A Bilinear Immersed Finite Volume Element Method for the Diffusion Equation with Discontinuous Coefficient†

X.-M. He1, T. Lin1,∗ and Y. Lin2,3

1 Department of Mathematics, Virginia Tech, Blacksburg, VA 24060, USA.
2 Department of Mathematical and Statistical Science, University of Alberta, Edmonton AB, T6G 2G1, Canada.
3 Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong.

Received 17 December 2007; Accepted (in revised version) 6 June 2008
Available online 24 November 2008

Abstract. This paper is to present a finite volume element (FVE) method based on the bilinear immersed finite element (IFE) for solving the boundary value problems of the diffusion equation with a discontinuous coefficient (interface problem). This method possesses the usual FVE method’s local conservation property and can use a structured mesh or even the Cartesian mesh to solve a boundary value problem whose coefficient has discontinuity along piecewise smooth nontrivial curves. Numerical examples are provided to demonstrate features of this method. In particular, this method can produce a numerical solution to an interface problem with the usual $O(h^2)$ (in $L^2$ norm) and $O(h)$ (in $H^1$ norm) convergence rates.

AMS subject classifications: 65N15, 65N30, 65N50, 35R05

Key words: Interface problems, immersed interface, finite volume element, discontinuous coefficient, diffusion equation.

1 Introduction

In many applications, a simulation domain is often formed by several materials separated by curves or surfaces from each other, and this often leads to the so called interface problem consisting of the usual boundary value problem of the diffusion equation, the usual boundary condition, plus jump conditions across the material interface.

†This paper is dedicated to Richard E. Ewing on the occasion of his 60th birthday.
∗Corresponding author. Email addresses: xiaoming@vt.edu (X.-M. He), tlin@vt.edu (T. Lin), yanlin@ualberta.ca (Y. Lin)
required by pertinent physics. It is well known that efficiently solving this type of inter-
face problem is critical in many applications of engineering and sciences, including
flow problems [10, 11, 27, 29, 30, 43, 52], electromagnetic problems [4, 16, 61, 66–70, 78],
shape/topology optimization problems [13–15, 29, 36–38, 46, 74], and the modeling of
nonlinear phenomena [41, 79, 86], to name just a few. In this paper, we present a finite
volume element method with bilinear immersed finite element (IFE) [39, 59] for solving
this popular interface problem. This method possesses both the advantages of finite vol-
ume element method and those of IFE. In particular, this method can use a Cartesian
mesh to solve a boundary value problem with a discontinuous coefficient whose inter-
face consists of nontrivial piecewise smooth curves.

![Figure 1: A sketch of the domain for the interface problem.](image)

To be specific, we consider the following boundary value problem:

\[
- \nabla \cdot (\beta \nabla u) = f, \quad (x,y) \in \Omega, \\
u|_{\partial \Omega} = g.
\]

Here, see the sketch in Fig. 1, without loss of generality, we assume that \( \Omega \subset IR^2 \) is a
rectangular domain, the interface \( \Gamma \) is a curve separating \( \Omega \) into two sub-domains \( \Omega^- \),
\( \Omega^+ \) such that \( \Omega = \Omega^- \cup \Omega^+ \cup \Gamma \), and the coefficient \( \beta(x,y) \) is a piecewise constant function
defined by

\[
\beta(x,y) = \begin{cases} 
\beta^-, & (x,y) \in \Omega^- \\
\beta^+, & (x,y) \in \Omega^+ 
\end{cases}
\]

Because of the discontinuity in the coefficient \( \beta \), jump conditions are also imposed on the
interface \( \Gamma \):

\[
[u]|_\Gamma = 0, \quad (1.3) \\
[\beta \frac{\partial u}{\partial n}]|_\Gamma = 0. \quad (1.4)
\]

Of course, conventional numerical methods can be used to solve interface problem
(1.1)-(1.4). Standard discretization techniques such as finite difference (FD), see [73] and
references therein, finite volume (FV), see [40] and references therein, and finite element

... references therein, are all applicable, provided that meshes used by these methods are tailored to resolve the interfaces, see Fig. 2. Otherwise, the lack of smoothness in the exact solution across the interface will prevent a numerical method to perform satisfactorily [9, 17, 25]. In general, this requires a conventional method to use a mesh such that each of its element is basically occupied by one of the materials, and consequently, this prevents the usage of a structured mesh such as a Cartesian mesh if the interface is nontrivial. Therefore, conventional methods have limitations for them to solve interface problems efficiently in many applications. As an incomplete list of their limitations, we first note that for applications with moving interfaces, the meshes used by these methods have to be regenerated again and again according to current location of the interfaces at the moment the interface problems have to be solved. Second, there are many applications in which structured meshes are preferred, for example, Particle In Cell method for Plasma Particle Simulation, see [48, 62] and references therein. Last, but not the least, we note that the algebraic system based on a structured mesh often requires much less computational time to solve because efficient algebraic solvers such as fast FFT and multigrid can be easily implemented.

