Gradient recovery for elliptic interface problem: II. Immersed finite element methods

Hailong Guo, Xu Yang*

Department of Mathematics, University of California Santa Barbara, CA, 93106, United States

A R T I C L E   I N F O

Article history:
Received 30 July 2016
Received in revised form 4 January 2017
Accepted 3 March 2017
Available online 9 March 2017

Keywords:
Elliptic interface problem
Immersed finite element method
Gradient recovery
Superconvergence
A posteriori error estimator

A B S T R A C T

This is the second paper on the study of gradient recovery for elliptic interface problem. In our previous work Guo and Yang (2016) [17], we developed a novel gradient recovery technique for finite element method based on the body-fitted mesh. In this paper, we propose new gradient recovery methods for two immersed interface finite element methods: symmetric and consistent immersed finite method (Ji et al. (2014) [23]) and Petrov–Galerkin immersed finite element method (Hou et al. (2004) [22], and Hou and Liu (2005) [20]). Compared to the body-fitted mesh based gradient recovery method, the new methods provide a uniform way of recovering gradient on regular meshes. Numerical examples are presented to confirm the superconvergence of both gradient recovery methods. Moreover, they provide asymptotically exact a posteriori error estimators for both immersed finite element methods.

Published by Elsevier Inc.

1. Introduction

We are interested in developing gradient recovery methods for the following elliptic interface problem

\[-\nabla \cdot (\beta(z)\nabla u(z)) = f(z), \quad \text{in } \Omega \setminus \Gamma,\]
\[u = 0, \quad \text{on } \partial \Omega,\]  

where \(\Omega\) is a bounded polygonal domain with Lipschitz boundary \(\partial \Omega\) in \(\mathbb{R}^2\), and \(\Gamma\) is the interface which splits \(\Omega\) into two disjoint subdomains \(\Omega^-\) and \(\Omega^+\). Note that the interface \(\Gamma\) can be given by a zero level of level set function [33,38].

The interface problem is characterized by the following piecewise smooth diffusion coefficient \(\beta(z) \geq \beta_0\),

\[\beta(z) = \begin{cases} 
\beta^-(z) & \text{if } z \in \Omega^- , \\
\beta^+(z) & \text{if } z \in \Omega^+ ,
\end{cases}\]

which has a finite jump of function value across the interface \(\Gamma\). We consider homogeneous jump conditions at the interface \(\Gamma\) as below,

\[|u|^- = u^+ - u^- = 0,\]
\[|\beta \partial_n u|^+ = \beta^+ u_n^+ - \beta^- u_n^- = 0,\]

where \(\partial_n u = \nabla u \cdot n\) denotes the normal flux with \(n\) being the unit outer normal vector of the interface \(\Gamma\).

* Corresponding author.
E-mail addresses: hlguo@math.ucsb.edu (H. Guo), xuyang@math.ucsb.edu (X. Yang).

http://dx.doi.org/10.1016/j.jcp.2017.03.003
0021-9991/Published by Elsevier Inc.
Simulation of the interface problem (1.1)–(1.5) is an important problem in the fields of fluid dynamics and material science, where the background is composed by rather different materials. Discontinuities of coefficients at interfaces lead to nonsmooth solutions in general, and thus raise a challenge for designing efficient numerical methods for (1.1)–(1.5).

Two main streams of existing numerical methods for (1.1)–(1.5) are body-fitted mesh-based methods and immersed boundary/interfacing methods. Body-fitted mesh-based methods resolve discontinuities by generating mesh grids to align with the interface, and then use standard finite element methods. This type of methods can provide high order accuracy, with nearly optimal error estimates established in, for example, [2,4,9,41]. Despite its merit of accuracy, the main drawback of such methods is the requirement of a body-fitted mesh generator, which can be technically involved and time-consuming especially when the geometry of interface becomes complicated. Therefore, it will be more convenient to develop numerical methods based unfitted mesh (e.g. Cartesian mesh). A rich literature can be found in this direction including immersed boundary method (IBM) by Peskin [34,35] and immersed interface method (IIM) by Leveque and Li [25], just to name a few.

In IBM, Dirac δ-function is used to model discontinuity and discretize to distribute a singular source to the nearest grid point. In IIM, a special finite difference scheme is constructed near the interface to get an accurate approximation of the solution. Moreover, IIM was also developed in the framework of finite element method [26,29,28]. Interested readers are referred to [27] for a review of this type of methods. In [28], Li, Lin, and Wu proposed a nonconforming immersed finite element method (IFEM) by modifying the basis functions on elements crossing interface. Chou et al. established optimal error estimates in $L^2$ and $H^1$ norms in [13]. However, it only achieved first order (suboptimal) convergence in $L^\infty$ norm due to discontinuities of test functions. To overcome this drawback, Ji, Chen, and Li added a correction term into the bilinear form of the nonconforming IFEM to penalize the discontinuities at interface [23], which showed optimal convergence rate in $L^2$ and $H^1$ norms. They also numerically verified that the method achieved second order convergence in $L^\infty$ norm. Another weak form formulation was derived in [20–22] based on the Petrov–Galerkin method for the discretization of elliptic interface problem, which has been numerically verified to have optimal convergence rate in $L^2$, $H^1$ and $L^\infty$ norms.

Superconvergence analysis of elliptic interface problem has been a challenging problem due to the lack of regularity of solution at the interface. Standard gradient recovery methods [43,44,42,32,1,8] only work well for elliptic problems with a smooth coefficient. As far as we know, only limited work has been done in the development of gradient recovery methods for elliptic interface problem. For example, [11,12] proposed two special interpolation formulas to recover flux for linear and quadratic immersed finite element method in one-dimension. A more recent work [39] showed a supercloseness between finite element solution and linear interpolation of the true solution for linear finite element method based on the body-fitted mesh. In our previous work [17], we developed an improved polynomial preserving recovery (IPPR) method based on the body-fitted mesh and proved its superconvergence for both mildly unstructured and adaptive refined meshes.

As a continuous study of [17], we propose new gradient recovery methods in this paper based on two immersed finite element methods: symmetric and consistent immersed finite element (SCIFEM) [23] and Petrov–Galerkin immersed finite element method (PGIFEM) [20–22]. The development of the methods is based on the following two observations: firstly, the solution is piecewise smooth on each subdomain despite its low global regularity; secondly, finite element solution is discontinuous at interface even though the exact solution is continuous. Accordingly, we design the gradient recovery methods by two steps: enriching and smoothing. We first define an enriching operator to enrich the discontinuous finite element solution into continuous one on a local body-fitted mesh obtained by adding extra nodes [28]. Such type of enriching operator has been well studied for nonconforming finite element and plays an important role in a priori error estimates [16] and convergence analysis of multigrid methods [5–7]. Then we apply the IPPR gradient recovery operator developed in [17] to the enriched finite element solution. We prove that the proposed gradient recovery operator is a bounded linear operator, and numerically verify that the recovered gradient is $O(h^{1.5})$ superconvergent to the exact gradient. As a byproduct, we observe the $O(h^{1.5})$ supercloseness between finite element solution and linear interpolation of true solution for both SCIFEM [23] and PGIFEM [20–22].

