Comput. Methods Appl. Mech. Engrg. 200 (2011) 494–508

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

Discontinuous bubble scheme for elliptic problems with jumps in the solution

Kwang S. Chang, D.Y. Kwak *,[1]

Department of Mathematical Sciences, Korea Advanced Institute of Science and Technology, Daejeon 305-701, Republic of Korea

ARTICLE INFO

Article history:
Received 13 July 2009
Received in revised form 30 March 2010
Accepted 24 June 2010
Available online 30 July 2010

Keywords:
Jumps in the solution
Discontinuous bubble
Immersed interface
Immersed finite element method
Uniform grid
Robust convergence

ABSTRACT

We propose a new numerical method to solve an elliptic problem with jumps both in the solution and derivative along an interface. By considering a suitable function which has the same jumps as the solution, we transform the problem into one without jumps. Then we apply the immersed finite element method in which we allow uniform meshes so that the interface may cut through elements to discretize the problem as introduced in [1–3]. Some convenient way of approximating the jumps of the solution by piecewise linear functions is suggested. Our method can also handle the case when the interface passes through grid points. We believe this paper presents the first resolution of such cases. Numerical experiments for various problems show second-order convergence in L^2 and first order in H^1-norms. Moreover, the convergence order is very robust for all problems tested.

1. Introduction

In recent years, there have been extensive efforts to find numerical methods for solving problems with interface. There are many physical problems where the underlying partial differential equations have an interface. For example, second-order elliptic equations with discontinuous coefficients are often used to model problems in material sciences and porous media when two or more distinct materials or media with different conductivities, densities or permeability are involved. The solution of these interface problems must satisfy certain interface jump conditions due to conservation laws. Interface problems with jump conditions also arise in many other applications. For example, they are used to describe the electroporation state of a biological cell under an electric field [4,5]. Other examples of a problem with interface are time-dependent parabolic equations having a moving interface, the incompressible Navier–Stokes equations which describe two phase fluids, and the motion of elastic body embedded in some fluids.

The solution of the interface problem is smooth only on the individual regions where the coefficients are smooth, but due to the jumps of the coefficients across the interface, the global regularity is usually low and the solution usually belongs to $H^1(\Omega)$ if the jumps are homogeneous; however, if the jumps are inhomogeneous, then the solution or derivative is often discontinuous, hence the corresponding numerical method is more difficult to design. Because of the discontinuity of solution, achieving accuracy is very difficult with standard finite element methods, unless the elements fit with the interface of general shape.

For these problems, many numerical methods have been suggested. An early attempt to study the interface problem goes back, at least, to 1977 where Peskin [6] used the Dirac δ function in describing the motion of a heart embedded in the human body fluid. Since then there have been many attempts to solve the problem with immersed interface. For example, the Immersed Boundary Method (IBM) was introduced in [6–9]. IBM uses a finite difference method in conjunction with a regularized δ-function. However this method smears out the solution near the interface and shows first order accuracy only.

To resolve the low order accuracy, some new types of methods have been proposed based on the immersed boundary idea of Peskin. In [10], LeVeque and Li proposed the "Immersed Interface Method (IIM)" where they derived jump conditions in the solution from the effect of δ-function and used finite difference to incorporate them into the finite difference scheme, thus avoiding large errors arising from discretizing the δ-function directly. This seems one of the early results to have second-order accuracy with Cartesian grid. Also, the numerical solution does not smear near the interface.

One of the advantages of IIM is the use of uniform Cartesian grids. Ideas similar to IIM were subsequently applied to other interface problems such as Hele–Shaw, Stokes flow, Navier–Stokes flow problem, etc., see [11–15] and references therein. Although these methods were demonstrated to be very effective, there is room to improve the schemes. For example, in most of IIM schemes, the coefficients for the numerical stencil are derived with

* Corresponding author. Tel.: +82 42 869 2720; fax: +82 42 869 2710.
E-mail addresses: ckslove@kaist.ac.kr (K.S. Chang), kdy@kaist.ac.kr (D.Y. Kwak).
[1] The work of this author was supported by KOSEF(contract number R01-2007-000-10062-0), Korea.

0045-7825/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2010.06.029

a minimization method to preserve the M-matrix property; thus in general the resulting system is nonsymmetric even though the original system is symmetric and positive definite and the stencil is complicated to derive. This makes it impossible to use easy and fast solvers such as conjugate gradient, ADI, etc. Other disadvantage is that the IIM is derived from Taylor series on each side the region which requires higher order regularity of the solution while most solutions of an elliptic problem only have H^2 regularity in each subdomain. Thus, for problems having rough data or corners, IIM may not work well. Other related works in this direction can be found in [16–22] and references therein.

Above listed methods are all finite difference based methods. While, for finite element method sides, Li, Lin, Rogers and Ryan designed an Immersed Finite Element Method (IFEM) for rectangular grid in [3], Lin, Lin, Rogers and Wu considered IFEM for triangular grid and proved approximation properties in [1,2]. In these methods the interface is allowed to cut through the interior of the element, while the basis function is designed to satisfy the continuity and jump of the flux along the interface. Their numerical examples demonstrated optimal orders of the error.

A rigorous convergence analysis for this method for elliptic case was provided in a recent paper by the authors in [24,25], where they have shown that the convergence order is $O(h)$ in H^1 and $O(h^2)$ in L^2 norm independently of the location of interface.

IFEM is quite convenient in several aspects. First of all, the stencil is a five point stencil, while that of IIM is six points, and the coefficients are computed element-by-element just like the standard finite element methods. Hence the structure of stiffness matrix is exactly the same as that of the standard finite element method. Hence, the method can be easily incorporated into existing software packages. This is in contrast to the IIM of Li et al., where they need to solve a linear system involving 9 unknowns for each grid point. Secondly, IFEM results in a symmetric positive definite matrix when the underlying problem is symmetric and uniformly elliptic. Hence many known efficient solvers can be exploited. Third, when moving interface problems such as Hele–Shaw problem, which involves two phase flows with different viscosity and variable surface tension, one does not need to generate a new grid as time evolves. This saves considerable amount of time and storage. So far the IFEM was only developed for problems with homogeneous jumps.

A finite element approach for nonhomogeous jump is considered in [18] where they used triangular grid with built-in jump conditions in the basis function. In their approach, the trial finite function belongs to an affine space which is not a vector space. Hence their approach is different from the usual finite element method. Furthermore, the resulting linear system is nonsymmetric which is expensive to solve. Also, they cannot deal with the case when the interface passes through grid points. So they perturb the interface by a small amount, adjusting the level set function to avoid such a case.

Some other methods related to this problems are XFEM type which were developed for fluid–structure problems, see [26,27], where they enrich standard finite element space by augmenting extra functions near the interface to handle the jump. Usually they require extra degrees of freedom and mesh refinement.

In this paper, we propose a fast accurate algorithm for problems with nonhomogeneous jump conditions. The algorithm is based on the finite element method based algorithm which can use any regular triangular finite element meshes including uniform meshes. The idea is to consider certain a singular function in a neighborhood of the interface whose jumps match the given jump conditions. By subtracting this function, we obtain a problem in which the solution has homogeneous jumps. Then we can use any reasonable numerical method to solve the resulting equation. We specifically suggest a method of constructing piecewise linear interpolation near the interface whose jumps match the given jump conditions. Then we use the IFEM to solve the resulting equation. It obtains a sharp solution near the interface contrary to approaches based on discrete Dirac source terms. Thus, our new method (we call it *Discontinuous Bubble Immersed Finite Element Method*-DBIFEM) provides a natural framework to incorporate the jumps of solutions in the immersed finite element formulation. The resulting discretization leads to a symmetric system that can be efficiently solved with standard algorithms such as conjugate gradient method, ADI method or multigrid method. Furthermore, our DBIFEM can handle the case when the interface passes through a vertex or many vertices (Examples 6, 7). To the authors' knowledge, this is the first algorithm that can handle these cases. See Section 3.1 for details. Numerical experiments show our method gives optimal order convergence for various examples.

Some features of our method are:

- We can use any shape regular grid including the uniform grid;
- We do not add any extra degree of freedom near the interface;
- We subtract a piecewise linear discrete singular function with a thin support which satisfies the given jump conditions, so the cost is cheap;
- Our method shows optimal convergence order in L^2 and in H^1 for all of the problems tested;
- The resulting system is symmetric and positive definite when the original equation is symmetric and uniformly elliptic;
- Our stencil is independent of the jump condition, while some IIM based methods have jump conditions built into the stencil. Thus the condition number of the matrix is independent of jumps, which is not the case for most other methods (see the comment in [20, p. 231]);
- Our method allows the interface to pass one or more grid points exactly, while all the other method known so far cannot handle these cases; they usually avoid such cases by perturbing the interface slightly, see [18, p. 418].