Figure 2: The plot on the left shows how elements are placed along an interface in a standard FE method. Each of the elements is essentially on one side of the interface. An element not allowed in a standard FE method is illustrated by the plot on the right.

To alleviate these limitations, FD methods are modified by reformulating the interface problem in elements cut through by interface or by employing finite difference stencils sophisticated enough to capture the discontinuity at the nodes in the neighborhood of the interface. Along this direction, Peskin’s immersed boundary method [71, 72] is one of the early representative ideas followed by many publications with applications in numerous fields [3, 21, 26, 28, 31, 35, 42, 45, 49–51, 53, 55, 65, 80, 81, 84, 85]. For FE methods, special local basis functions have been developed to handle the interface jump conditions in elements cut through by interface. Early work can be found in [8] and [6, 7], which have developed basis functions for treating rough coefficients. Further development can be found in the partition of unity methods and the extended finite element methods (X-FEMs) [12, 64, 77]. Another class of FE methods along this idea are the recently introduced immersed finite element (IFE) methods [1, 2, 22, 33, 39, 47, 48, 54, 57–60, 76]. More references can also be found in [56].

We like to point out that the early works by Babuska et al. in [5–7] proposed and analyzed several classes of finite element methods for interface problems, in particular for
linear element in one-dimension, these are equivalent or identical to the linear immersed finite elements, therefore the work [54] is just a special case of [5–7] with somewhat different settings. However for high order elements in one-dimension the immersed finite element proposed in [1, 2, 59, 60] and etc are different from those in [5–7].

In IFE methods, standard finite element functions are used in elements occupied by one of the materials, but piecewise polynomials patched by interface jump conditions are employed in elements formed by multiple materials. Particularly, the meshes used by IFE methods can be independent of the interface; hence, structured meshes, even the Cartesian mesh, can be used to solve boundary value problems with rather sophisticated interfaces between materials.

Up to now, IFE has been applied to solve interface problems in the Galerkin formulation, see [1, 2, 22, 47, 54, 57–60]. On the other hand, finite volume element (FVE) has the local conservation property which is very much desired in many applications, see [18–20, 23, 24, 32, 34, 44, 63, 75, 82] and related reference therein. We believe that the combination of the FVE’s local conservation property and IFE’s flexibility to handle interface jump conditions without using complicated meshes can generate competitive numerical methods for solving interface problems.

The rest of this article is organized as follows. In Section 2, we recall the definition of the bilinear IFE space to be used and its basic features. In Section 3, we present the finite volume element method based on this bilinear IFE space. In Section 4, we present several numerical examples to demonstrate features of this immersed FVE method. The conclusion is given in Section 5.

2 The bilinear immersed finite element space

In this section, we recall the bilinear IFE space discussed in [39, 59]. We will also list some of its basic properties and refer reader to the references above for more details.

Let \( T_h, h > 0 \) be a family of rectangular meshes of the solution domain \( \Omega \) that can be a union of rectangles. For each mesh \( T_h \), we let

\[
N_h = \{ X_i, \quad i = 1, 2, \ldots, N \}
\]

be the set of its nodes, and let \( N_h^\circ = N_h \cap \Omega \). We first consider a typical rectangle element \( T \in T_h \) assuming that the vertices of \( T \) are \( A_i, i = 1, 2, 3, 4 \), with \( A_i = (x_i, y_i)^T \). For a nontrivial interface \( \Gamma \), some of the elements in a mesh will be cut through by \( \Gamma \) and we will call them interface elements. The meaning of non-interface elements is obvious. If \( T \) is an interface element, then we use \( D = (x_n, y_n)^T \) and \( E = (x_e, y_e)^T \) to denote the intersection points of \( \Gamma \) and the edge of \( T \). In general, there are two types of rectangular interface elements. Type I are those for which the interface intersects two of its adjacent edges; Type II are those for which the interface intersects two of its opposite edges, see Fig. 3.

