) n_1 \frac{\partial w_1^e}{\partial x_1} - \lambda^+ n_1 \frac{\partial w_2^e}{\partial x_2} - \mu^+ n_2 \frac{\partial w_1^e}{\partial x_2} - \mu^+ n_2 \frac{\partial w_2^e}{\partial x_1}, \\
b_2^J &= q_2 - (\lambda^+ + 2\mu^+) n_2 \frac{\partial w_2^e}{\partial x_2} - \lambda^+ n_2 \frac{\partial w_1^e}{\partial x_1} - \mu^+ n_1 \frac{\partial w_1^e}{\partial x_2} - \mu^+ n_1 \frac{\partial w_2^e}{\partial x_1};
\end{aligned}$$

S4. Let A^J to be a 2×2 matrix and b^J, x^J to be 2×1 vectors, we have

$$A^J = \begin{pmatrix} a_{11}^J & a_{12}^J \\ a_{21}^J & a_{22}^J \end{pmatrix}, \qquad x^J = \begin{pmatrix} J_1 \\ J_2 \end{pmatrix}, \qquad b^J = \begin{pmatrix} b_1^J \\ b_2^J \end{pmatrix}, \tag{5.17}$$

then (5.15) and (5.16) can be written as

$$A^J x^J = b^J.$$

J_1 and J_2 can be solved by $x^J = (A^J)^{-1} b^J$.

Since **E1** and the equation (3.3), all entries of A^J and b^J are smooth, so J_1 and J_2 are also smooth on Γ. The existence of J_1^e, J_2^e which satisfy **A1**, **A2** and **A3** is given in Section 3.2, like q_1^e and q_2^e.

By **E1**, **A1** and (3.3), from (5.1) and (5.2), it is easy to see $\tilde{u}_1$ and $\tilde{u}_2$ are smooth functions. Moreover, $\hat{u}_1^+$ and $\hat{u}_2^+$ are smooth on Ω^+, obviously, $\hat{u}_1^- = 0$ and $\hat{u}_2^- = 0$ are smooth on Ω^-. Since **E3** and **A3**, $\hat{u}_1 = g_1$ and $\hat{u}_2 = g_2$ on $\partial\Omega$. By (3.1) , **E2** and **A2**,

$$[\hat{u}_1]_\Gamma = \hat{u}_1^+\big|_\Gamma - \hat{u}_1^-\big|_\Gamma = w_1,$$

$$[\hat{u}_2]_\Gamma = \hat{u}_2^+\big|_\Gamma - \hat{u}_2^-\big|_\Gamma = w_2.$$

The conditions (5.7) and (5.8) are hold by the procedures **S1-S4**. In this section, we construct $\hat{u} = (\hat{u}_1, \hat{u}_2)$ that has all expected properties: **B1**, **B2** and **B3**.

5.2 Removing Source Singularities and Modifying the Load Vector

The elasticity interface problems that we are dealing with in this chapter have non-homogeneous jump conditions. We still use the conforming finite element space we discussed in subsection 4.2.3 to solve u by letting $u = u^H + \hat{u}$, solving u^H and modifying the load vector of u^H to get the finite element solution of u.

Since u and $\hat{u}$ have the same jump conditions across the interface Γ:

$$[u]_\Gamma = [\hat{u}]_\Gamma = W,$$
$$[\sigma n]_\Gamma = [\hat{\sigma} n]_\Gamma = Q,$$

where $\hat{\sigma}$ is the stress tensor associated with $\hat{u}$, so $u^H = u - \hat{u}$ has the homogeneous jump conditions:

$$\left[u^H\right]_\Gamma = [u]_\Gamma - [\hat{u}]_\Gamma = W - W = 0, \tag{5.18}$$
$$\left[\sigma^H n\right]_\Gamma = [\sigma n]_\Gamma - [\hat{\sigma} n]_\Gamma = Q - Q = 0, \tag{5.19}$$