2. Preliminaries

2.1. Mathematical formulation

Let Ω be a convex domain with Lipschitz continuous boundary in $\mathbb{R}^2$, which is separated into two disjoint subdomains $\Omega^+$and Ω^- by a interface Γ as in Fig. 1. Let $\mathbf{n}$ be the unit outward normal vector to Γ. We consider the following elliptic interface problem

$$-\nabla\cdot(\beta\nabla u) = f \quad \text{in } \Omega^{\pm}, \quad u = g \quad \text{on } \partial\Omega, \tag{1}$$

together with the jump conditions on the interface:

$$[u]_\Gamma(\mathbf{x}) \equiv u^-(\mathbf{x}) - u^+(\mathbf{x}) = J_1(\mathbf{x}) \quad \text{on } \Gamma, \tag{2a}$$

$$\left[\beta\frac{\partial u}{\partial \mathbf{n}}\right]_\Gamma(\mathbf{x}) \equiv \beta^-\frac{\partial u^-}{\partial \mathbf{n}} - \beta^+\frac{\partial u^+}{\partial \mathbf{n}} = J_2(\mathbf{x}) \quad \text{on } \Gamma, \tag{2b}$$

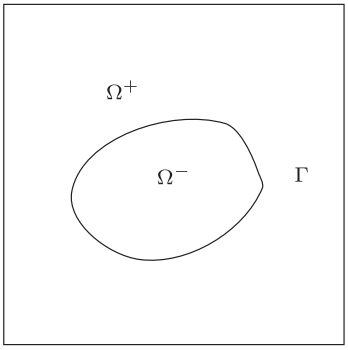


Fig. 1. Sketch of the domain Ω for the interface problem.

where $f \in L^2(\Omega)$, $g \in H^{3/2}(\partial\Omega)$, $J_1 \in H^{3/2}(\Gamma)$, and $J_2 \in H^{1/2}(\Gamma)$. For simplicity we may assume the domain is the rectangle $\Omega = [-1,1] \times [-1,1]$, but our presentation works for general domain with Lipschitz boundary. We assume $\beta(\mathbf{x})$ is a positive function greater than a constant $\beta_0 > 0$ having discontinuity along Γ, i.e.,

$$\beta(\mathbf{x}) = \beta^-(\mathbf{x}) \quad \text{for } \mathbf{x} \in \Omega^-; \beta(\mathbf{x}) = \beta^+(\mathbf{x}) \quad \text{for } \mathbf{x} \in \Omega^+. \tag{3}$$

Here, superscripts "+, −" refer to limits taken from within the subdomains $\Omega^{\pm}$.

We introduce some notations of function spaces: For any subdomain D, we write $D^+ = D \cap \Omega^+$, $D^- = D \cap \Omega^-$ and let $H^m(D), H_0^1(D)$ and $H^m(\partial D)$, etc., be ordinary Sobolev spaces on D and ∂D. Define the following spaces:

$$H_P^m(D) := H^m(D^+) \cap H^m(D^-)$$

with piecewise norms

$$\|u\|^2_{H_P^m(D)} = \|u\|^2_{H^m(D^+)} + \|u\|^2_{H^m(D^-)}. \tag{4}$$

Finally, we define some subsets of functions in $H_P^2(D)$ which have prescribed jump conditions: For any $\gamma_1, \gamma_2 \in L^2(D \cap \Gamma)$,

$$\mathcal{U}^{\gamma_1,\gamma_2}(D) := \left\{ u \in H_P^2(D) | [u] = \gamma_1, \left[\beta \tfrac{\partial u}{\partial \mathbf{n}}\right] = \gamma_2 \quad \text{on } D \cap \Gamma \right\},$$
$$\mathcal{U}_0^{\gamma_1,\gamma_2}(D) := \{ u \in \mathcal{U}^{\gamma_1,\gamma_2}(D) | u = 0 \quad \text{on } \partial D \}.$$

If the pair (γ_1, γ_2) is equal to (J_1, J_2), then we omit the notation (J_1, J_2), i.e., we shall write $\mathcal{U}(D), \mathcal{U}_0(D)$ for $\mathcal{U}^{J_1,J_2}(D), \mathcal{U}_0^{J_1,J_2}(D)$ respectively. We cite without proof the following result obtained in [28] that asserts that the problem (1) is well-posed:

Theorem 1. *Assume that $f \in L^2(\Omega)$. Then the problem (1) has a unique solution $u \in \mathcal{U}(\Omega)$ such that for some constant $C > 0$*

$$\|u\|_{H_P^2(\Omega)} \leqslant C\left(\|f\|_{L^2(\Omega)} + \|g\|_{H^{3/2}(\partial\Omega)} + \|J_1\|_{H^{3/2}(\Gamma)} + \|J_2\|_{H^{1/2}(\Gamma)} \right). \tag{5}$$

2.2. Weakly enforced jump conditions

Without loss of generality, we assume $g \equiv 0$ from now on. To find a weak formulation for the interface problem, we multiply (1) by v and integrate on each subdomain:

$$-\int_{\partial\Omega^{\pm}} \beta \frac{\partial u}{\partial \mathbf{n}} v \, ds + \int_{\Omega^{\pm}} \beta \nabla u \cdot \nabla v \, d\mathbf{x} = \int_{\Omega^{\pm}} f v \, dx, \quad \forall v \in H_0^1(\Omega). \tag{6}$$

By summation, we get:

$$\int_{\Omega^+} \beta \nabla u \cdot \nabla v \, d\mathbf{x} + \int_{\Omega^-} \beta \nabla u \cdot \nabla v \, dx = \int_{\Gamma} \left[\beta \frac{\partial u}{\partial \mathbf{n}}\right] v \, ds + \int_{\Omega} f v \, dx, \quad \forall v \in H_0^1(\Omega). \tag{7}$$

Define

$$a(u,v) = \int_{\Omega^+} \beta \nabla u \cdot \nabla v \, d\mathbf{x} + \int_{\Omega^-} \beta \nabla u \cdot \nabla v \, dx, \quad \forall u, v \in H^1(\Omega^+) \cap H^1(\Omega^-).$$

Then the problem (1), (2) becomes: find $u \in H_P^2(\Omega)$ satisfying the jump condition (2a) and

$$a(u,v) = < J_2, v >_{\Gamma} + (f, v), \forall v \in H_0^1(\Omega). \tag{8}$$

Here $(\cdot,\cdot)$ denotes the $L^2(\Omega)$ inner product and $< \cdot, \cdot >_{\Gamma}$ denotes the $L^2(\Gamma)$ inner product.

We shall show this is equivalent to the original problem (1), (2). Suppose that Eq. (8) holds. Then, for any $v \in H_0^1(\Omega^{\pm})$, we have:

$$\int_{\Omega^+} \beta \nabla u \cdot \nabla v \, d\mathbf{x} + \int_{\Omega^-} \beta \nabla u \cdot \nabla v \, d\mathbf{x} = \int_{\Omega} f v dx. \tag{9}$$

So, we obtain the following equations (holding on each subdomain Ω^+ and Ω^- respectively).

$$-\nabla \cdot (\beta \nabla u) = f \quad \text{a.e. on } \Omega^{\pm}. \tag{10}$$

Now let $v \in H_0^1(\Omega)$. Integrating by parts, we see the left hand side of (8) becomes:

$$\begin{aligned} &\sum\nolimits_{s=\pm} \left(\int_{\partial\Omega^s} \beta \frac{\partial u}{\partial \mathbf{n}} v \, ds - \int_{\Omega^s} \nabla \cdot (\beta \nabla u) v \, d\mathbf{x} \right) \\ &= \int_{\Gamma} \left[\beta \frac{\partial u}{\partial \mathbf{n}}\right] v \, ds - \sum_{s=\pm} \int_{\Omega^s} \nabla \cdot (\beta \nabla u) v \, d\mathbf{x} \\ &= \int_{\Gamma} \left[\beta \frac{\partial u}{\partial \mathbf{n}}\right] v \, ds + \int_{\Omega} f v \, d\mathbf{x} \quad (\text{by}(10)) \end{aligned} \tag{11}$$

Comparing this with the right hand side (8), we obtain:

$$\left[\beta \frac{\partial u}{\partial \mathbf{n}}\right] = J_2.$$

Thus, we have shown that (8) is equivalent to the original problem (1), (2).