The basic idea of IFE method is to use standard finite element functions in non-interface elements, and use special finite element functions in interface elements that are...
constructed according to jump conditions across the interface. Hence, our main concern is the finite element functions in a typical interface element $T \in \mathcal{T}_h$. Note that the interface $\Gamma$ separates an interface element $T$ into two subsets $T^s = T \cap \Omega^s$, $s = \pm$. This suggests us to form a piecewise function with two bilinear polynomials on $T$ patched up together by interface jump conditions as follows:

$$\phi(x,y) = \begin{cases} 
\phi^-(X) = \phi^-(x,y) = a^-x + b^-y + c^- + d^-xy, & X = (x,y) \in T^-, \\
\phi^+(X) = \phi^+(x,y) = a^+x + b^+y + c^+ + d^+xy, & X = (x,y) \in T^+,
\end{cases}$$

$$\int_{DE} \left( \beta^- \frac{\partial \phi^-}{\partial n_{DE}} - \beta^+ \frac{\partial \phi^+}{\partial n_{DE}} \right) ds = 0,$$

(2.1)

where $n_{DE}$ is the unit vector perpendicular to the line $DE$. Further, we let $\phi_i(X)$ be the piecewise linear function described by (2.1) such that

$$\phi_i(A_j) = \phi_i(x_j,y_j) = \begin{cases} 
1, & \text{if } i = j, \\
0, & \text{if } i \neq j
\end{cases}$$

(2.2)

for $1 \leq i, j \leq 4$. We then introduce the local bilinear IFE space on element $T \in \mathcal{T}_h$:

$$S_h(T) = \text{span}\{\phi_i, i = 1, 2, 3, 4\},$$

where $\phi_i$, $i = 1, 2, 3, 4$ are the usual bilinear nodal basis functions if $T$ is a non-interface element; otherwise, $\phi_i$, $i = 1, 2, 3, 4$ are the piecewise bilinear polynomials defined by (2.1) and (2.2). Then, for each node $X_i \in \mathcal{N}_h$, we define $\Phi_i(X) = \Phi_i(x,y)$ to be a piecewise bilinear function such that $\Phi_i|_T \in S_h(T)$, $\Phi_i(X_j) = \delta_{ij}$, $\forall T \in \mathcal{T}_h$. Finally, we define the bilinear IFE space on $\Omega$ by $S_h(\Omega) = \text{span}\{\Phi_i(X) \mid X_i \in \mathcal{N}_h\}$.

The word “immersed” is used for this finite element space just to emphasize the fact that the mesh can be independent of the interface such that the interface can be immersed inside elements of this mesh. Fig. 4 illustrates the difference between the bilinear IFE local nodal basis function and the standard bilinear local nodal basis functions. Fig. 5 provides a sketch of the surface of a global bilinear IFE basis function over its support.
Figure 4: The plot on the left is for one of the bilinear IFE local nodal basis functions, the plot on the right is the corresponding regular bilinear local nodal basis function on the same element.

Figure 5: The plot on the left is the surface of one global bilinear IFE basis over its support, the plot on the right shows the elements forming the support and the interface.

We now list some basic properties of $S_h(\Omega)$, and we refer readers to [39] for details. From now on, for a given function $f(X)$, we let $f^s = f|_{\Omega^s}$, $s = \pm$.

- For a given rectangular mesh, the IFE space $S_h(\Omega)$ has the same number of nodal basis functions as that in the usual bilinear FE space.
- For a rectangular mesh $T_h$ fine enough, most of its elements are non-interface elements, and most of the nodal basis functions of the IFE space $S_h(\Omega)$ are just the usual bilinear nodal basis functions except for few nodes in the vicinity of the interface $\Gamma$.
- For any $\phi \in S_h(\Omega)$, we have
  \[ \phi|_{\Omega \setminus \Omega'} \in H^1(\Omega \setminus \Omega'), \]
  where $\Omega'$ is the union of interface rectangles.
- If $\Gamma \cap T$ is a line segment, then
  \[ \phi^-|_{\Gamma \cap T} = \phi^+|_{\Gamma \cap T}, \quad \forall \phi \in S_h(\Omega). \]
- Every function $\phi \in S_h(T)$ satisfies the flux jump condition on $\Gamma \cap T$ exactly in a weak sense as follows:
  \[ \int_{\Gamma \cap T} (\beta^- \nabla \phi^- - \beta^+ \nabla \phi^+) \cdot \mathbf{n} \, ds = 0. \]
• The bilinear IFE local nodal basis functions on an interface element $T$ satisfy the partition of unity, i.e.,
\[
\phi_1(X) + \phi_2(X) + \phi_3(X) + \phi_4(X) = 1, \quad \forall X \in T.
\]

• The bilinear IFE space is consistent with the usual bilinear finite element space in the sense that when $\beta^- = \beta^+$, we have
\[
\phi^- = \phi^+
\]
and $\phi$ become a usual bilinear polynomial for any $\phi \in S_h(T), \forall T \in T_h$.