The rest of the paper is organized as follows. In Section 2, we briefly review two immersed finite element methods, SCIFEM and PGIFEM, as a preparation for designing gradient recovery methods. In Section 3, we first define an enriching operator and prove several properties of the operator. Then, we propose the gradient recovery methods for SCIFEM and PGIFEM and prove that the gradient recovery operator is a linear, bounded and consistent operator. In Section 4, several numerical examples are presented to confirm the superconvergence of the gradient recovery methods. We make conclusive remarks in Section 5.

2. Review on immersed finite element methods

In this section, we briefly review two immersed finite element methods, symmetric and consistent immersed finite element method [23] and Petrov–Galerkin immersed finite element method [20–22], based on which we shall develop superconvergent gradient recovery methods for elliptic interface problem (1.1)–(1.5) in Section 3.

2.1. Notations

We first summarize the notations that will be used in this paper. We will use standard notations for Sobolev spaces and their associate norms given in [8,14,15]. For a subdomain $A$ of $\Omega$, let $P_m(A)$ be the space of polynomials of degree less than or equal to $m$ in $A$ and $n_m$ be the dimension of $P_m(A)$ which equals to $\frac{1}{2}(m+1)(m+2)$. $W^{k,p}(A)$ denotes the Sobolev
space with norm $\| \cdot \|_{k,p,A}$ and seminorm $| \cdot |_{k,p,A}$. When $p = 2$, $W^{k,2}(A)$ is simply denoted by $H^k(A)$ and the subscript $p$ is omitted in its associate norm and seminorm. As in $[39]$, denote $W^{k,p}(\Omega^- \cup \Omega^+)$ as the function space consisting of piecewise Sobolev function $w$ such that $w|_{\Omega^-} \in W^{k,p}(\Omega^-)$ and $w|_{\Omega^+} \in W^{k,p}(\Omega^+)$. For the function space $W^{k,p}(\Omega^- \cup \Omega^+)$, define its associated norm as

$$\| w \|_{k,p,\Omega^- \cup \Omega^+} = \left( \| w \|_{k,p,\Omega^-}^p + \| w \|_{k,p,\Omega^+}^p \right)^{1/p},$$

and associated seminorm as

$$| w |_{k,p,\Omega^- \cup \Omega^+} = \left( | w |_{k,p,\Omega^-}^p + | w |_{k,p,\Omega^+}^p \right)^{1/p}.$$

Let $C$ denote a generic positive constant which may be different at different occurrences. For the sake of simplicity, we use $x \lesssim y$ to mean that $x \leq Cy$ for some constant $C$ independent of mesh size and the location of the interface.

Without loss of generality, we simply suppose $T_h$ is a uniform triangulation of $\Omega$ with $h = \text{diam}(T)$. Assume $h$ is small enough so that the interface $\Gamma$ never crosses any edge of $T_h$ more than two times. The elements of $T_h$ can be divided into two categories: regular elements and interface elements. We call an element $T$ an interface element if the interface $\Gamma$ passes the interior of $T$; otherwise, we call it a regular element. Remark that if $\Gamma$ only passes two vertices of an element $T$, we treat the element $T$ as a regular element. Let $T_h^r$ and $T_h^i$ denote the set of all interface elements and regular elements respectively. The set of all vertices of $T_h$ is denoted by $N_h$.

2.2. Variational formula

The variational formulation to elliptic interface problem (1.1)–(1.5) is given by finding $u \in H^1_0(\Omega)$ such that

$$(\beta \nabla u, \nabla v) = (f, v), \quad \forall v \in H^1_0(\Omega),$$

where $(\cdot, \cdot)$ is standard $L^2$-inner product in the spaces $L^2(\Omega)$. By the positiveness of $\beta$, Lax–Milgram Theorem implies (2.1) has a unique solution. $[39, 37]$ proved that $u \in H^r(\Omega^- \cup \Omega^+)$ for $0 \leq r \leq 2$ and

$$\| u \|_{r,\Omega^- \cup \Omega^+} \lesssim \| f \|_{0,\Omega} + \| g \|_{r-3/2,\Gamma},$$

if $f \in L^2(\Omega)$ and $g \in H^{r-3/2}(\Gamma)$.

2.3. Immersed finite element methods

The key idea of immersed interface methods is to construct special basis functions in interface elements to incorporate jump conditions (1.4) and (1.5). As an illustration, we consider a typical interface element $T$ as in Fig. 1. Let $z_4$ and $z_5$ be the intersection points between the interface $\Gamma$ and edges of the element. Connect the line segment $z_4z_5$ and it forms an approximation of interface $\Gamma$ in the element $T$, denoted by $\Gamma_h|_T$. Then the element $T$ is split into two parts: $T^-$ and $T^+$. The special basis $\phi_i$ on the interface element $T$ is constructed as the following piecewise linear function

$$\phi_1(z) = \begin{cases} \phi_1^{+} = a^++b^+x+c^+y, & z = (x, y) \in T^+, \\ \phi_1^{-} = a^- + b^-x + c^-y, & z = (x, y) \in T^-, \end{cases}$$

where the coefficients are determined by the following linear system
\[ \phi_i(z_1) = \delta_{i1}, \phi_i(z_2) = \delta_{i2}, \phi_i(z_3) = \delta_{i3}, \tag{2.4} \]

\[ \phi_i^+(z_4) = \phi_i^-(z_4), \phi_i^+(z_5) = \phi_i^-(z_5), \beta^+ \partial_u \phi_i^+ = \beta^- \partial_u \phi_i^- \tag{2.5} \]

for \( i = 1, 2, 3 \). The immersed finite element space \( V_h \) [28] is defined as

\[ V_h := \{ v \in V_h : v|_T \in V_h(T) \text{ and } v \text{ is continuous on } \mathcal{N}_h \}, \tag{2.6} \]

\[ V_{h,0} = \{ v \in V_h : v(z) = 0 \text{ for all } z \in \mathcal{N}_h \cap \partial \Omega \}, \tag{2.7} \]

where

\[ V_h(T) := \begin{cases} \{ v | v \in P_1(T) \}, & \text{if } T \in T_h^i; \\ \{ v | v \text{ is defined by (2.3)} \}, & \text{if } T \in T_h^i. \tag{2.8} \end{cases} \]

Note that in general \( V_h \) is a nonconforming finite element space and [29] shows it has optimal approximation capability.