where σ^H is the stress tensor associated with u^H. The PDE

$$\nabla \cdot \sigma^H + F^H = 0 \tag{5.20}$$

can be written as

$$-\left\{(\lambda+2\mu)\frac{\partial^2 u_1^H}{\partial x_1^2}+(\lambda+\mu)\frac{\partial^2 u_2^H}{\partial x_1\partial x_2}+\mu\frac{\partial^2 u_1^H}{\partial x_2^2}\right\}=f_1^H, \tag{5.21}$$

$$-\left\{(\lambda+2\mu)\frac{\partial^2 u_2^H}{\partial x_2^2}+(\lambda+\mu)\frac{\partial^2 u_1^H}{\partial x_1\partial x_2}+\mu\frac{\partial^2 u_2^H}{\partial x_1^2}\right\}=f_2^H, \tag{5.22}$$

where the body force $F^H=(f_1^H,f_2^H)$ associated with u^H are given by

$$f_1^H=f_1+\left\{(\lambda+2\mu)\frac{\partial^2 \hat{u}_1}{\partial x_1^2}+(\lambda+\mu)\frac{\partial^2 \hat{u}_2}{\partial x_1\partial x_2}+\mu\frac{\partial^2 \hat{u}_1}{\partial x_2^2}\right\}, \tag{5.23}$$

$$f_2^H=f_2+\left\{(\lambda+2\mu)\frac{\partial^2 \hat{u}_2}{\partial x_2^2}+(\lambda+\mu)\frac{\partial^2 \hat{u}_1}{\partial x_1\partial x_2}+\mu\frac{\partial^2 \hat{u}_2}{\partial x_1^2}\right\}. \tag{5.24}$$

The weak form of the interface problem (5.21), (5.22), (5.18) and (5.19) is

$$\begin{aligned} a(u^H,v) &= \int_\Omega\left\{(\lambda+2\mu)\frac{\partial u_1^H}{\partial x_1}\frac{\partial v_1}{\partial x_1}+(\lambda+2\mu)\frac{\partial u_2^H}{\partial x_2}\frac{\partial v_2}{\partial x_2}+\mu\frac{\partial u_1^H}{\partial x_2}\frac{\partial v_1}{\partial x_2}+\mu\frac{\partial u_2^H}{\partial x_1}\frac{\partial v_2}{\partial x_1}\right. \\ &+ \left.\lambda\frac{\partial u_1^H}{\partial x_1}\frac{\partial v_2}{\partial x_2}+\lambda\frac{\partial u_2^H}{\partial x_2}\frac{\partial v_1}{\partial x_1}+\mu\frac{\partial u_1^H}{\partial x_2}\frac{\partial v_2}{\partial x_1}+\mu\frac{\partial u_2^H}{\partial x_1}\frac{\partial v_1}{\partial x_2}\right\}dx \\ &= L(v)=\int_\Omega\left(f_1^H v_1+f_2^H v_2\right)\,dx \qquad \forall v\in\{H_0^1(\Omega)\}^2. \end{aligned}$$

By applying the conforming, linear, immersed interface finite element method, the resulting linear system of u^H is $AU^H=F^{H_L}$, where F^{H_L} indicates the load vector associated with u^H to avoid the confusion with the body force F^H associated with

u^H. The entries of A and F^{H_L} are

$$\begin{aligned} a_{ij} &= \int_\Omega \Big\{ (\lambda+2\mu)\frac{\partial\phi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_1} + (\lambda+2\mu)\frac{\partial\psi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_2} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_2} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_1} \\ &+ \lambda\frac{\partial\phi_j}{\partial x_1}\frac{\partial\psi_i}{\partial x_2} + \lambda\frac{\partial\psi_j}{\partial x_2}\frac{\partial\phi_i}{\partial x_1} + \mu\frac{\partial\phi_j}{\partial x_2}\frac{\partial\psi_i}{\partial x_1} + \mu\frac{\partial\psi_j}{\partial x_1}\frac{\partial\phi_i}{\partial x_2} \Big\} dx, \end{aligned}$$