• The bilinear IFE space $S_h(\Omega)$ has the usual approximation capability expected:
\[
\| I_h u - u \|_{0,T} + h \| I_h u - u \|_{1,T} \leq Ch^2
\]
provided that $u$ has the required regularity.

**Remark 2.1.** It is possible to extend the method in this article to handle variable discontinuous coefficient and non-homogeneous jump conditions. In order to deal with the variable discontinuous coefficients, we need to replace the constants $\beta^-$ and $\beta^+$ in (2.1) by the corresponding variable discontinuous coefficients. The idea of homogenization by using level-set method from [83] seems to be a viable approach to treat non-homogeneous jump conditions.

### 3 The immersed finite volume element method

Since the bilinear IFE space has the usual approximation capability expected from bilinear polynomials [39], we now apply it to solve the interface problem of the diffusion equation in the finite volume element formulation. To describe the method, for each mesh $T_h$ of $\Omega$, we introduce a dual mesh $\hat{T}_h$ by connecting the nearby centers of the elements in $T_h$ in the vertical and horizontal directions, see the illustration in Fig. 6 where the dual mesh $\hat{T}_h$ is sketched by the dash lines while $T_h$ is sketched by solid lines.

First, we derive a weak form on each element of the dual mesh. Assume that the source term $f(X)$ is smooth enough so that the exact solution has the required smoothness in the discussion below. Let $\hat{K}_i$ be an element of $\hat{T}_h$ containing the node $X_i$ of $T_h$. First, we integrate the differential equation (1.1) over $\hat{K}_i$ to have
\[
- \int_{\hat{K}_i} \nabla \cdot (\beta \nabla u) \; dx dy = \int_{\hat{K}_i} f \; dx dy. \tag{3.1}
\]
If $\hat{K}_i$ is not an interface element, then a straightforward application of the Green’s formula leads to
\[
- \int_{\partial \hat{K}_i} \beta \frac{\partial u}{\partial n} \; ds = \int_{\hat{K}_i} f \; dx dy. \tag{3.2}
\]
If \( \hat{K}_i \) is an interface element, then, by applying the Green’s formula piecewisely, we have
\[
-\int_{\hat{K}_i} \nabla \cdot (\beta \nabla u) \, dxdy - \int_{\partial \hat{K}_i} \beta \frac{\partial u}{\partial n} \, ds - \int_{\partial \hat{K}_i} \frac{\partial u}{\partial n} \, ds = \int_{\hat{K}_i} f \, dxdy,
\]
which leads to (3.2) again because of the flux jump condition (1.4). Hence, we conclude that the weak form (3.2) holds for any element \( \hat{K}_i \in \hat{T}_h \). This weak form enables us to introduce the bilinear immersed FVE method as follows: find \( u_h \in S_{h,E}(\Omega) \) such that
\[
-\int_{\partial \hat{K}_i} \beta \frac{\partial u_h}{\partial n} \, ds = \int_{\hat{K}_i} f \, dxdy, \quad \forall X_i \in N_h^\infty. \tag{3.3}
\]
Here, \( S_{h,E}(\Omega) = \{ v_h \in S_h(\Omega), v_h(X) = g(X) \forall X \in N_h \cap \partial \Omega \} \). We would like to point out that (3.3) indicates that the immersed FVE solution also have the local conservation property.

We now discuss some details in the implementation of the bilinear immersed FVE method. The key issue is the integrals used in this method. On each non-interface element \( \hat{K}_i \), standard Gaussian quadratures can be applied because we can assume that all the integrands involved are smooth enough. If \( \hat{K}_i \) is an interface element, both the line integral and the area integral in the bilinear immersed FVE method need to be treated carefully because of the discontinuity across the interface.

First, let us consider the area integral \( \int_{\hat{K}_i} f \, dxdy \) on the right hand side of (3.3). Under the assumption that \( f(X) \) is piecewise smooth with respect to the interface \( \Gamma \), we can approximate its integration over \( \hat{K}_i \) piecewisely by suitably partitioning \( \hat{K}_i \) into several sub-triangles. Assume that \( \hat{K}_i \) has vertices \( \hat{X}_j, j = 1,2,3,4 \) and interface \( \Gamma \) intersects with the boundary of \( \hat{K}_i \) at \( D \) and \( E \) on two adjacent edges, see Fig. 7. We can then use points \( D \) and \( E \) to partition \( \hat{K}_i \) into 4 triangles by adding 3 line segments: \( \overline{DE}, \overline{DX_3}, \overline{EX_3} \). Note
that the last two line segments are formed by connecting $D$ and $E$ to the vertex of $\hat{K}_i$ not on the edges containing $D$ and $E$. Hence,

$$\int_{\hat{K}_i} f \, dx dy = \int_{\triangle_{\hat{X}_1D}} f^- \, dx dy + \int_{\triangle_{\hat{X}_2\hat{X}_3}} f^+ \, dx dy + \int_{\triangle_{\hat{X}_3\hat{X}_4}} f^+ \, dx dy.$$ 