### 2.3.1. Symmetric and consistent immersed finite element method

Let \( \mathcal{E}_h \) denote the set of all edges in \( T_h \), and then \( \mathcal{E}_h \) consists of interface edges \( \mathcal{E}_h^i \) and regular edges \( \mathcal{E}_h^r \), defined by

\[ \mathcal{E}_h^i = \{ e \in \mathcal{E}_h : e \cap \Gamma \neq \emptyset \}, \mathcal{E}_h^r = \mathcal{E}_h \setminus \mathcal{E}_h^i. \tag{2.9} \]

For any interior edge \( e \), there exist two triangles \( T_1 \) and \( T_2 \) such that \( T_1 \cap T_2 = e \). Denote \( n_e \) as the unit normal of \( e \) pointing from \( T_1 \) to \( T_2 \), and define

\[ \{ \nabla u \} = \frac{1}{2} (\nabla u|_{T_1} + \nabla u|_{T_2}), \tag{2.10} \]

\[ [u] = u|_{T_1} - u|_{T_2}. \tag{2.11} \]

The symmetric and consistent immersed finite element method (SCIFEM) [23] seeks \( u_h^{sc} \in V_{h,0} \) such that

\[ a_h^{sc}(u_h^{sc}, v_h) = (f, v_h), \quad \forall v_h \in V_{h,0}, \tag{2.12} \]

where

\[ a_h^{sc}(u, v) = \sum_{T \in T_h} \int_{\partial T} \beta \nabla u \cdot \nabla v d\nu + \sum_{e \in \mathcal{E}_h^i} \int_e (\{ \beta \nabla u \} \cdot \{ \beta \nabla v \}) \cdot n_e d\sigma. \tag{2.13} \]

In [23], Ji, Chen, and Li showed the bilinear form (2.13) was consistent and numerically verified its coercivity. Moreover, [23] proved the following convergence results:

**Theorem 2.1.** Let \( u \) be the solution of (1.1)–(1.5) and \( u_h \) be the solution of (2.12). Then the following error estimates hold:

\[ \left( \sum_{T \in T_h} |u - u_h^{sc}|_{H^1(T)}^2 \right)^{1/2} \lesssim h \| u \|_{2, \Omega^+ \cup \Omega^-}, \tag{2.14} \]

\[ \| u - u_h^{sc} \|_{0, \Omega} \lesssim h^{2} \| u \|_{2, \Omega^+ \cup \Omega^-}. \tag{2.15} \]

**Remark 2.2.** The main difference between SCIFEM and classical immersed finite element method [28] is that the bilinear form of SCIFEM (2.13) contains one more term to penalize the discontinuous of basis function at the intersecting points of interface and edge. Numerical results in [23] show that SCIFEM has \( O(h^2) \) convergence in \( L^\infty \)-norm.

#### 2.3.2. Petrov–Galerkin immersed finite element method

Denote the standard \( C^0 \) linear finite element space on \( T_h \) by \( S_h \) and \( S_{h,0} = S_h \cap H^1_0(\Omega) \). Then the Petrov–Galerkin immersed finite element method (PGIFEM) [22,20,21] is to find \( u_h^{pg} \in V_{h,0} \) such that

\[ a_h(u_h^{pg}, v_h) = (f, v_h), \quad \forall v_h \in S_{h,0}, \tag{2.16} \]

where

\[ a_h(u, v) = \sum_{T \in T_h} \int_{\partial T} \beta \nabla u \cdot \nabla v d\nu. \tag{2.17} \]

**Remark 2.3.** To our best knowledge, there has been no analytical results on estimating PGIFEM, however, plenty of numerical simulations indicate that it can achieve optimal convergence rate in both \( L^2, H^1 \) and \( L^\infty \) norms [22,20,21].
3. Gradient recovery for immersed finite element methods

In the section, we systematically introduce gradient recovery methods for SCIFEM and PGIFEM reviewed in the last section. We first define an enriching operator, and then apply the improved polynomial preserving recovery operator [17] to the enriched finite element solution.

3.1. Enriching operator

To define the enriching operator, one needs to generate a local body-fitted mesh \( \hat{T}_h \) based on \( T_h \) by adding new vertices into \( \Lambda_h \) which divides interface element into three subtriangles. Then the new triangulation is constructed as below [28]:

1. Keep all regular elements unchanged.
2. For each interface element \( T \), split it into a small triangle and a quadrilateral by connecting two intersection points, and then divide the quadrilateral into two subtriangles by an auxiliary line connecting a vertex and an intersection point. The choice of auxiliary line is made so that there at least exists one angle between \( \frac{\pi}{4} \) and \( \frac{3\pi}{4} \) in the two new subtriangles.

Remark 3.1. Note that the new triangulation may contain narrow triangles, and thus standard linear finite element method deteriorates on \( \hat{T}_h \). However, the propose of introducing the body-fitted mesh \( \hat{T}_h \) is just for enriching existing immersed finite element solution instead of solving interface problem directly on it.

Let \( \hat{X}_h \) be the \( C^0 \) linear finite element space defined on \( \hat{T}_h \). We construct an enriching operator \( E_h : V_h \to \hat{X}_h \) by averaging the discontinuous values at intersection points. Let \( \hat{N}_h \) denote all vertices in \( \hat{T}_h \), and one has \( N_h \subset \hat{N}_h \). For any \( z \in \hat{N}_h \), let \( \hat{T}_z \) denote the set of all triangles in \( \hat{T}_h \) having \( z \) as their vertex and define

\[
(E_h v)(z) = \frac{1}{|\hat{T}_z|} \sum_{\hat{T} \in \hat{T}_z} v_{\hat{T}}(z),
\]

with \(|\hat{T}_z|\) being the cardinality of \( \hat{T}_z \) and \( v_{\hat{T}} = v|_{\hat{T}} \). We can define \( E_h v \) on \( \Omega \) by standard linear finite element interpolation in \( \hat{X}_h \) after obtaining the values \((E_h v)(z)\) at all vertices. It is easy to see that \((E_h v)(z) = v(z)\) for all \( z \in \hat{N}_h \cap N_h \), which means \((E_h v)(z) = v(z)\) for all \( z \in \hat{N}_h \) provided that \( v \) is continuous.

Remark 3.2. The purpose of the enriching operator is to make the discontinuous immersed finite element solution become continuous as the true solution.

For the enriching operator \( E_h \), we can prove the following error estimate.