Since the solution u belongs to an affine subset $H_P^2(\Omega)$, which is not a subspace, we would like to change it to a problem in $H_0^1(\Omega)$. For this purpose, we subtract a function satisfying nonhomogeneous jump conditions: Let u^* be any function in $H_P^2(\Omega)$ satisfying the jump condition and having zero boundary condition on $\partial\Omega$. Then we have the following decomposition for u:

$$u = u^0 + u^*, \tag{12}$$

where $u^0 \in H_0^1(\Omega)$. Then the problem (1) becomes:

$$-\nabla \cdot (\beta \nabla u^0) = f + \nabla \cdot (\beta \nabla u^*) \quad \text{in } \Omega^+, \tag{13a}$$
$$-\nabla \cdot (\beta \nabla u^0) = f + \nabla \cdot (\beta \nabla u^*) \quad \text{in } \Omega^-, \tag{13b}$$
$$u = 0 \quad \text{on } \partial\Omega.$$

Hence the variational form for the new problem becomes: find $u^0 \in H_0^1(\Omega)$ satisfying

$$a(u^0, v) = < J_2, v >_{\Gamma} + (f, v) - a(u^*, v), \quad \forall v \in H_0^1(\Omega). \tag{14}$$

Then, we have the following regularity theorem for the weak solution u of the variational problem (8).

Theorem 2. *Assume that $f \in L^2(\Omega)$. Then the variational problem (8) has a unique solution $u \in \mathcal{U}_0(\Omega)$ which satisfies*

$$\|u\|_{H_P^2(\Omega)} \leqslant C \inf(\|f\|_0 + \|u^*\|_{H_P^2(\Omega)}), \tag{15}$$

where infimum is taken over all $u^* \in \mathcal{U}_0(\Omega)$.

Proof. We have by (12), (13a) and (13b),

$$\begin{aligned} \|u\|_{H_P^2(\Omega)} &\leqslant \|u^0\|_{\widetilde{H^2(\Omega)}} + \|u^*\|_{H_P^2(\Omega)} \\ &\leqslant C \sum_{s=\pm} \|f + \nabla \beta \nabla u^*\|_{L^2(\Omega^s)} + \|u^*\|_{H_P^2(\Omega)}. \end{aligned}$$

Now the result follows by taking infimum for $u^* \in \mathcal{U}_0(\Omega)$. □

Remark 1

1. Since the choice of u^* is not unique, we have to find one such u^* in numerical computation. The idea is to choose u^* (or its approximation) whose norm is as small as possible. As a simple choice, a function u^* with small support seems to be easiest for implementation and good enough as our numerical experiments show. Another choice is to let u^* be the solution of a biharmonic problem:

$$-\Delta^2 u^* = 0 \quad \text{in } \Omega^-$$
$$[u^*] = J_1, \quad \left[\beta \tfrac{\partial u^*}{\partial \mathbf{n}}\right] = J_2 \quad \text{on } \Gamma \equiv \partial\Omega^-$$

and compute a finite element approximation to u^*. This would be a robust method. However, solving a biharmonic problem is a heavy task, especially on irregular grids. As our numerical experiments show, our scheme works well even for the nonconstant coefficient case.

2. If the jumps are homogenous, i.e., $J_1 = J_2 = 0$, then $u^0 := u$ belongs to $H_0^1(\Omega)$ so that we have:

$$a(u^0, v) = (f, v), \quad \forall v \in H_0^1(\Omega). \tag{16}$$

The numerical methods for problems with homogeneous jump conditions have been considered by many authors in finite difference approach and a few authors in finite element approach; In particular, the immersed finite element method considered in [1–3] can solve this problem very efficiently. The error estimate of such method is recently shown in [25].

In the next section, we shall describe our DBIFEM using any shape regular grid independent of the interface for solving problems with nonhomogeneous jump. When jump becomes homogeneous, our method reduces to IFEM in [2].

3. Numerical method

To discretize the problem, we define the following finite-dimensional spaces: let

$$\mathcal{B}_h(D) := \{u | u \text{ is piecewise linear on } T^{\pm}, \text{ is continuous at vertices of } T \text{ satisfying jump conditions (2a), (2b) for } T \subset D, \ \forall T \in \mathcal{T}_h\},$$

$$\mathcal{B}_{h,0}(D) := \{u \in \mathcal{B}_h(D) |\ u = 0 \quad \text{on } \partial D\}$$

with the piecewise norm

$$\|u\|_{m,h}^2(D) := \sum_{\substack{T \subset D \\ T \in \mathcal{T}_h}} \|u\|_{H_p^m(T)}^2, \quad m = 0, 1. \tag{17}$$

Note that a function in the space $\mathcal{B}_h(D)$ is allowed to be discontinuous across edges between elements.

Assuming we have a u^* and its approximation u_h^*, we consider the decomposition of the discrete solution u_h as $u_h^0 + u_h^*$ satisfying

$$[u_h^0]_\Gamma = 0 \quad [u_h^*]_\Gamma = J_1(\mathbf{x}),$$
$$\left[\beta \frac{\partial u_h^0}{\partial \mathbf{n}}\right]_\Gamma = 0 \quad \left[\beta \frac{\partial u_h^*}{\partial \mathbf{n}}\right]_\Gamma = J_2(\mathbf{x}). \tag{18}$$

Then the discrete weak formulation for (14) with the numerical solution $u_h \in \mathcal{B}_{h,0}(\Omega)$ is: Find $u_h^0 \in \widehat{S}_h(\Omega)$ such that

$$\sum_{T \in \mathcal{T}_h} \left(\int_{T^+} \nabla u_h^0 \cdot \nabla v_h \ \mathrm{d}\mathbf{x} + \int_{T^-} \nabla u_h^0 \cdot \nabla v_h \, \mathrm{d}\mathbf{x} \right)$$
$$= \sum_{T \in \mathcal{T}_h^I} \int_{\Gamma_{seg}} \left[\beta \frac{\partial u}{\partial \mathbf{n}}\right] v_h \, \mathrm{d}s + \sum_{T \in \mathcal{T}_h} \left(\int_T f v_h \, \mathrm{d}\mathbf{x} - \int_{T^\pm} \nabla u_h^* \cdot \nabla v_h \, \mathrm{d}\mathbf{x} \right),$$
$$\forall v_h \in \widehat{S}_h(\Omega). \tag{19}$$

To solve this problem, we need to construct $u_h^* \in \mathcal{B}_{h,0}(\Omega)$ corresponding to the exact nonhomogeneous part $u^* \in \mathcal{U}_0(\Omega)$. So, the above scheme is divided into three parts:

1. Choose a suitable u^* satisfying jump conditions.
2. Find a good approximation u_h^* to u^*.
3. Solve the fem formulation (19) by IFEM.

In the next section, we assume u^* is supported in a thin strip Ω_h^I near Γ, say the union of interface elements, and we suggest a convenient way to construct $u_h^* \in \mathcal{B}_{h,0}(\Omega_h^I)$.

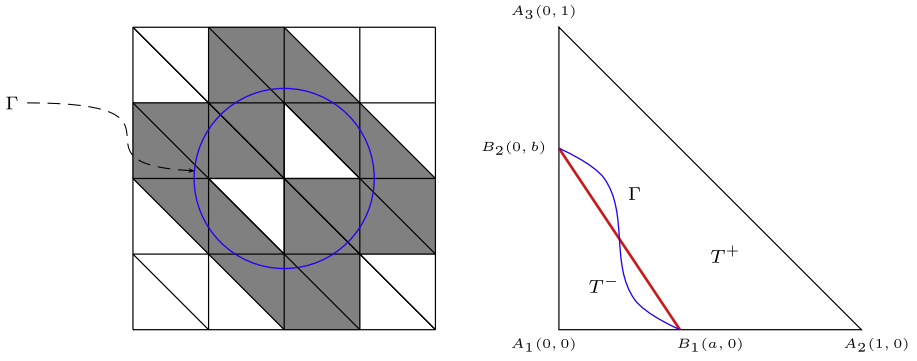
3.1. Construction of u_h^*

The idea is similar to the construction of immersed finite element method introduced in [2,3,25] to solve an interface problem with homogeneous jumps ($J_1 = J_2 = 0$). They use any shape regular grid so that the general interface is allowed to cut through elements. The basis functions are constructed to satisfy the homogeneous jump conditions.