Gaussian quadratures with enough degree of precision can be applied straightforwardly to handle integrations on those sub-triangles within either $\Omega^-$ or $\Omega^+$. A little extra care is needed to handle the sub-triangles whose interiors intersect both $\Omega^-$ and $\Omega^+$. For the case illustrated in Fig. 7, when applying a Gaussian quadrature to compute $\int_{\triangle_{DE\hat{X}_3}} f^+ \, dx dy$,

we can replace the value of $f$ at a quadrature node outside $\Omega^+$ by the value of $f$ at a point on $\Gamma$ so long as this replacement has an $O(h^2)$ accuracy which can be achieved if $\Gamma$ is smooth enough within $\hat{K}_i$ [25]. A similar procedure can be developed for handling the case in which the interface $\Gamma$ intersect with the boundary of $\hat{K}_i$ at $D$ and $E$ on two opposite edges.

![Figure 7: A dual element $\hat{K} = \square \hat{X}_1 \hat{X}_2 \hat{X}_3 \hat{X}_4 \in \hat{T}_h$ sketched by dash lines and 4 adjacent elements of $T_h$. This element can be partitioned into 4 sub-triangles for the area integrals in the immerse FVE method.](image)

For an interface element $\hat{K}_i$, the line integral on the left hand side of (3.3) also needs to be treated piecewisely to handle the discontinuity. Again, let us consider a dual element $\hat{K}_i = \square \hat{X}_1 \hat{X}_2 \hat{X}_3 \hat{X}_4$, see Fig. 8. Since $\hat{K}_i$ has 4 edges, we have

$$-\int_{\partial \hat{K}_i} \beta \frac{\partial u_h}{\partial n} \, ds = -\int_{\hat{X}_1\hat{X}_2} \beta \frac{\partial u_h}{\partial n} \, ds - \int_{\hat{X}_2\hat{X}_3} \beta \frac{\partial u_h}{\partial n} \, ds$$

$$- \int_{\hat{X}_3\hat{X}_4} \beta \frac{\partial u_h}{\partial n} \, ds - \int_{\hat{X}_4\hat{X}_1} \beta \frac{\partial u_h}{\partial n} \, ds.$$ 

Note that the flux $\beta \frac{\partial u_h}{\partial n}$ on the boundary of $\hat{K}_i$ is discontinuous at the points where $\partial \hat{K}_i$ intersects either the edges of $T_h$ or the interface $\Gamma$. Therefore, the line integrals on the
Figure 8: A dual element $\tilde{K}_i = \square \tilde{X}_1 \tilde{X}_2 \tilde{X}_3 \tilde{X}_4 \in \mathcal{T}_h$ sketched by dash lines and 4 adjacent elements of $\mathcal{T}_h$. The edges of $\tilde{K}_i$ is partitioned by the discontinuous points of the flux for the line integrals in the immersed FVE method.

right hand side above need to be computed on the small line segments between these discontinuous points. For the example demonstrated in Fig. 8, we have

$$\int_{X_1 X_2} \beta \frac{\partial u_h}{\partial n} \, ds = \int_{X_1 A} \beta^- \frac{\partial u_h}{\partial n} \, ds + \int_{A X_2} \beta^+ \frac{\partial u_h}{\partial n} \, ds,$$

$$\int_{X_2 X_3} \beta \frac{\partial u_h}{\partial n} \, ds = \int_{X_2 B} \beta^+ \frac{\partial u_h}{\partial n} \, ds + \int_{B X_3} \beta^- \frac{\partial u_h}{\partial n} \, ds,$$

$$\int_{X_3 X_4} \beta \frac{\partial u_h}{\partial n} \, ds = \int_{X_3 C} \beta^+ \frac{\partial u_h}{\partial n} \, ds + \int_{C X_4} \beta^- \frac{\partial u_h}{\partial n} \, ds,$$

$$\int_{X_4 X_1} \beta \frac{\partial u_h}{\partial n} \, ds = \int_{X_4 D} \beta^+ \frac{\partial u_h}{\partial n} \, ds + \int_{D X_1} \beta^- \frac{\partial u_h}{\partial n} \, ds.$$

We note that all the integrands in the line integrals on the right hand sides above are polynomials; hence, a Gaussian quadrature with enough degree precision can be used to compute all of them precisely. As a consequence, this leads to another interesting fact that the matrix in the immersed FVE can be assembled exactly even if the interface $\Gamma$ is a general curve. On the contrary, the matrices in the immersed finite element methods [39, 48, 57–59] cannot be formed precisely unless the interface $\Gamma$ is trivial. In assembling the matrix in any of these immersed finite element methods over an interface element $K \in \mathcal{T}_h$, assuming that the interface $\Gamma$ intersects the edges of $K$ at $D$ and $E$, the error in the computation of the area integral over the region enclosed by $DE$ and $\Gamma$ is inevitable if $\Gamma$ is a general curve.