$$F_i^{H_L} = \int_\Omega \left(f_1^H\phi_i + f_2^H\psi_i \right) dx.$$

We define U, $\hat{U}$ and U^H to be $2N \times 1$ vectors:

$$U = (u_1^{(1)}, u_2^{(1)}, u_1^{(2)}, u_2^{(2)}, \ldots, u_1^{(i)}, u_2^{(i)}, \ldots, u_1^{(N)}, u_2^{(N)})^T,$$

$$\hat{U} = (\hat{u}_1^{(1)}, \hat{u}_2^{(1)}, \hat{u}_1^{(2)}, \hat{u}_2^{(2)}, \ldots, \hat{u}_1^{(i)}, \hat{u}_2^{(i)}, \ldots, \hat{u}_1^{(N)}, \hat{u}_2^{(N)})^T,$$

$$U^H = ((u_1^H)^{(1)}, (u_2^H)^{(1)}, (u_1^H)^{(2)}, (u_2^H)^{(2)}, \ldots, (u_1^H)^{(i)}, (u_2^H)^{(i)}, \ldots, (u_1^H)^{(N)}, (u_2^H)^{(N)})^T,$$

where $(u_1^{(i)}, u_2^{(i)})$,$((u_1^H)^{(i)}, (u_2^H)^{(i)})$ are the finite element solutions of u and u^H at i^{th} node (N_i), $i = 1, \ldots, N$ and $(\hat{u}_1^{(i)}, \hat{u}_2^{(i)})$ are the exact values of the function $\hat{u}$ at i^{th} node (N_i), $i = 1, \ldots, N$. By $U = U^H + \hat{U}$, we have

$$AU = A(U^H + \hat{U}) = AU^H + A\hat{U} = F^{H_L} + A\hat{U} = F.$$

For the elliptic interface problems with non-homogeneous jump conditions, we discussed in Section 3.5, the computation for the stiffness matrix remains to be the same as in the case of homogeneous jump conditions and the load vector needs to be modified for certain triangle near interface. Similarly, for the elasticity interface

problems with non-homogeneous jump conditions, we only need to adjust some entries of the load vector F based on F^{H_L} and A keeps same. The right hand side of the load vector can be summarized as the following,

$$F_i = \begin{cases} \displaystyle\int_\Omega (f_1\phi_i + f_2\psi_i)\, dx & \text{if } \Omega^s(\phi_i,\psi_i)\cap\Gamma = \emptyset, \\ \displaystyle\int_\Omega (f_1\phi_i + f_2\psi_i)\, dx & \\ \displaystyle + \int_\Omega \left\{(\lambda+2\mu)\frac{\partial^2\hat{u}_1}{\partial x_1^2} + (\lambda+\mu)\frac{\partial^2\hat{u}_2}{\partial x_1\partial x_2} + \mu\frac{\partial^2\hat{u}_1}{\partial x_2^2}\right\}\phi_i\, dx & \\ \displaystyle + \int_\Omega \left\{(\lambda+2\mu)\frac{\partial^2\hat{u}_2}{\partial x_2^2} + (\lambda+\mu)\frac{\partial^2\hat{u}_1}{\partial x_1\partial x_2} + \mu\frac{\partial^2\hat{u}_2}{\partial x_1^2}\right\}\psi_i\, dx & \\ \displaystyle + \sum_{j=1}^{2N} a_{ij}\hat{U}_j & \text{otherwise}, \end{cases} \tag{5.25}$$

where $\Omega^s(\phi_i,\psi_i)$ is the non-zero support region of ϕ_i and ψ_i.