The basis functions are piecewise linear but generally broken along the interface. Let $\mathcal{T}_h$ be the usual quasi-uniform finite element triangulations of the domain Ω. We call an element $T \in \mathcal{T}_h$ an *interface element* if the interface Γ passes through the interior of T, otherwise we call T a *non-interface element*. We shall assume the interface Γ meets the edges of an interface element $T \in \mathcal{T}_h^I$ at no more than two intersections. Now, we introduce some symbols (see Fig. 2(a)):

$$\mathcal{T}_h^I = \text{the set of all interface elements},$$
$$\mathcal{T}_h^N = \text{the set of all non-interface elements},$$
$$\Omega_h^I = \bigcup_{T \in \mathcal{T}_h^I} T.$$

Using the idea similar to construct basis for immersed finite element method, we shall construct a general piecewise function satisfying the nonhomogeneous jumps.



(a) The union Ω_h^I of all interface elements.

(b) A typical interface triangle T

Fig. 2. Interface elements in uniform triangular mesh.

Consider an interface element T. For simplicity, we assume the three vertices are given by $A_1 = (0,0)$, $A_2 = (1,0)$, $A_3 = (0,1)$ (see Fig. 2(b)). For any element T in general position, all the constructions to be presented below carries over through affine equivalence. Assume the interface meets with the element's edges at points B_1 and B_2.

Let ℓ_i be the usual linear Lagrange nodal basis function associated with the vertex A_i for $i = 1, 2, 3$. Then $\ell_1 = 1 - x - y$, $\ell_2 = x$, $\ell_3 = y$. For any given function ψ on $T \in \mathcal{T}_h^I$ satisfying jump conditions, we shall construct ψ_p which is linear on T^+ and T^- respectively, retaining vertex values $d_i = \psi(A_i)$ for $i = 1, 2, 3$, and satisfying the jump conditions on $\overline{B_1B_2}$. Since $\psi_p(A_i) = d_i$, we note that we can write ψ_p in the following form:

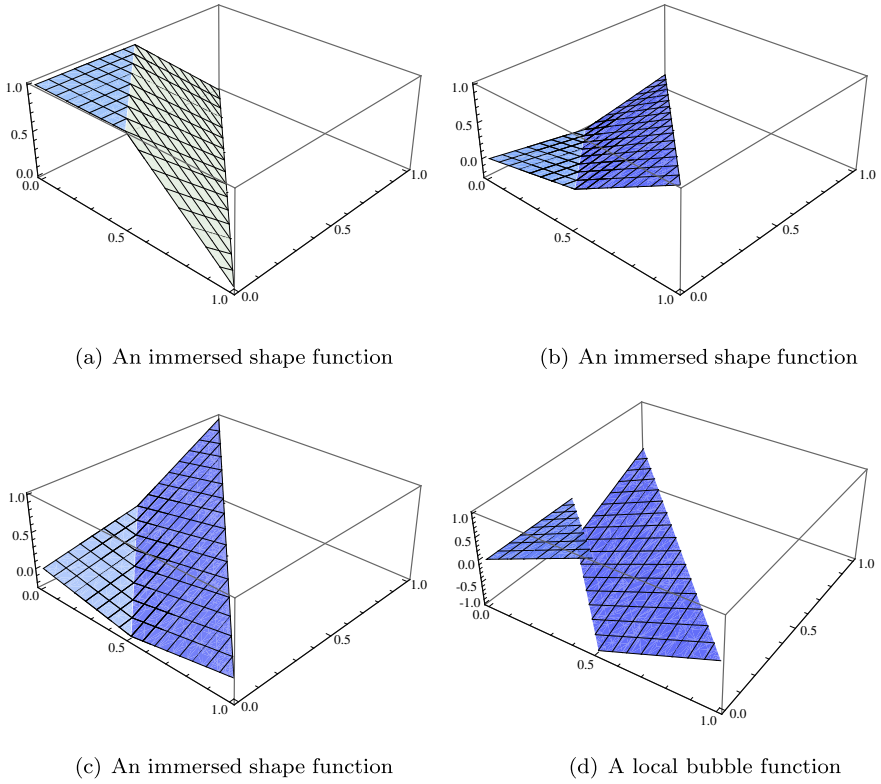
$$\psi_p = \begin{cases} \psi_p^- = d_1\ell_1 + c_2\ell_2 + c_3\ell_3 & \text{in } T^-, \\ \psi_p^+ = c_1\ell_1 + d_2\ell_2 + d_3\ell_3 & \text{in } T^+. \end{cases} \tag{20}$$

Now imposing the jump conditions on Γ, we have:

$$\psi_p^-(\mathbf{x}) - \psi_p^+(\mathbf{x}) = J_1(\mathbf{x}), \quad \mathbf{x} = B_1, B_2, \tag{21a}$$

$$\left\{\beta^- \frac{\partial \psi_p^-}{\partial \mathbf{n}} - \beta^+ \frac{\partial \psi_p^+}{\partial \mathbf{n}}\right\}_{\overline{B_1B_2}} = J_f := (J_2(B_1) + J_2(B_2))/2, \tag{21b}$$

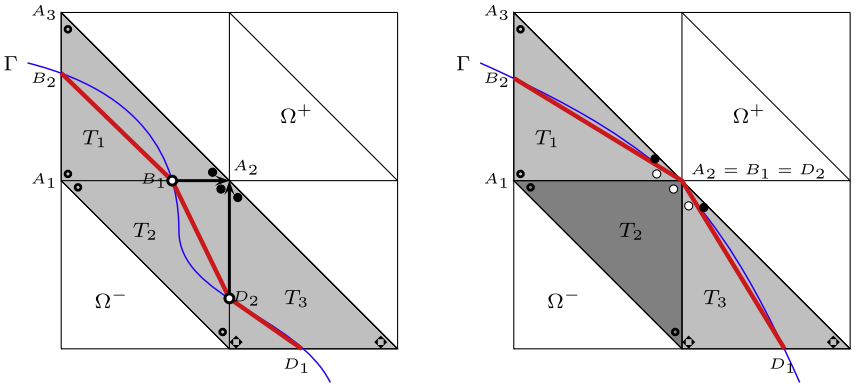
where the notation "$\{v\}$" refer to the average of a function v on the interface segment $\Gamma_{seg} = \overline{B_1B_2}$. Since ψ_p^- and ψ_p^+ are linear, (21b) can be written as:



(a) An immersed shape function (b) An immersed shape function

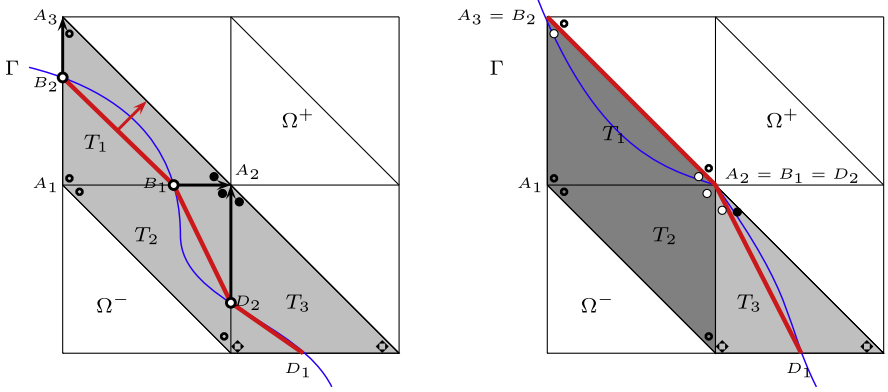
(c) An immersed shape function (d) A local bubble function

Fig. 3. (a), (b), (c) are immersed shape functions and the local bubble function on an interface element where $a = b = 0.5, \beta^- = 1, \beta^+ = 100, J_1(B_1) = 2, J_1(B_2) = 1$, $J_2(\overline{B_1B_2}) = 100$.



(a) Γ does not meet any vertices. B_1, D_2 move to a same point A_2 (b) Γ passes one vertex in each element T_1, T_2, T_3.