Finally, we would like to point out that, for any given rectangular mesh $\mathcal{T}_h$ of $\Omega$, the algebraic system of this bilinear immersed FVE method has the same structure as the algebraic system in the usual bilinear finite element method for the Dirichlet boundary value problem of the Poisson equation. The matrix in its algebraic system is guaranteed to be symmetric positive definite.
Table 1: Errors of the FV-IFE solution for the case with $\beta^- = 1$, $\beta^+ = 10$.

| $h$   | $||u_h - u||_0$ | $||u_h - u||_1$ | $||u_h - u||_{\infty}$ |
|-------|----------------|----------------|------------------------|
| 1/8   | $7.7394 \times 10^{-3}$ | $1.1705 \times 10^{-1}$ | $2.5110 \times 10^{-3}$ |
| 1/16  | $1.9658 \times 10^{-3}$ | $5.8644 \times 10^{-2}$ | $6.5026 \times 10^{-4}$ |
| 1/32  | $4.8172 \times 10^{-3}$ | $2.9255 \times 10^{-2}$ | $2.5110 \times 10^{-3}$ |
| 1/64  | $1.2173 \times 10^{-3}$ | $1.4550 \times 10^{-2}$ | $4.1413 \times 10^{-3}$ |
| 1/128 | $3.0115 \times 10^{-3}$ | $7.2699 \times 10^{-3}$ | $1.0611 \times 10^{-3}$ |
| 1/256 | $7.5436 \times 10^{-4}$ | $3.6362 \times 10^{-3}$ | $2.6485 \times 10^{-4}$ |

Table 2: Errors of the FV-IFE solution for the case with $\beta^- = 1$, $\beta^+ = 10000$.

| $h$   | $||u_h - u||_0$ | $||u_h - u||_1$ | $||u_h - u||_{\infty}$ |
|-------|----------------|----------------|------------------------|
| 1/8   | $1.8420 \times 10^{-3}$ | $4.1025 \times 10^{-2}$ | $1.4562 \times 10^{-3}$ |
| 1/16  | $4.0555 \times 10^{-3}$ | $2.1051 \times 10^{-2}$ | $4.2813 \times 10^{-4}$ |
| 1/32  | $7.6016 \times 10^{-5}$ | $1.0193 \times 10^{-2}$ | $2.5606 \times 10^{-4}$ |
| 1/64  | $2.4890 \times 10^{-6}$ | $7.2699 \times 10^{-3}$ | $5.0649 \times 10^{-5}$ |
| 1/128 | $5.1332 \times 10^{-6}$ | $2.4100 \times 10^{-3}$ | $1.8048 \times 10^{-5}$ |
| 1/256 | $1.1050 \times 10^{-6}$ | $1.2110 \times 10^{-3}$ | $4.7363 \times 10^{-6}$ |

4 Numerical examples

In this section, we present numerical examples for the bilinear immersed finite volume element method to illustrate its features. We consider the interface problem defined by (1.1)-(1.4) on the typical rectangular domain $\Omega = [-1,1] \times [-1,1]$. The interface curve $\Gamma$ is a circle with radius $r_0 = \pi / 6.28$ that separates $\Omega$ into two sub-domains $\Omega^-$ and $\Omega^+$ with $\Omega^- = \{(x,y) \mid x^2 + y^2 \leq r_0^2\}$. The coefficient function is

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

The boundary condition function $g(x,y)$ and the source term $f(x,y)$ are chosen such that the following function $u$ is the exact solution.

$$
u(x,y) = \begin{cases} 
r^\alpha/\beta^-, & \text{if } r \leq r_0, \\
r^\alpha/\beta^+ + \left(1/\beta^- - 1/\beta^+\right)r_0^\alpha, & \text{otherwise},
\end{cases}
$$

(4.1)

with $\alpha = 5$, $r = \sqrt{x^2 + y^2}$. For simplicity, we only use the simple rectangular Cartesian meshes in our numerical experiments.