5.3 Numerical Results

Example 5.1. *Elasticity interface problem with non-homogeneous jump conditions.*

We consider the problem (1.7)–(1.10) with $\Omega = (-1,1)\times(-1,1)$, Γ being the circle centered at point $(0,0)$ with radius $R = 0.5$, and $\lambda^- = \mu^- = 1$ and $\lambda^+ = \mu^+ = 100$. The exact solution $u = (u_1, u_2)$:

$$u_1 = \begin{cases} r^2 & \text{if } r \le R, \\ \dfrac{1}{b}\left\{r^2 + \log(r)\right\} & \text{otherwise}, \end{cases}$$

$$u_2 = \begin{cases} r^2 & \text{if } r \le R, \\ \dfrac{1}{b}\left\{r^2 + \sin(x_1)\cos(x_2)\right\} & \text{otherwise}, \end{cases}$$

where $r = \sqrt{x_1^2 + x_2^2}$ and $b = max\{(\lambda^+/\lambda^-), (\mu^+/\mu^-)\}$. The exact solutions have non-homogeneous jump conditions.

In Figure 5.1, we show the results of the errors of the finite element solution $(u - u^h)$ by using the conforming, linear, immersed interface finite element method in L^∞ norm. The convergence order from the sample meshes ranging from 20 to 500 with 20 increment is 1.9095, see Figure 5.1 (a). When cutting the results from coarse meshes, the convergence order increases to 2.0222, see Figure 5.1 (b). It shows clearly that the method has second order accuracy.

(a). (b).

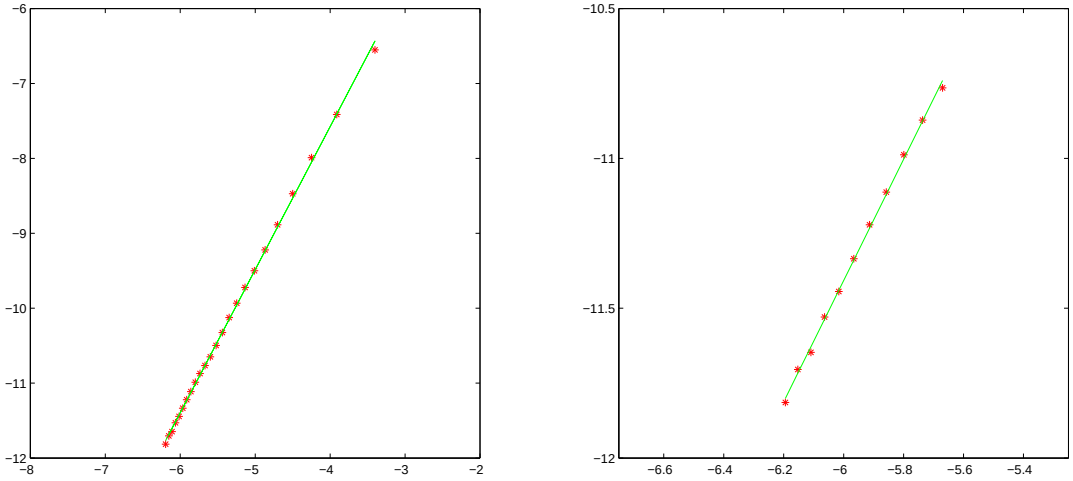

Figure 5.1: The linear regression analysis of $(u-u^h)$ in the L^∞ norm in log-log scale by using the conforming, linear, immersed interface finite element space. (a). the mesh is varying according to $N = 20 + 20k$, $k = 0, 1, \ldots, 24$, the slope (convergence order) is 1.9095; (b). $N = 300 + 20k$, $k = 0, 1, \ldots, 10$, the slope is 2.0222.

Chapter 6

Conclusions

We have developed a class of immersed-interface finite-element methods for solving elliptic and elasticity system interface problems with *homogeneous and non-homogeneous jump conditions*. The major contributions of this thesis are the following:

(a) We have developed conforming, piecewise quadratic, immerse-interface finite-element basis functions for irregular nodes that satisfy the homogeneous jump conditions for the elliptic interface problems;

(b) We have derived the immersed-interface finite -element method for the elliptic interface problems in which the non-homogeneous jump conditions are removed by using the level-set representation of the interface;

(c) We have constructed conforming, piecewise linear, immerse-interface finite-element basis functions for irregular nodes that satisfy the homogeneous jump conditions for the elasticity interface problems;

(d) We have presented a way to solve the elasticity interface problems with non-homogeneous jump conditions;

(e) We have several techniques of numerical implementation for the resulting finite-element equations.