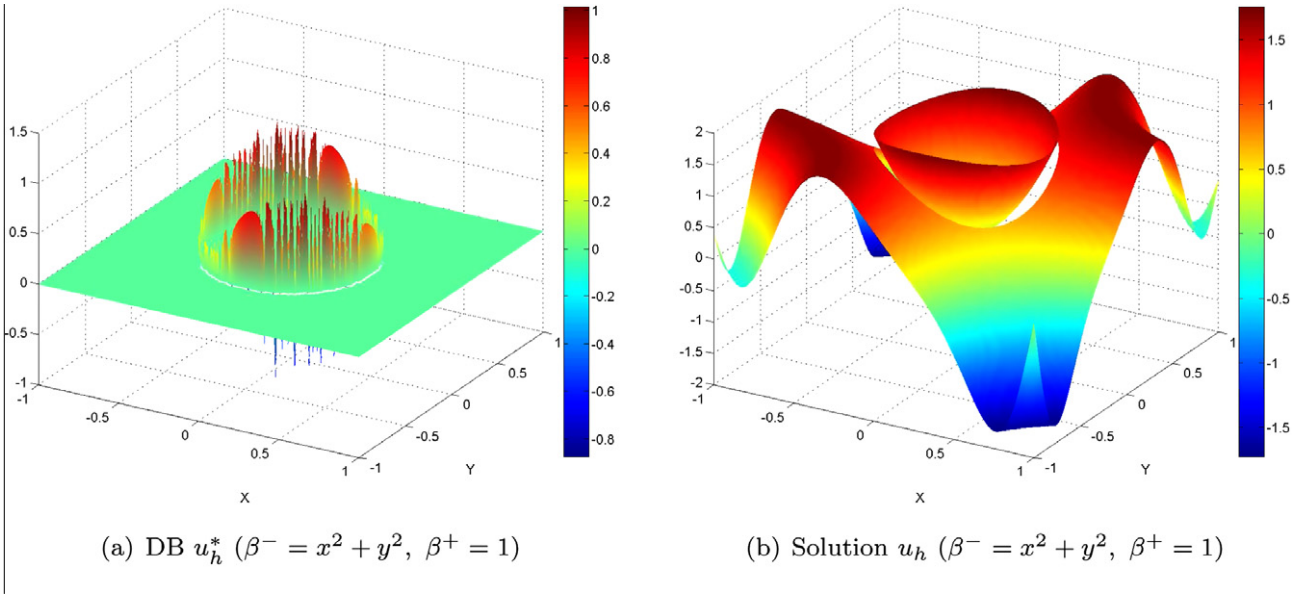
Fig. 4. The bubble function containing an interface vertex A_2.



(a) Γ does not meet any vertices. B_1, D_2 move to a same point A_2 and B_2 move to a point A_3

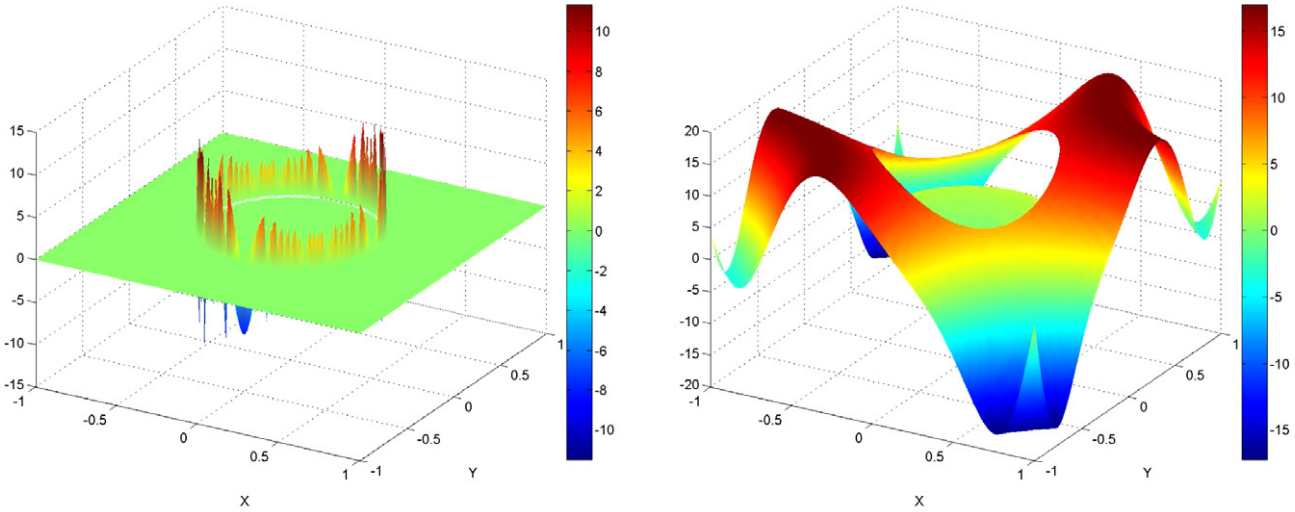
(b) Γ passes two vertices in the element T_1.

Fig. 5. The bubble function containing an interface edge $\overline{A_2A_3}$.



(a) DB u_h^* $(\beta^- = x^2 + y^2,\ \beta^+ = 1)$

(b) Solution u_h $(\beta^- = x^2 + y^2,\ \beta^+ = 1)$



(c) DB u_h^* $(\beta^- = x^2 + y^2,\ \beta^+ = 0.1)$

(d) Solution u_h $(\beta^- = x^2 + y^2,\ \beta^+ = 0.1)$

Fig. 6. A circle interface which passes four vertices.

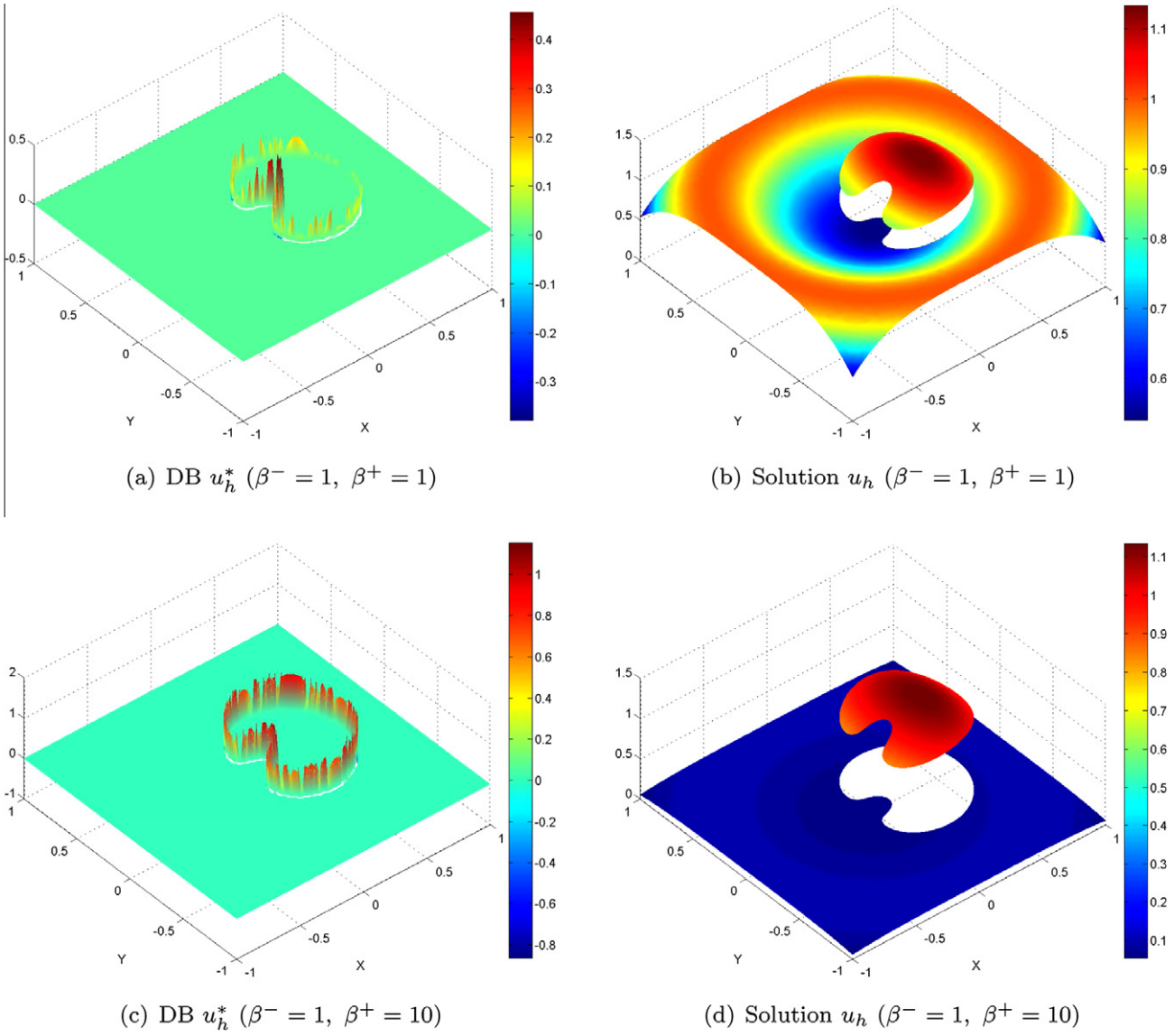


Fig. 7. A heart shape interface.