Table 1 contains the errors of the bilinear immersed FVE solution $u_h$ with various mesh size $h$ and $\beta^- = 1$, $\beta^+ = 10$. Table 2 contains the errors of the bilinear immersed FVE solution $u_h$ with $\beta^- = 1$, $\beta^+ = 10000$ representing a large jump. Table 3 contains the errors
Table 3: Errors of the FV-IFE solution for the case with $\beta^- = 10$, $\beta^+ = 1$.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$|u_h - u|_0$</th>
<th>$|u_h - u|_1$</th>
<th>$|u_h - u|_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/8</td>
<td>$7.6119 \times 10^{-2}$</td>
<td>$1.0927 \times 10^0$</td>
<td>$2.6593 \times 10^{-2}$</td>
</tr>
<tr>
<td>1/16</td>
<td>$1.9110 \times 10^{-2}$</td>
<td>$5.4809 \times 10^{-1}$</td>
<td>$6.6274 \times 10^{-3}$</td>
</tr>
<tr>
<td>1/32</td>
<td>$4.7894 \times 10^{-3}$</td>
<td>$2.7425 \times 10^{-1}$</td>
<td>$1.6796 \times 10^{-3}$</td>
</tr>
<tr>
<td>1/64</td>
<td>$1.1967 \times 10^{-3}$</td>
<td>$1.3715 \times 10^{-1}$</td>
<td>$4.1590 \times 10^{-4}$</td>
</tr>
<tr>
<td>1/128</td>
<td>$2.9946 \times 10^{-4}$</td>
<td>$6.8576 \times 10^{-2}$</td>
<td>$1.0489 \times 10^{-4}$</td>
</tr>
<tr>
<td>1/256</td>
<td>$7.4846 \times 10^{-5}$</td>
<td>$3.4288 \times 10^{-2}$</td>
<td>$2.6144 \times 10^{-5}$</td>
</tr>
</tbody>
</table>

Table 4: Errors of the FV-IFE solution for the case with $\beta^- = 10000$, $\beta^+ = 1$.

<table>
<thead>
<tr>
<th>$h$</th>
<th>$|u_h - u|_0$</th>
<th>$|u_h - u|_1$</th>
<th>$|u_h - u|_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/8</td>
<td>$7.6026 \times 10^{-2}$</td>
<td>$1.0927 \times 10^0$</td>
<td>$2.6270 \times 10^{-2}$</td>
</tr>
<tr>
<td>1/16</td>
<td>$1.9119 \times 10^{-2}$</td>
<td>$5.4813 \times 10^{-1}$</td>
<td>$6.7172 \times 10^{-3}$</td>
</tr>
<tr>
<td>1/32</td>
<td>$4.7613 \times 10^{-3}$</td>
<td>$2.7425 \times 10^{-1}$</td>
<td>$1.6608 \times 10^{-3}$</td>
</tr>
<tr>
<td>1/64</td>
<td>$1.1930 \times 10^{-3}$</td>
<td>$1.3714 \times 10^{-1}$</td>
<td>$4.0496 \times 10^{-4}$</td>
</tr>
<tr>
<td>1/128</td>
<td>$2.9813 \times 10^{-4}$</td>
<td>$6.8575 \times 10^{-2}$</td>
<td>$1.0490 \times 10^{-4}$</td>
</tr>
<tr>
<td>1/256</td>
<td>$7.4494 \times 10^{-5}$</td>
<td>$3.4288 \times 10^{-2}$</td>
<td>$2.6902 \times 10^{-5}$</td>
</tr>
</tbody>
</table>

of the bilinear immersed FVE solution $u_h$ with various mesh size $h$ and $\beta^- = 10$, $\beta^+ = 1$. Table 4 contains the errors of the bilinear immersed FVE solution $u_h$ with $\beta^- = 10000$, $\beta^+ = 1$. In these tables, $\|\cdot\|_0$ represents the usual $L^2$ norm, $\|\cdot\|_1$ is the usual semi-$H^1$ norm, and of course, they are computed numerically according to the mesh used. The quantity $\|\cdot\|_\infty$ is the discrete infinity norm which is the maximum of the absolute values of the given function at all the nodes of a mesh.