Theorem 3.3. For any \( v \in V_h \), one has

\[
\sum_{T \in \hat{T}_h} \| E_h v - v \|^2_{0,T} \lesssim h^2 \sum_{T \in T_h} |v|^2_{1,T}.
\]

Proof. For any \( z \in \hat{N}_h \setminus N_h \), there exists an \( e \in E^1_h \) so that \( z \in e \). Let \( T_1 \) and \( T_2 \) be the two triangles in \( T_h \) so that \( T_1 \cap T_2 = e \). Then \( \hat{T} \subset T_1 \) or \( \hat{T} \subset T_2 \) for any \( \hat{T} \in \hat{T}_h \). Hence \( v_{\hat{T}}(z) = v_{T_1}(z) \) or \( v_{\hat{T}}(z) = v_{T_2}(z) \). Then for any \( \hat{T} \), \( \hat{T} \in \hat{T}_h \), we can find \( p \in N_h \) such that

\[
|v_{\hat{T}}(z) - v_{\hat{T}_h}(z)|^2 \\
\leq |v_{T_1}(z) - v_{T_2}(z)|^2 \\
\leq |v_{T_1}(z) - v_{T_1}(p)|^2 + |v_{T_1}(p) - v_{T_2}(p)|^2 + |v_{T_2}(p) - v_{T_2}(z)|^2 \\
= |v_{T_1}(z) - v_{T_1}(p)|^2 + |v_{T_2}(p) - v_{T_2}(z)|^2 \\
\lesssim |v|^2_{1,T_1 \cup T_2},
\]

where we have used the fact that \( v_{T_1}(p) = v_{T_2}(p) \) since \( p \in N_h \) in the first equality and the mean value theorem [7] in the last inequality. Combining (3.1) and (3.3) gives, for any \( \hat{T} \in \hat{T}_h \),

\[
|(E_h v - v_{\hat{T}})(z)|^2 \lesssim |v|^2_{1,T_1 \cup T_2}, \forall v \in V_h,
\]

which implies that
\[ \|E_h v - v\|^2_{0,T} \leq |\hat{T}| \sum_{z \in \mathcal{A}(\hat{T})} \left( (E_h v - v)(z) \right)^2 \]
\[ \lesssim h^2 \sum_{T \in \mathcal{T}(\hat{T})} |v|_{1,T}^2. \]  
(3.5)

where \( \mathcal{T}(\hat{T}) = \{ T \in T_h : T \cap \hat{T} \neq \emptyset \} \). Taking summation over all \( \hat{T}_h \) produces the inequality (3.2). \( \square \)

**Corollary 3.4.** For any \( v \in V_h \), we have
\[ \|E_h v\|_{0,\Omega} \lesssim \|v\|_{0,\Omega}. \]  
(3.6)
\[ |E_h v|_{1,\Omega} \lesssim |v|_{1,\Omega}. \]  
(3.7)

**Proof.** We first prove the inequality (3.6). Notice that
\[ \|E_h v\|_{0,\Omega} \lesssim \|E_h v - v\|_{0,\Omega} + \|v\|_{0,\Omega} \]
\[ \lesssim h\|\nabla v\|_{0,\Omega} + \|v\|_{0,\Omega} \]
\[ \lesssim \|v\|_{0,\Omega}. \]  
(3.8)

where we have used the standard inverse estimate [14, 8] in the last inequality. Using (3.2) and standard inverse estimate yields
\[ |E_h v|_{1,\Omega} \leq |E_h v - v|_{1,\Omega} + |v|_{1,\Omega} \]
\[ \lesssim h^{-1}\|E_h v - v\|_{0,\Omega} + |v|_{1,\Omega} \]
\[ \lesssim |v|_{1,\Omega}. \]  
(3.9)

which completes our proof. \( \square \)

3.2. Gradient recovery operator

The edges of \( \hat{T}_h \) with both ending points lying on \( \Gamma \) form an approximation of the interface \( \Gamma \), denoted by \( \Gamma_h \), then the triangulation \( \hat{T}_h \) is divided into the following two disjoint sets by \( \Gamma_h \):
\[ \hat{T}_h^- := \left\{ T \in \hat{T}_h \mid \text{all three vertices of } T \text{ are in } \overline{\Omega^-} \right\}, \]  
(3.10)
\[ \hat{T}_h^+ := \left\{ T \in \hat{T}_h \mid \text{all three vertices of } T \text{ are in } \overline{\Omega^+} \right\}. \]  
(3.11)

Suppose \( \tilde{X}_h^- \) and \( \tilde{X}_h^+ \) are the continuous linear finite element spaces defined on \( \hat{T}_h^- \) and \( \hat{T}_h^+ \) respectively.

Let \( G_h^I : \tilde{X}_h \to (\tilde{X}_h^- \cup \tilde{X}_h^+) \times (\tilde{X}_h^- \cup \tilde{X}_h^+) \) be the improved polynomial preserving recovery (IPPR) operator introduced in [17]. Let \( u_h \) be the solution of either symmetric and consistent immersed finite element method or Petrov–Galerkin immersed finite element method. The recovered gradient of \( u_h \) is defined as
\[ R_h u_h = G_h^I(E_h u_h). \]  
(3.12)

**Remark 3.5.** The proposed gradient recovery method consists of two steps: firstly, we enrich the immersed finite element solution by the enriching operator; then we recover the gradient of the enriched solution.

**Remark 3.6.** The gradient recovery method requires doing a least-squares fitting at very vertex of \( \mathcal{T}_h \) with computation cost of order \( O(1) \). Hence, the total computational cost of recovery procedure is of order \( O(N) \). It can be ignored compared to the cost of solving original problem.

It is easy to see that \( R_h \) is a linear operator from \( V_h \) to \( (\tilde{X}_h^- \cup \tilde{X}_h^+) \times (\tilde{X}_h^- \cup \tilde{X}_h^+) \), and one can prove the following boundedness results.

**Theorem 3.7.** Denote \( R_h \) to be the recovered operator defined in (3.12), and then
\[ \|R_h u_h\|_{0,\Omega^- \cup \Omega^+} \lesssim |u_h|_{1,h}. \]  
(3.13)
**Proof.** By the definition of IPPR recovery operator in [17], we have

\[ \| R_h u_h \|_{0, \Omega^-} = \| C_h^h E_h u_h \|_{0, \Omega^-} \lesssim \| E_h u_h \|_{1, \Omega^-}, \]

and

\[ \| R_h u_h \|_{0, \Omega^+} = \| C_h^h E_h u_h \|_{0, \Omega^+} \lesssim \| E_h u_h \|_{1, \Omega^+}. \]

Then the estimate follows by that

\[ \| R_h u_h \|_{0, \Omega^- \cup \Omega^+} \lesssim \| R_h u_h \|_{0, \Omega^-} + \| R_h u_h \|_{0, \Omega^+} \]

\[ \lesssim \| E_h u_h \|_{1, \Omega^-} + \| E_h u_h \|_{1, \Omega^+} \]

\[ \lesssim \| E_h u_h \|_{1, \Omega} \]

\[ \lesssim \| u_h \|_{1, \Omega}, \]

where we have used Corollary 3.4. □

**Theorem 3.7** implies \( R_h \) is a linear bounded operator. Moreover, we have the following consistency result:

**Theorem 3.8.** Let \( R_h : V_h \to (\hat{X}_h^- \cup \hat{X}_h^+) \times (\hat{X}_h^- \cup \hat{X}_h^+) \) be the gradient recovery operator defined in (3.12). Given \( u \in H^3(\Omega^- \cup \Omega^+) \cap C^0(\Omega) \), one has

\[ \| R_h u_I - \nabla u \|_{0, \Omega} \lesssim h^2 \| u \|_{3, \Omega^- \cup \Omega^+}, \]

where \( u_I \) is interpolation of \( u \) into linear finite element space \( \hat{X}_h \).