$$\overline{\beta^-}\frac{\partial \psi_p^-}{\partial \mathbf{n}} - \overline{\beta^+}\frac{\partial \psi_p^+}{\partial \mathbf{n}} = J_f,$$

where $\overline{\beta^-}, \overline{\beta^+}$ are the averages over the segment $\overline{B_1B_2}$. Hence our construction works naturally for variable coefficient. Now, the conditions (21a), (21b) become

$$\begin{aligned}
&d_1(1-a) + c_2 a = c_1(1-a) + d_2 a + J_1(B_1),\\
&d_1(1-b) + c_3 b = c_1(1-b) + d_3 b + J_2(B_2),\\
&\overline{\beta^-}(d_1\nabla\ell_1 + c_2\nabla\ell_2 + c_3\nabla\ell_3)\cdot\mathbf{n} = \overline{\beta^+}(c_1\nabla\ell_1 + d_2\nabla\ell_2 + d_3\nabla\ell_3)\cdot\mathbf{n} + J_f.
\end{aligned}$$

In matrix form, the coefficient c_i's are determined by the following relations:

$$\begin{aligned}
&\begin{bmatrix} a-1 & a & 0\\ b-1 & 0 & b\\ -\overline{\beta^+}v_1 & \overline{\beta^-}v_2 & \overline{\beta^-}v_3 \end{bmatrix}\begin{pmatrix} c_1\\ c_2\\ c_3 \end{pmatrix}\\
&= \begin{bmatrix} a-1 & a & 0\\ b-1 & 0 & b\\ -\overline{\beta^-}v_1 & \overline{\beta^+}v_2 & \overline{\beta^+}v_3 \end{bmatrix}\begin{pmatrix} d_1\\ d_2\\ d_3 \end{pmatrix} + \begin{pmatrix} J_1(B_1)\\ J_1(B_2)\\ J_f \end{pmatrix},
\end{aligned} \tag{22}$$

Table 1
(Circle) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
Case (a)	8×8	0.3124280	–	4.3886981	–
$\beta^- = x^2+y^2$,	16×16	0.0885818	1.82	2.2376150	0.97
$\beta^+ = 1$	32×32	0.0244866	1.86	1.1224202	1.00
	64×64	0.0065874	1.89	0.5632125	1.00
	128×128	0.0013879	2.25	0.2827034	0.99
	256×256	0.0003367	2.04	0.1420148	0.99
	512×512	0.0000882	1.93	0.0723010	0.97
Case (b)	8×8	3.6991611	–	39.625309	–
$\beta^- = x^2+y^2$,	16×16	1.1037592	1.75	20.951918	0.97
$\beta^+ = 0.1$	32×32	0.2806672	1.98	10.709964	0.99
	64×64	0.0728569	1.95	5.3930402	0.99
	128×128	0.0136462	2.42	2.7018477	1.00
	256×256	0.0031685	2.11	1.3518338	1.00
	512×512	0.0007942	2.00	0.6764209	1.00

where $v_i = \nabla\ell_i\cdot\mathbf{n}$, $(i = 1,2,3)$. Let $\mathcal{A}$ be the matrix on the left hand side. Then, we see it has unique solution since

$$\det(\mathcal{A}) = (a^2((1-b)\overline{\beta^-} + b\overline{\beta^+}) + b^2((1-a)\overline{\beta^-} + a\overline{\beta^+}))/\sqrt{a^2+b^2} > 0,$$

for $0 < \overline{\beta^+}, \overline{\beta^-}$ and $0 < a, b \leqslant 1$.

Table 2
(Heart shape) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
Case (a) $\beta^- = 1$, $\beta^+ = 1$	8×8	0.1596806	–	0.7770398	–
	16×16	0.0721703	1.15	0.4287888	0.86
	32×32	0.0295279	1.29	0.1929842	1.15
	64×64	0.0086040	1.78	0.0758858	1.35
	128×128	0.0020612	2.06	0.0354244	1.10
	256×256	0.0005715	1.85	0.0175126	1.02
	512×512	0.0000677	3.07	0.0086673	1.02
Case (b) $\beta^- = 1$, $\beta^+ = 10$	8×8	0.3644745	–	0.4036730	–
	16×16	0.2035256	0.84	0.2511432	0.69
	32×32	0.0893858	1.19	0.1091449	1.20
	64×64	0.0223604	2.00	0.0386975	1.50
	128×128	0.0053210	2.07	0.0156123	1.31
	256×256	0.0016342	1.70	0.0077259	1.02
	512×512	0.0001743	3.23	0.0039854	0.96

Table 3
(Starfish) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
$\beta^- = x^2 + y^2$, $\beta^+ = 1$	8×8	0.3984737	–	4.5337993	–
	16×16	0.1300373	1.61	2.3349810	0.95
	32×32	0.0429278	1.59	1.1596418	1.01
	64×64	0.0175488	1.29	0.5754026	1.01
	128×128	0.0015901	3.46	0.2831408	1.02
	256×256	0.0004215	1.92	0.1427670	0.99
	512×512	0.0000919	2.19	0.0721531	0.99

If the last term of right hand side of (22) is not present, then ψ_p constructed here is exactly the immersed finite element shape function proposed in [2,24] (see Fig. 3(a)–(c)).

Since Lagrange basis $\{\ell_i\}_{i=1,2,3}$ form a partition of unity ($\sum_{i=1}^{3}\ell_i \equiv 1$), we have:

$$\sum_{i=1}^{3} v_i = \sum_{i=1}^{3} \nabla \ell_i \cdot \mathbf{n} = \left(\nabla \sum_{i=1}^{3} \ell_i\right) \cdot \mathbf{n} = (\nabla 1) \cdot \mathbf{n} \equiv 0. \tag{23}$$

Then, solving the Eq. (22) with $d_j = \delta_{ij}$, we see the coefficients of immersed basis functions $\widehat{\psi}_i$ are represented as follows:

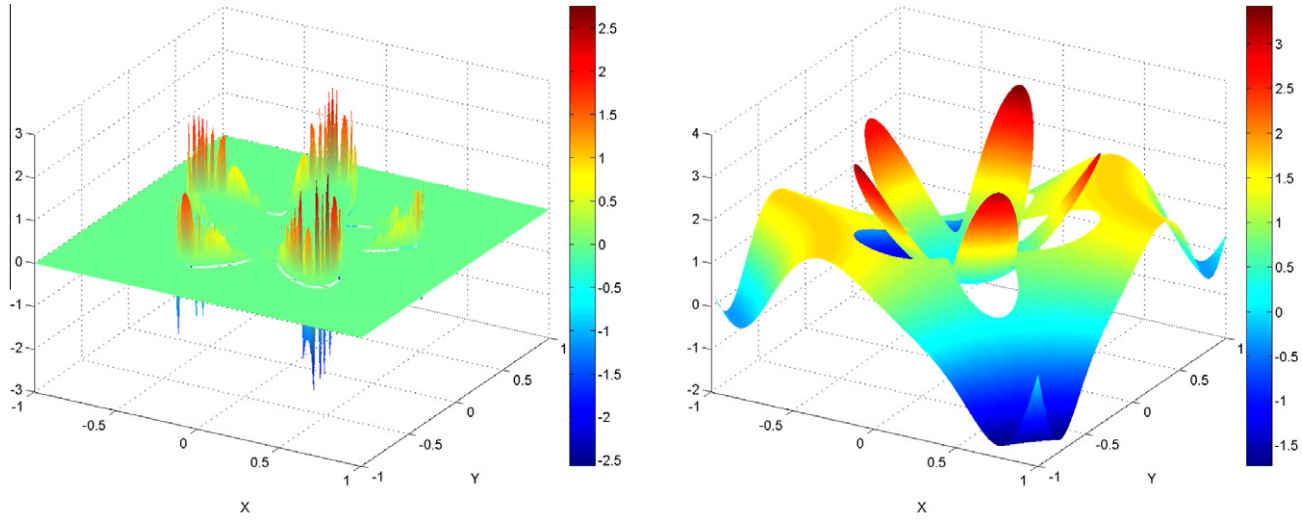
Table 4
(Many interface vertices) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
Case (a) $\beta^- = 1$, $\beta^+ = 1$	8×8	0.3903390	–	5.7814835	–
	16×16	0.1321147	1.56	3.2310644	0.83
	32×32	0.0365502	1.85	1.6795824	0.94
	64×64	0.0093927	1.96	0.8487105	0.98
	128×128	0.0023648	1.99	0.4255416	0.99
	256×256	0.0005922	2.00	0.2129317	1.00
	512×512	0.0001481	2.00	0.1064890	1.00
Case (b) $\beta^- = 1$, $\beta^+ = 1000$	8×8	0.3879398	–	5.7798689	–s
	16×16	0.1321048	1.55	3.2272795	0.84
	32×32	0.0364295	1.85	1.6782963	0.94
	64×64	0.0093603	1.96	0.8483104	0.98
	128×128	0.0023565	1.99	0.4254178	0.99
	256×256	0.0005901	1.99	0.2128912	1.00
	512×512	0.0001476	2.00	0.1064744	1.00
Case (c) $\beta^- = 1000$, $\beta^+ = 1$	8×8	0.0066192	–	0.0858238	–
	16×16	0.0017208	1.94	0.0432789	0.98
	32×32	0.0004386	1.97	0.0217335	0.99
	64×64	0.0001107	1.98	0.0108902	0.99
	128×128	0.0000278	1.99	0.0054509	0.99
	256×256	0.0000069	1.99	0.0027269	1.00
	512×512	0.0000017	2.00	0.0013638	1.00