We can easily see that the data in the second and third columns of these tables satisfy

$$
\|u_h - u\|_0 \approx \frac{1}{4} \|u_h - u\|_0, \quad |u_h - u|_1 \approx \frac{1}{2} |u_h - u|_1,
$$

for $h = \hat{h}/2$. Using linear regression, we can see that the data in Table 1 obey

$$
\|u_h - u\|_0 \approx 0.5008h^{2.0024}, \quad |u_h - u|_1 \approx 0.9427 h^{1.0025}, \quad \|u_h - u\|_\infty \approx 0.1559 h^{1.9788},
$$

the data in Table 2 obey

$$
\|u_h - u\|_0 \approx 0.1422 h^{2.1154}, \quad |u_h - u|_1 \approx 0.3514 h^{1.0246}, \quad \|u_h - u\|_\infty \approx 0.0486 h^{1.6390},
$$

the data in Table 3 obey

$$
\|u_h - u\|_0 \approx 4.8643 h^{1.9983}, \quad |u_h - u|_1 \approx 8.7375 h^{0.9990}, \quad \|u_h - u\|_\infty \approx 1.6923 h^{1.9974},
$$

and the data in Table 4 obey

$$
\|u_h - u\|_0 \approx 4.8715 h^{1.9995}, \quad |u_h - u|_1 \approx 8.7379 h^{0.9990}, \quad \|u_h - u\|_\infty \approx 1.6301 h^{1.9861}.
$$
Figure 9: The plot on the left is for the linear regression of the data in Table 1 and the plot on the right is for the linear regression of the data in Table 2.

Figure 10: The plot on the left is for the linear regression of the data in Table 3 and the plot on the right is for the linear regression of the data in Table 4.

See Figs. 9 and 10 for these linear regressions. These results further indicates that the bilinear immersed IVE solution $u_h$ converges to the exact solution with convergence rates $O(h^2)$ and $O(h)$ in the $L^2$ norm and $H^1$ norm, respectively. However, the actual computational results show that the solution does not always have the second order convergence in the $L^\infty$ norm even though the mesh is fine enough. Similar phenomenon has been observed for IFE method, see [39, 57]. We guess this is mainly due to the non-conforming feature of the IFE space, and we plan to investigate this issue in our future research.

For a given rectangular mesh of $\Omega$, we note that the linear system in this bilinear immerse IVE method has the same structure as that in the IVE method based on the standard bilinear finite elements for the Poisson’s equation, especially from the point view of the number of non-zero entries and their locations in the matrix of the related linear system. This suggests that, on any given computer, the CPU time needed to solve the
Table 5: Comparison of the computational costs for solving linear systems in both the bilinear FVE method and the bilinear immersed FVE method.

<table>
<thead>
<tr>
<th></th>
<th>bilinear FVE</th>
<th>$\beta^+ : \beta^- = 1:1.1$</th>
<th>$\beta^+ : \beta^- = 1:2$</th>
<th>$\beta^+ : \beta^- = 1:10$</th>
</tr>
</thead>
<tbody>
<tr>
<td># of iterations</td>
<td>221</td>
<td>222</td>
<td>279</td>
<td>299</td>
</tr>
</tbody>
</table>

bilinear immersed FVE method should be comparable to that needed to solve the linear system in the standard bilinear FVE for simple Poisson’s equation. Since it has become more and more difficult to obtain the precise CPU time usage of a computational procedure on a modern computer because of the complexity of the CPU unit (multi cores, cache, hardware parallelization, etc.) and the software (operating system, fire-wall, virus scan, etc.), we choose the number of iterations needed to make the preconditioned conjugate gradient (PCG) method to converge for a given error tolerance to illustrate the above observation, see Table 5. For the $\Omega$ specified at the beginning of this section, we use a rectangular mesh with $h = \frac{1}{128}$, the incomplete Cholesky preconditioner, and the error tolerance $tol = 10^{-10}$ in all the computations. From this table, we can see that, while the linear system in the bilinear FVE method for the Poisson’s equations uses 221 PCG iterations, the linear system in the bilinear immersed FVE method uses a 222 PCG iteration for the interface problem described in this section with $\beta^+ : \beta^- = 1:1.1$. We have also observed that the number of PCG iterations needed by the bilinear immersed FVE method gradually increases as the ratio $\beta^+ : \beta^-$ becomes larger. This increase is due to the fact that the interface problem is essentially more difficult than the simple boundary value problem of the Poisson’s equation and will inevitably cost more time to solve by any method.

5 Conclusion

In this paper, we have presented an bilinear immersed finite volume element method for solving the interface problem of the diffusion equation whose domain is formed with multiple materials. This method possesses both the advantages of local conservation in a FVE method and the capability of IFE for handling the jump conditions across material interface. This method can use a Cartesian mesh even if the interface separating the materials is nontrivial, and fast algebraic solvers such as multigrid can be easily applied to generate numerical solutions efficiently for a problem with rather complicated interface. The numerical examples show that this method does have an approximation capability usually expected from bilinear polynomials.

Acknowledgments

This work is partially supported by NSF grant DMS-0713763 and NSERC (Canada). We also appreciate the referee’s suggestions and comments which have improved the paper.
References