Our methods have several advantages. For instance, the resultant linear system of equations is symmetric positive definite. Moreover, they can be coupled with the level-set method for fast simulations of interface dynamics. Our basic error analysis and numerical tests demonstrate that such methods have optimal convergence properties.

List of References

[1] L. M. Adams and Z. Li. The immersed interface/multigrid methods for interface problems. *SIAM J. Sci. Comput.*, 24:463–479, 2002.

[2] R. Adams. *Sobolev Spaces.* Academic Press, New York, 1975.

[3] J. E. Akin. *Application and Implementation of Finite Element Methods.* Academic Press, Orlando, FL, USA, 1982.

[4] A. Almgren, J. Bell, P. Collella, and T. Marthaler. A cartesian grid projection method for the incompressible Euler equations in complex geometries. *SIAM J. Sci. Comput.*, 18:1289–1309, 1997.

[5] E. Bänsch, F. Haußer, O. Lakkis, B. Li, and A. Voigt. Finite element method for epitaxial growth with attachment-detachment kinetics. *J. Comput. Phys.*, 194:409–434, 2004.

[6] E. B. Becker, G. F. Carey, and J. T. Oden. *Finite Elements: An Introduction.* Prentice Hall, 1981.

[7] D. Braess. *Finite Elements.* Cambridge University Press, 1997.

[8] W. K. Burton, N. Cabrera, and F. C. Frank. The growth of crystals and the equilibrium of their surfaces. *Phil. Trans. Roy. Soc. London Ser. A*, 243(866):299–358, 1951.

[9] R. E. Caflisch and B. Li. Analysis of island dynamics in epitaxial growth of thin films. *Multiscale Model. and Anal.*, 1:150–171, 2003.

[10] D. Calhoun. *A Cartesian grid method for solving the streamfunction-vorticity equations in irregular geometries*. PhD thesis, University of Washington, 1999.

[11] D. Calhoun. A cartesian grid method for solving the streamfunction-vorticity equation in irregular regions. *J. Comput. Phys.*, 176:231–275, 2002.

[12] Z. Chen and J. Zou. Finite element methods and their convergence for elliptic and parabolic interface problems. *Numer. Math.*, 79:175–202, 1998.

[13] A. Demlow and G. Dziuk. An adaptive finite element method for the laplace-beltrami operator on surfaces. *SIAM J. Numer. Anal.*, in press, 2007.

[14] Y. Gong, B. Li, and Z. Li. Immersed-interface finite-element methods for elliptic interface problems with non-homogeneous jump conditions. Preprint, 2007.

[15] P. Gremaud, C. Kuster, and Z. Li. A study of numerical methods for the level set approach. *Appl. Numer. Math.*, 57:837–846, 2007.

[16] D. V. Griffiths and I. M. Smith. *Numerical Methods for Engineers : A Programming Approach*. CRC Press, 1991.

[17] P. Grisvard. *Elliptic Problems in Nonsmooth Domains*. Pitman Advanced Publishing Program, 1985.

[18] J. K. Hastings, M. A. Juds, and J. R. Brauer. Accuracy and economy of finite element magnetic analysis. Proceedings of 33rd Annual National Relay Conference, 1985.

[19] S. Hou and X. Liu. A numerical method for solving variable coefficient elliptic equation with interfaces. *J. Comput. Phys.*, 202:411–445, 2005.

[20] J. N. Israelachvili. *Intermolecular and Surface Forces with Applications to Colloidal and Biological Systems*. Academic Press, 2nd edition, 1990.

[21] C. Johnson. *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge University Press, 1987.