**Proof.** Since \( u \in C^0(\Omega) \), one has that \( u_I \in C^0(\Omega) \) and then \( E_h u_I = u_I \). Therefore, we have \( R_h u_I = G_h^I E_h u_I = G_h u_I \). Theorem 3.6 in [17] implies that

\[ \| R_h u_I - \nabla u \|_{0, \Omega} = \| G_h^I u_I - \nabla u \|_{0, \Omega} \lesssim h^2 \| u \|_{3, \Omega^- \cup \Omega^+}, \]

which completes our proof. □

**Remark 3.9.** **Theorem 3.8** implies \( R_h \) is consistent. In addition, it is a local gradient recovery operator. Therefore, \( R_h \) satisfies the three conditions of a good gradient recovery operator described in [1], and should serve as an ideal candidate of gradient recovery operator for both SCIFEM and PGIFEM.

**Remark 3.10.** One of the most practical applications of gradient recovery techniques is to construct asymptotically exact \( a \) posteriori error estimators \([1, 3, 19, 31, 43, 44]\) for adaptive computational methods. Based on the recovery operator \( R_h \), one can define a local \( a \) posteriori error estimator on element \( T \in T_h \) as

\[ \eta_T = \begin{cases} \| \beta^{1/2} (R_h u_h - \nabla u_h) \|_{0, T}, & \text{if } T \in T_h^+, \\ \left( \frac{\sum_{T \subset T} \| \beta^{1/2} (R_h u_h - \nabla u_h) \|_{0, T}^2}{\sum_{T \subset T} |T|^2} \right)^{1/2}, & \text{if } T \in T_h^-. \end{cases} \]

and the corresponding global error estimator as

\[ \eta_h = \left( \sum_{T \in T_h} \eta_T^2 \right)^{1/2}, \]

which provides an asymptotically exact \( a \) posteriori error estimator for SCIFEM and PGIFEM. The readers are referred to [10, 40] for residual-type \( a \) posteriori error estimator for immersed finite element methods.

4. Numerical results

In the section, we give several numerical examples to verify the superconvergence of gradient recovery methods for both SCIFEM and PGIFEM. The computational domain of the first four examples are chosen as \( \Omega = [-1, 1] \times [-1, 1] \). The uniform triangulation of \( \Omega \) is obtained by dividing \( \Omega \) into \( N^2 \) subsquares and then dividing each subsquare into two right triangles. In the first four tests, we take \( N = 2^k \) with \( k = 5, 6, 7, 8, 9, 10, 11 \). In the last example, we consider a nonlinear interface problem with homogeneous jump conditions on a rectangle domain with a hole. For convenience, we shall use the following error norms in all examples:

\[ D e := \| u - u_h \|_{1, \Omega}, \quad D^1 e := \| \nabla u_I - \nabla u_h \|_{0, \Omega}, \quad D^e r := \| \nabla u - R_h u_h \|_{0, \Omega}. \]
Example 4.1. In this example, we consider the elliptic interface problem (1.1) with a circular interface of radius $r_0 = 0.6$ as studied in [28]. The exact solution is

$$u(z) = \begin{cases} 
\frac{r^2}{r^2} & \text{if } z \in \Omega_-, \\
\frac{1}{r^2} - \frac{1}{r^3} & \text{if } z \in \Omega_+. 
\end{cases}$$

where $r = \sqrt{x^2 + y^2}$.

Tables 1–6 show the numerical results of both SCIFEM and PGIFEM with three typical different jump ratios: $\beta^-/\beta^+ = 1/10$ (moderate jump), $\beta^-/\beta^+ = 1/1000$ (large jump), and $\beta^-/\beta^+ = 1000$ (large jump). In all different cases, optimal $O(h)$ convergence can be observed for $H^1$-semi error of finite element solution, which consists with the numerical results in [23, 20, 22]. The recovered gradient superconverges to the exact gradient at a rate of $O(h^{1.5})$. Moreover, we numerically observe the supercloseness between gradient of the finite element solution and its finite element interpolation for both SCIFEM and PGIFEM; see column 5 of Tables 1–6.
Table 5
Numerical results of SCIFEM for Examples 4.1 with $\beta^+ = 1, \beta^- = 1000$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$D_e$</th>
<th>Order</th>
<th>$D\delta_e$ Order</th>
<th>$D\delta_e$ Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>1.95e-01</td>
<td>-</td>
<td>1.35e-02</td>
<td>-</td>
</tr>
<tr>
<td>64</td>
<td>9.79e-02</td>
<td>1.00</td>
<td>3.60e-03</td>
<td>1.91</td>
</tr>
<tr>
<td>128</td>
<td>4.90e-02</td>
<td>1.00</td>
<td>1.48e-03</td>
<td>1.28</td>
</tr>
<tr>
<td>256</td>
<td>2.45e-02</td>
<td>1.00</td>
<td>5.56e-04</td>
<td>1.42</td>
</tr>
<tr>
<td>512</td>
<td>1.23e-02</td>
<td>1.00</td>
<td>1.81e-04</td>
<td>1.61</td>
</tr>
<tr>
<td>1024</td>
<td>6.13e-03</td>
<td>1.00</td>
<td>6.44e-05</td>
<td>1.49</td>
</tr>
<tr>
<td>2048</td>
<td>3.06e-03</td>
<td>1.00</td>
<td>2.33e-05</td>
<td>1.47</td>
</tr>
</tbody>
</table>

Table 6
Numerical results of PGIFEM for Example 4.1 with $\beta^+ = 1, \beta^- = 1000$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$D_e$</th>
<th>Order</th>
<th>$D\delta_e$ Order</th>
<th>$D\delta_e$ Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>5.95e-02</td>
<td>-</td>
<td>3.27e-02</td>
<td>-</td>
</tr>
<tr>
<td>64</td>
<td>2.91e-02</td>
<td>1.03</td>
<td>9.35e-03</td>
<td>1.81</td>
</tr>
<tr>
<td>128</td>
<td>1.44e-02</td>
<td>1.02</td>
<td>4.03e-03</td>
<td>1.21</td>
</tr>
<tr>
<td>256</td>
<td>7.10e-03</td>
<td>1.02</td>
<td>1.45e-03</td>
<td>1.48</td>
</tr>
<tr>
<td>512</td>
<td>3.53e-03</td>
<td>1.01</td>
<td>5.68e-04</td>
<td>1.35</td>
</tr>
<tr>
<td>1024</td>
<td>1.76e-03</td>
<td>1.01</td>
<td>1.90e-04</td>
<td>1.58</td>
</tr>
<tr>
<td>2048</td>
<td>8.76e-04</td>
<td>1.00</td>
<td>6.80e-05</td>
<td>1.48</td>
</tr>
</tbody>
</table>

Example 4.2. In this example, we consider the elliptic interface problem (1.1) with shape edge as in [23,24]. The level set function of the interface is $\phi = -y^2 + ((x-1)\tan(\theta))^2$ with $\theta$ being a parameter. The interface is displayed in Fig. 2(a).