$$\widehat{\psi}_1 : \begin{cases} c_{11} = (-\overline{\beta^-}(bv_2 + av_3))/K, \\ c_{12} = \left((-1+a)b(\overline{\beta^-} - \overline{\beta^+})(v_2+v_3)\right)/K, \\ c_{13} = \left(a(-1+b)(\overline{\beta^-} - \overline{\beta^+})(v_2+v_3)\right)/K, \end{cases}$$

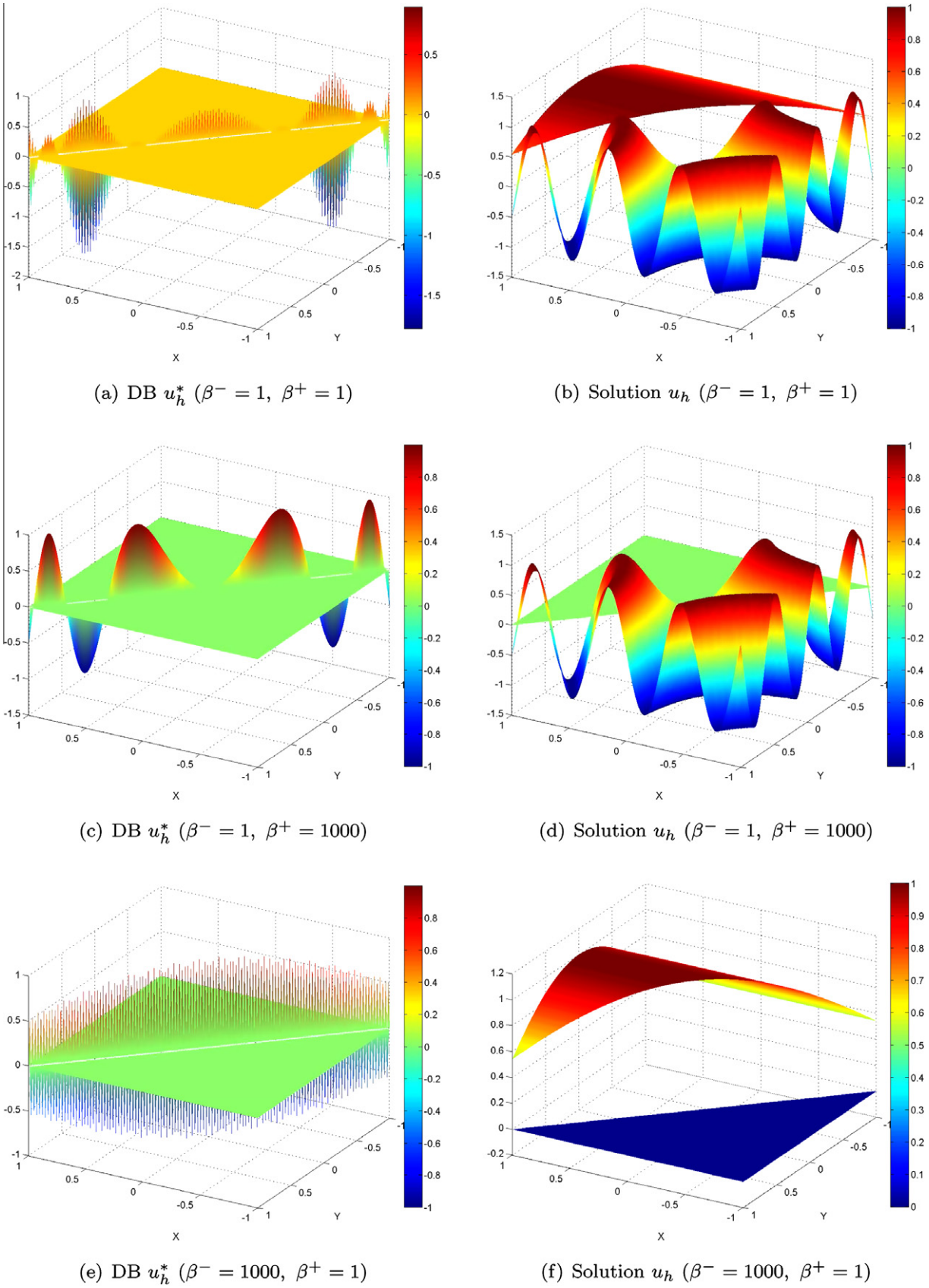
$$\widehat{\psi}_2 : \begin{cases} c_{21} = \left(ab(\overline{\beta^-} - \overline{\beta^+})v_2\right)/K, \\ c_{22} = \left(-b\overline{\beta^+}v_2 + a(\overline{\beta^-} - b\overline{\beta^-} + b\overline{\beta^+})v_3\right)/K, \\ c_{23} = \left(a(1-b)(\overline{\beta^-} - \overline{\beta^+})v_2\right)/K, \end{cases} \tag{24}$$

$$\widehat{\psi}_3 : \begin{cases} c_{31} = \left(ab(\overline{\beta^-} - \overline{\beta^+})v_3\right)/K, \\ c_{32} = \left((1-a)b(\overline{\beta^-} - \overline{\beta^+})v_3\right)/K, \\ c_{33} = \left(b((-1+a)\overline{\beta^-} - a\overline{\beta^+})v_2 - a\overline{\beta^+}v_3\right)/K, \end{cases}$$



(a) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1)$ (b) Solution u_h $(\beta^- = 1,\ \beta^+ = 1)$

Fig. 8. A starfish interface.



(a) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1)$

(b) Solution u_h $(\beta^- = 1,\ \beta^+ = 1)$

(c) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1000)$

(d) Solution u_h $(\beta^- = 1,\ \beta^+ = 1000)$

(e) DB u_h^* $(\beta^- = 1000,\ \beta^+ = 1)$

(f) Solution u_h $(\beta^- = 1000,\ \beta^+ = 1)$

Fig. 9. An interface which has large number of interface vertices.

where $K = -a\overline{\beta^-}v_3 + b(a(v_2+v_3)(\overline{\beta^-}-\overline{\beta^+}) - \overline{\beta^-})$. Then, we can easily check

$$\sum_{i=1}^{3}\widehat{\psi}_i(\mathbf{x}) = \sum_{i=1}^{3}\sum_{j=1}^{3}c_{ij}l_j(\mathbf{x}) = \sum_{j=1}^{3}\left(\sum_{i=1}^{3}c_{ij}\right)l_j(\mathbf{x}) = \sum_{j=1}^{3}(1)l_j(\mathbf{x}) \equiv 1.$$

Remark 2. In the above construction, we have assumed the variables a and b are given constants. For a moving interface such as Hele-Shaw problem, the variables a and b are functions of the solution. In this case, one can either compute a and b from the solution of previous time step (Forward Euler) as in [23] or compute by implicit scheme (such as Heun's Predictor–Corrector method): Let a^*, b^* be the interface points obtained by Forward Euler method. Then apply above method to find the solution. Call this solution $u_h^{1/2}$ (predictor step). Using $u_h^{1/2}$ as an intermediate step, we compute the evolving interface either by front tracking or level set method. Using this new interface, we solve the problem once more (correcting step). Since the cost of computing the bubble is almost nil, the computational complexity in this case is equivalent to solving elliptic problem once more.