[22] H. J. Jou, P. H. Leo, and J. S. Lowengrub. Microstructural evolution in inhomogeneous elastic media. *J. Comput. Phys.*, 131:109–148, 1997.

[23] D. R. Kincaid, J. R. Respess, D. M. Young, and R. G. Grimes. Itpack 2c: A fortran package for solving large sparse linear systems by adaptive accelerated iterative methods. http://rene.ma.utexas.edu/CNA/ITPACK/.

[24] R. J. LeVeque. Cartesian grid methods for flow in irregular regions. In K. W. Morton and M. J. Baines, editors, *Num. Meth. Fl. Dyn. III*, pages 375–382. Clarendon Press, 1988.

[25] R. J. LeVeque and Z. Li. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.*, 31:1019–1044, 1994.

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

[27] Z. Li. An overview of the immersed interface method and its applications. *Taiwanese J. Mathematics*, 7:1–49, 2003.

[28] Z. Li and K. Ito. Maximum principle preserving schemes for interface problems with discontinuous coefficients. *SIAM J. Sci. Comput.*, 23:1225–1242, 2001.

[29] Z. Li and K. Ito. *The Immersed Interface Method – Numerical Solutions of PDEs Involving Interfaces and Irregular Domains*. SIAM Frontier Series in Applied mathematics, FR33, 2006.

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

[31] Z. Li, W-C. Wang, I-L. Chern, and M-C. Lai. New formulations for interface problems in polar coordinates. *SIAM J. Sci. Comput.*, 25:224–245, 2003.

[32] Z. Li, H. Zhao, and H. Gao. A numerical study of electro-migration voiding by evolving level set functions on a fixed cartesian grid. *J. Comput. Phys.*, 152:281–304, 1999.

[33] R. H. Macneal. *Finite Elements: Their Design and Performance*. Marcel Dekker, 1994.

[34] Young's modulus. Mcgraw-hill dictionary of scientific and technical terms. McGraw-Hill Companies, Inc., 2003. Answers.com 28 Jul. 2007. http://www.answers.com/topic/young-s-modulus.

[35] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, New York, 2002.

[36] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, Cambridge, UK, 1988.

[37] Poisson's ratio. Mcgraw-hill dictionary of scientific and technical terms. McGraw-Hill Companies, Inc., 2003. Answers.com 28 Jul. 2007. http://www.answers.com/topic/poisson-s-ratio.

[38] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 2nd edition, 1999.

[39] I. M. Smith and D. V. Griffiths. *Programming the Finite Element Method*. John Wiley and Sons, 4th edition, 2004.

[40] J. Stoer and R. Bulirsch. *Introduction to Numerical Analysis*. Springer-Verlag, New York, 1980.

[41] M. J. Turner, R. W. Clough, H. C. Martin, and L. C. Topp. Stiffness and deflection analysis of complex structures. *Journal of the Aeronautical Sciences*, 23:805–823, 1956.

[42] Wikipedia. Finite element analysis — wikipedia, the free encyclopedia, 2007. Online; accessed 28-July-2007.

[43] Wikipedia. Interpolation — wikipedia, the free encyclopedia, 2007. Online; accessed 28-July-2007.

[44] Wikipedia. Level set method — wikipedia, the free encyclopedia, 2007. Online; accessed 28-July-2007.

[45] Wikipedia. Regular grid — wikipedia, the free encyclopedia, 2007. Online; accessed 28-July-2007.

[46] X. Yang. *The Immersed Interface Method for Elasticity Problems with Interfaces.* PhD thesis, North Carolina State University, 2004.

[47] Z. Yang. *A Cartesian grid method for elliptic boundary value problems in irregular regions.* PhD thesis, University of Washington, 1996.

[48] D. Young and D. Kincaid. The ITPACK package for large sparse linear systems. In M. Schultz et al., editors, *Elliptic Problem Solvers*, pages 163–185. Academic Press, 1981.