The right hand function $f$ is chosen to fit the exact solution $u(x, y) = \phi(x, y)/\beta$.

Numerically we test the case $\beta^- = 1$ and $\beta^+ = 1000$ when $\theta = 40$. The corresponding numerical results are shown in Tables 7 and 8, from which one can see that $D_e$ decays at a optimal rate of $O(h)$, while $D\delta_e$ and $D\delta_e$ tend to zero at a superconvergent rate of $O(h^{1.5})$. Fig. 2(b) plots the numerical solution of PGIFEM on the coarsest mesh and Fig. 3 shows the recovered gradient.

![Fig. 2](image-url)

Table 7
Numerical results of SCIFEM for Examples 4.2.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$D_e$</th>
<th>Order</th>
<th>$D\delta_e$ Order</th>
<th>$D\delta_e$ Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>3.04e-02</td>
<td>-</td>
<td>7.01e-03</td>
<td>-</td>
</tr>
<tr>
<td>64</td>
<td>1.54e-02</td>
<td>0.98</td>
<td>4.63e-03</td>
<td>0.60</td>
</tr>
<tr>
<td>128</td>
<td>7.44e-03</td>
<td>1.05</td>
<td>8.71e-04</td>
<td>2.41</td>
</tr>
<tr>
<td>256</td>
<td>3.71e-03</td>
<td>1.01</td>
<td>3.55e-04</td>
<td>1.30</td>
</tr>
<tr>
<td>512</td>
<td>1.85e-03</td>
<td>1.00</td>
<td>1.27e-04</td>
<td>1.49</td>
</tr>
<tr>
<td>1024</td>
<td>9.24e-04</td>
<td>1.00</td>
<td>4.31e-05</td>
<td>1.56</td>
</tr>
<tr>
<td>2048</td>
<td>4.62e-04</td>
<td>1.00</td>
<td>1.55e-05</td>
<td>1.48</td>
</tr>
</tbody>
</table>
Fig. 3. Plots of recovered gradient based on PGIFEM for Examples 4.2 with $\beta^+ = 10$, $\beta^- = 1$: (a) $x$-component; (b) $y$-component.

Table 8
Numerical results of PGIFEM for Examples 4.2.

<table>
<thead>
<tr>
<th>N</th>
<th>$D e$</th>
<th>Order</th>
<th>$D^3 e$</th>
<th>Order</th>
<th>$D^5 e$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>3.57e−02</td>
<td>–</td>
<td>1.98e−02</td>
<td>–</td>
<td>2.09e−02</td>
<td>–</td>
</tr>
<tr>
<td>64</td>
<td>1.70e−02</td>
<td>1.08</td>
<td>8.37e−03</td>
<td>1.24</td>
<td>8.09e−03</td>
<td>1.37</td>
</tr>
<tr>
<td>128</td>
<td>7.96e−03</td>
<td>1.09</td>
<td>2.95e−03</td>
<td>1.50</td>
<td>2.73e−03</td>
<td>1.57</td>
</tr>
<tr>
<td>256</td>
<td>3.82e−03</td>
<td>1.06</td>
<td>9.97e−04</td>
<td>1.57</td>
<td>9.41e−04</td>
<td>1.54</td>
</tr>
<tr>
<td>512</td>
<td>1.88e−03</td>
<td>1.02</td>
<td>3.72e−04</td>
<td>1.42</td>
<td>3.54e−04</td>
<td>1.41</td>
</tr>
<tr>
<td>1024</td>
<td>9.32e−04</td>
<td>1.02</td>
<td>1.29e−04</td>
<td>1.53</td>
<td>1.24e−04</td>
<td>1.51</td>
</tr>
<tr>
<td>2048</td>
<td>4.64e−04</td>
<td>1.01</td>
<td>4.57e−05</td>
<td>1.50</td>
<td>4.31e−05</td>
<td>1.52</td>
</tr>
</tbody>
</table>

Table 9
Numerical results of SCIFEM for Examples 4.3.

<table>
<thead>
<tr>
<th>N</th>
<th>$D e$</th>
<th>Order</th>
<th>$D^3 e$</th>
<th>Order</th>
<th>$D^5 e$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>1.19e+00</td>
<td>–</td>
<td>1.61e−01</td>
<td>–</td>
<td>1.93e−01</td>
<td>–</td>
</tr>
<tr>
<td>64</td>
<td>5.93e−01</td>
<td>1.00</td>
<td>5.98e−02</td>
<td>1.43</td>
<td>7.24e−02</td>
<td>1.42</td>
</tr>
<tr>
<td>128</td>
<td>2.96e−01</td>
<td>1.00</td>
<td>2.14e−02</td>
<td>1.48</td>
<td>2.69e−02</td>
<td>1.43</td>
</tr>
<tr>
<td>256</td>
<td>1.48e−01</td>
<td>1.00</td>
<td>7.80e−03</td>
<td>1.46</td>
<td>9.66e−03</td>
<td>1.48</td>
</tr>
<tr>
<td>512</td>
<td>7.41e−02</td>
<td>1.00</td>
<td>2.75e−03</td>
<td>1.50</td>
<td>3.49e−03</td>
<td>1.47</td>
</tr>
<tr>
<td>1024</td>
<td>3.70e−04</td>
<td>1.00</td>
<td>9.84e−04</td>
<td>1.48</td>
<td>1.23e−03</td>
<td>1.51</td>
</tr>
<tr>
<td>2048</td>
<td>1.85e−02</td>
<td>1.00</td>
<td>3.49e−04</td>
<td>1.50</td>
<td>4.37e−04</td>
<td>1.49</td>
</tr>
</tbody>
</table>

Table 10
Numerical results of PGIFEM for Examples 4.3.

<table>
<thead>
<tr>
<th>N</th>
<th>$D e$</th>
<th>Order</th>
<th>$D^3 e$</th>
<th>Order</th>
<th>$D^5 e$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>1.19e+00</td>
<td>–</td>
<td>1.55e−01</td>
<td>–</td>
<td>1.89e−01</td>
<td>–</td>
</tr>
<tr>
<td>64</td>
<td>5.93e−01</td>
<td>1.00</td>
<td>5.81e−02</td>
<td>1.42</td>
<td>7.19e−02</td>
<td>1.39</td>
</tr>
<tr>
<td>128</td>
<td>2.96e−01</td>
<td>1.00</td>
<td>2.09e−02</td>
<td>1.48</td>
<td>2.66e−02</td>
<td>1.43</td>
</tr>
<tr>
<td>256</td>
<td>1.48e−01</td>
<td>1.00</td>
<td>7.61e−03</td>
<td>1.45</td>
<td>9.61e−03</td>
<td>1.48</td>
</tr>
<tr>
<td>512</td>
<td>7.41e−02</td>
<td>1.00</td>
<td>2.75e−03</td>
<td>1.50</td>
<td>3.49e−03</td>
<td>1.47</td>
</tr>
<tr>
<td>1024</td>
<td>3.70e−04</td>
<td>1.00</td>
<td>9.84e−04</td>
<td>1.48</td>
<td>1.23e−03</td>
<td>1.51</td>
</tr>
<tr>
<td>2048</td>
<td>1.85e−02</td>
<td>1.00</td>
<td>3.49e−04</td>
<td>1.50</td>
<td>4.37e−04</td>
<td>1.49</td>
</tr>
</tbody>
</table>

Example 4.3. In the example, we consider the elliptic interface problem (1.1) with ellipse interface given by the zero level set of the function $\phi(x, y) = x^2 + \frac{y^2}{0.5^2} - 1$ as studied in [23,24]. Here, we choose the case of variable coefficient $\beta(x, y)$ as

$$
\beta(x, y) = \begin{cases}
1 + 0.5(x^2 - xy + y^2) & \text{if } (x, y) \in \Omega^-, \\
n & \text{if } (x, y) \in \Omega^+.
\end{cases}
$$

The right hand side function $f$ and boundary condition are given by the exact solution $u(x, y) = \phi(x, y) / \beta(x, y)$.

Tables 9 and 10 list the numerical errors, which provide a verification of the $O(h)$ convergence for semi-$H^1$ error, and $O(h^{1.5})$ supercloseness and superconvergence.
Example 4.4. In this example, we consider the interface problem (1.1) with a cardioid interface as in [20]. The interface curve $\Gamma$ is the zero level of the function

$$\phi(x, y) = (3(x^2 + y^2) - x)^2 - x^2 - y^2,$$

as shown Fig. 4(a). We choose the exact solution $u(x, y) = \phi(x, y)/\beta(x, y)$, where

$$\beta(x, y) = \begin{cases} xy + 3 & \text{if } (x, y) \in \Omega^-, \\ 100 & \text{if } (x, y) \in \Omega^+. \end{cases}$$

As pointed in [20], the difficulty of the problem is that the interface is not even Lipschitz-continuous and has a singular point at the origin. Fig. 4(b) plots the numerical solution of PGIFEM and Fig. 5 shows the recovered gradient. The numerical errors are given in Tables 11 and 12, from which, one can also observe the optimal convergence and superconvergence for both SCIFEM and PGIFEM even though the interface is not Lipschitz-continuous.

Example 4.5. In this example, we consider the following nonlinear interface problem

$$-\nabla \cdot (\beta(z) \nabla u(z)) + u^2 = f(z), \quad z \in \Omega \setminus \Gamma,$$

with homogeneous jump conditions (1.4) and (1.5) where $\Omega = [-2, 2] \times [-2, 2] \setminus [-0.5, 0.5] \times [-0.5, 0.5]$. The interface curve $\Gamma$ is a circle centered at origin with the radius $r_0 = \frac{\pi}{4}$. The exact solution is
Table 11
Numerical results of SCIFEM for Examples 4.4.

<table>
<thead>
<tr>
<th>N</th>
<th>(D_e)</th>
<th>Order</th>
<th>(D_f^e)</th>
<th>Order</th>
<th>(D_f^{\pm})</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>5.59e−02</td>
<td>−</td>
<td>9.84e−03</td>
<td>−</td>
<td>2.51e−02</td>
<td>−</td>
</tr>
<tr>
<td>64</td>
<td>2.86e−02</td>
<td>0.96</td>
<td>4.09e−03</td>
<td>1.27</td>
<td>7.98e−03</td>
<td>1.65</td>
</tr>
<tr>
<td>128</td>
<td>1.47e−02</td>
<td>0.98</td>
<td>1.57e−03</td>
<td>1.38</td>
<td>2.30e−03</td>
<td>1.80</td>
</tr>
<tr>
<td>256</td>
<td>7.39e−03</td>
<td>0.99</td>
<td>5.72e−04</td>
<td>1.46</td>
<td>7.21e−04</td>
<td>1.67</td>
</tr>
<tr>
<td>512</td>
<td>3.71e−03</td>
<td>0.99</td>
<td>2.05e−04</td>
<td>1.48</td>
<td>2.41e−04</td>
<td>1.58</td>
</tr>
<tr>
<td>1024</td>
<td>1.86e−03</td>
<td>1.00</td>
<td>7.25e−05</td>
<td>1.50</td>
<td>8.99e−05</td>
<td>1.42</td>
</tr>
<tr>
<td>2048</td>
<td>9.31e−04</td>
<td>1.00</td>
<td>2.54e−05</td>
<td>1.51</td>
<td>3.15e−05</td>
<td>1.51</td>
</tr>
</tbody>
</table>

Table 12
Numerical results of PGIFEM for Examples 4.4.

<table>
<thead>
<tr>
<th>N</th>
<th>(D_e)</th>
<th>Order</th>
<th>(D_f^e)</th>
<th>Order</th>
<th>(D_f^{\pm})</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>6.09e−02</td>
<td>−</td>
<td>2.48e−02</td>
<td>−</td>
<td>4.06e−02</td>
<td>−</td>
</tr>
<tr>
<td>64</td>
<td>3.01e−02</td>
<td>1.01</td>
<td>9.06e−03</td>
<td>1.45</td>
<td>1.36e−02</td>
<td>1.58</td>
</tr>
<tr>
<td>128</td>
<td>1.50e−02</td>
<td>1.01</td>
<td>3.32e−03</td>
<td>1.45</td>
<td>4.25e−03</td>
<td>1.68</td>
</tr>
<tr>
<td>256</td>
<td>7.48e−03</td>
<td>1.00</td>
<td>1.16e−03</td>
<td>1.51</td>
<td>1.45e−03</td>
<td>1.55</td>
</tr>
<tr>
<td>512</td>
<td>3.73e−03</td>
<td>1.00</td>
<td>4.13e−04</td>
<td>1.49</td>
<td>4.87e−04</td>
<td>1.57</td>
</tr>
<tr>
<td>1024</td>
<td>1.87e−03</td>
<td>1.00</td>
<td>1.44e−04</td>
<td>1.52</td>
<td>1.74e−04</td>
<td>1.49</td>
</tr>
<tr>
<td>2048</td>
<td>9.32e−04</td>
<td>1.00</td>
<td>5.11e−05</td>
<td>1.49</td>
<td>6.11e−05</td>
<td>1.51</td>
</tr>
</tbody>
</table>

Table 13
Numerical results of SCIFEM for Examples 4.5.

<table>
<thead>
<tr>
<th>Dof</th>
<th>(D_e)</th>
<th>Order</th>
<th>(D_f^e)</th>
<th>Order</th>
<th>(D_f^{\pm})</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>459</td>
<td>1.77e−01</td>
<td>−</td>
<td>2.01e−02</td>
<td>−</td>
<td>7.41e−02</td>
<td>−</td>
</tr>
<tr>
<td>1737</td>
<td>8.58e−02</td>
<td>0.55</td>
<td>6.38e−03</td>
<td>0.86</td>
<td>1.44e−02</td>
<td>1.23</td>
</tr>
<tr>
<td>6750</td>
<td>4.31e−02</td>
<td>0.51</td>
<td>2.37e−03</td>
<td>0.73</td>
<td>3.79e−03</td>
<td>0.99</td>
</tr>
<tr>
<td>26604</td>
<td>2.14e−02</td>
<td>0.51</td>
<td>8.12e−04</td>
<td>0.78</td>
<td>1.11e−03</td>
<td>0.89</td>
</tr>
<tr>
<td>105624</td>
<td>1.06e−02</td>
<td>0.51</td>
<td>2.82e−04</td>
<td>0.77</td>
<td>3.55e−04</td>
<td>0.83</td>
</tr>
<tr>
<td>420912</td>
<td>5.31e−03</td>
<td>0.50</td>
<td>9.71e−05</td>
<td>0.77</td>
<td>1.24e−04</td>
<td>0.76</td>
</tr>
<tr>
<td>1680480</td>
<td>2.65e−03</td>
<td>0.50</td>
<td>3.46e−05</td>
<td>0.75</td>
<td>4.33e−05</td>
<td>0.76</td>
</tr>
</tbody>
</table>

Table 14
Numerical results of PGIFEM for Examples 4.5.

<table>
<thead>
<tr>
<th>Dof</th>
<th>(D_e)</th>
<th>Order</th>
<th>(D_f^e)</th>
<th>Order</th>
<th>(D_f^{\pm})</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>459</td>
<td>1.83e−01</td>
<td>−</td>
<td>4.40e−02</td>
<td>−</td>
<td>6.99e−02</td>
<td>−</td>
</tr>
<tr>
<td>1737</td>
<td>8.73e−02</td>
<td>0.56</td>
<td>1.58e−02</td>
<td>0.77</td>
<td>1.43e−02</td>
<td>1.19</td>
</tr>
<tr>
<td>6750</td>
<td>4.35e−02</td>
<td>0.51</td>
<td>5.61e−03</td>
<td>0.76</td>
<td>4.27e−03</td>
<td>0.89</td>
</tr>
<tr>
<td>26604</td>
<td>2.15e−02</td>
<td>0.51</td>
<td>1.98e−03</td>
<td>0.76</td>
<td>1.35e−03</td>
<td>0.84</td>
</tr>
<tr>
<td>105624</td>
<td>1.07e−02</td>
<td>0.51</td>
<td>7.21e−04</td>
<td>0.73</td>
<td>4.75e−04</td>
<td>0.76</td>
</tr>
<tr>
<td>420912</td>
<td>5.32e−03</td>
<td>0.50</td>
<td>2.52e−04</td>
<td>0.76</td>
<td>1.63e−04</td>
<td>0.77</td>
</tr>
<tr>
<td>1680480</td>
<td>2.66e−03</td>
<td>0.50</td>
<td>8.96e−05</td>
<td>0.75</td>
<td>5.65e−05</td>
<td>0.77</td>
</tr>
</tbody>
</table>

\[ u(z) = \begin{cases} 
\frac{\log(r)}{\beta^e}, & \text{if } z \in \Omega_-, \\
\frac{\log(r)}{\beta^e} + \left(\frac{1}{\beta^-} - \frac{1}{\beta^e}\right) \log\left(\frac{r}{r_0}\right), & \text{if } z \in \Omega_+, 
\end{cases} \]

where \(r = \sqrt{x^2 + y^2}\). The right hand side function \(f\) and boundary condition are determined by the exact solution.

To generate initial uniform mesh, we first construct a uniform mesh on the domain \([-2, 2] \times [-2, 2]\) with mesh size \(h = \frac{1}{4}\) and then delete the parts on the domain \([-0.5, 0.5] \times [-0.5, 0.5]\). The other five level uniform meshes are obtained from uniformly refining the initial mesh. The discretized nonlinear problem is solved by Newton’s method. Tables 13 and 14 show the numerical results for SCIFEM and PGIFEM with \(\beta^-/\beta^+ = 1/1000\) respectively. Note Dof \(\approx h^{-2}\) for a two dimensional grid, the corresponding convergent rates with respect to the mesh size \(h\) are twice as many as what we present in the Tables 13 and 14. We can observe the same superconvergence and supercloseness results as linear problems on uniform meshes.

5. Conclusion

In this paper, we develop gradient recovery methods for both symmetric consistent immersed finite method and Petrov–Galerkin immersed finite element method. Theoretically, we prove that the proposed gradient recovery operator has consis-
tency, localization, and boundedness properties. The superconvergence of recovered gradient is confirmed by five numerical examples using both piecewise constant and piecewise variable diffusion coefficients. Moreover, we numerically observe the supercloseness between immersed finite element solution and the linear interpolation of exact solution. Compared to body-fitted mesh-based gradient recovery methods, the proposed gradient recovery methods provide a uniform way of recovering gradient on regular meshes. One of our ongoing research project is to provide a theoretic justification for the observed superconvergence and supercloseness phenomenon. We also remark that recently there have been very interesting studies on elliptic interface problem by Professor Zhilin Li and his collaborators based on the idea of mixed finite element method [30,36], which produce the same accurate gradient at interface as our proposed method.

Acknowledgements

This work was partially supported by the NSF grant DMS-1418936, KI-Net NSF RNMS grant 1107291, and Hellman Family Foundation Faculty Fellowship, UC Santa Barbara. Part of work was done during the visit of both authors to Beijing Computational Science Research Center, and we really appreciate their hospitality.

References