If $d_i = 0$ ($i = 1,2,3$) in the Eq. (22), then we obtain an approximation to u^*. We call such a ψ_p the (local) *discontinuous bubble*(DB) because it vanishes on the vertices of Ω_h^l(see Fig. 3(d)). Since the cost of solving this 3×3 element level system is almost nil, the total cost is similar to that of standard elliptic problem which is a big advantage of our scheme. Our scheme is different from the one in [18] where the corresponding discrete singular function is globally supported. Furthermore, we can even handle the case when the interface passes one or more vertices. So far, we have assumed the interface does not pass any vertex. When the interface passes vertices, we need to look more carefully; these situations can be understood as limiting cases which will be explained in detail below.

3.2. The interface passes one vertex (interface vertex)

We assume $a = 1$ in Fig. 2(b) so that $B_1 = A_2$, i.e., the interface passes one vertex. In this case, we see the Eq. (22) with $d_i = 0$ becomes:

$$\begin{bmatrix} 0 & 1 & 0 \\ b-1 & 0 & b \\ \frac{(b+1)\overline{\beta^+}}{\sqrt{1+b^2}} & \frac{b\overline{\beta^-}}{\sqrt{1+b^2}} & \frac{\overline{\beta^-}}{\sqrt{1+b^2}} \end{bmatrix}\begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} J_1(B_1) \\ J_1(B_2) \\ J_f \end{pmatrix}. \tag{25}$$

This system clearly has a unique solution. We note that ψ_p determined by:

$$\psi_p = \begin{cases} \psi_p^- = c_2 l_2 + c_3 l_3, & \text{in } T_1^-, \\ \psi_p^+ = c_1 l_1, & \text{in } T_1^+, \end{cases}$$

has support on T_1, but it has multi-values at A_2, i.e., $\psi_p(A_2)|_{T_1^-} = J_1(B_1)$ and $\psi_p(A_2)|_{T_1^+} = 0$. This can be regarded as a limit case as the point B_1 approaches A_2 (see Fig. 4). Similar computation determines ψ_p on T_3. The function ψ_p on T_2 is naturally determined by continuity.

Next, we consider the case $a = b = 0$. Then, we see (22) does not define $\{c_i\}$. However, this corresponds to the limiting case as B_1, $B_2 \Rightarrow A_1$ in Fig. 4(a). This is the same situation as B_1, $D_2 \Rightarrow A_2$ on T_2. Hence the value of ψ_p at A_2 on T_2 is naturally determined by continuity from T_1, T_3 and the bubble condition on the other vertices determines ψ_p on T_2.

3.3. The interface passes two vertices (interface edge)

Let us consider the case when interface passes two vertices as in Fig. 5(b). This corresponds to the case $a = b = 1$ in the Eq. (22) with $d_i = 0$ for $i = 1, 2, 3$. Hence ψ_p is uniquely determined on T_1 and similar construction works on T_3. This can be again viewed as a limiting case where B_1 approaches A_2, and B_2 approaches A_3(of course, D_2 approaches A_2). Finally, we extend it continuously on T_2. (see Figs. 6 and 7).

Remark 3. In the actual computation of ψ_p, we have to be consistent in choosing the viewpoint: Whenever the interface meet with vertex (or vertices), it must be viewed as a limiting case of the same side of subdomain; For example, we always view it as the limit from Ω^-. If we are not consistent, i.e., if we view it as the limit from Ω^+ on T_1, and as the limit from Ω^- on T_3, then we run into conflict of vertex values at A_2.

4. Numerical experiments

In all of the experiments, the domain is $(-1,1)\times(-1,1)$ and triangularized by uniform triangle grids with $h_x = h_y = 1/2^{n-1}$ for $n = 3,\ldots,9$. In order to describe the interface, we consider the level-set function $\Phi(\mathbf{x})$ for the interface Γ which is assumed to be smooth. Let $\Phi : \Omega \to \mathbb{R}$ be a continuous function such that

$$\Phi(\mathbf{x}) = \begin{cases} <0 & \mathbf{x} \quad \text{in } \Omega^-, \\ =0 & \mathbf{x} \quad \text{on } \Gamma, \\ >0 & \mathbf{x} \quad \text{in } \Omega^+. \end{cases} \tag{26}$$

We assume that $\Phi(\mathbf{x})$ is smooth and $\nabla\Phi$ is not zero in any neighborhood of the interface Γ. Then the unit normal vector $\mathbf{n}(\mathbf{x})$ is represented by $\frac{\nabla\Phi}{|\nabla\Phi|}$. In all of the figures in these examples, we draw the discontinuous bubbles on the left column, while the right column have solutions.

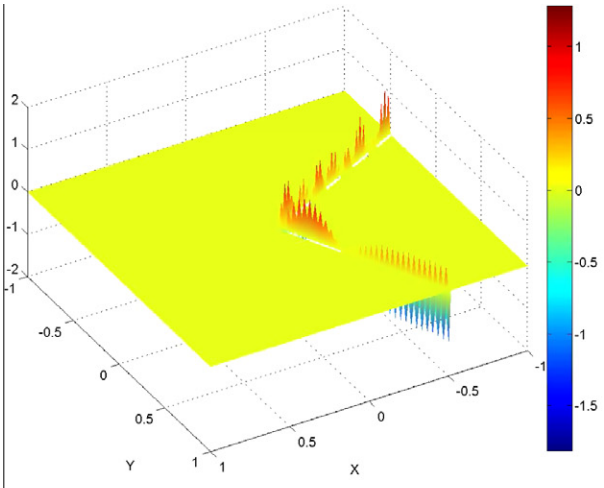
Example 1 (*Circle*). The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

$$\begin{aligned} &\Phi(\mathbf{x}) = \sqrt{x^2+y^2} - 0.5, \\ &(a)\ \beta^- = x^2+y^2, \quad \beta^+ = 1, \\ &(b)\ \beta^- = x^2+y^2, \quad \beta^+ = 0.1, \\ &u^- = 6x^2+7y^2, \quad u^+ = \frac{(\sin(3xy)+\cos(5x^2y^2))}{\beta^+}. \end{aligned} \tag{27}$$

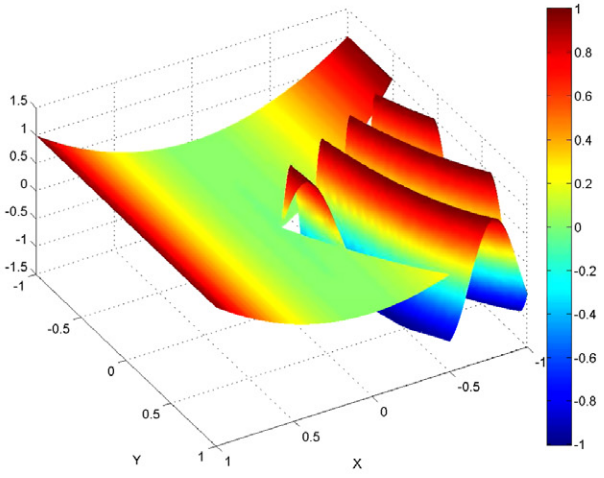
This interface has four interface vertices. Note that the interface vertex points are $(\pm 1/2, 0)$, $(0, \pm 1/2)$. We observe the robust 2nd order in L^2 and 1st order convergence in H^1-norm (see Table 1).

Table 5
(Kink interface) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

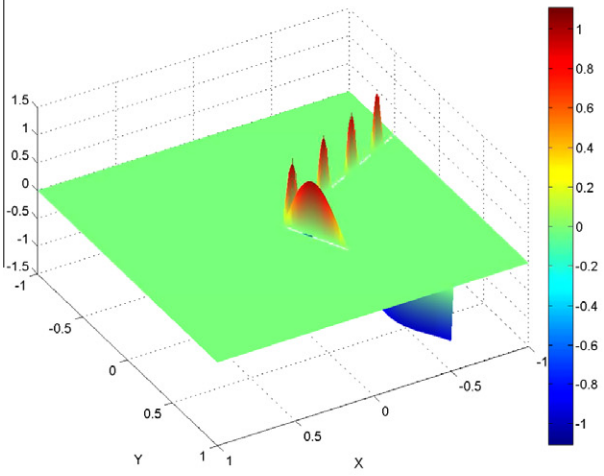
	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
Case (a) $\beta^- = 1$, $\beta^+ = 1$	8×8	0.7588050	–	12.222602	–
	16×16	0.4052782	0.91	6.7643648	0.85
	32×32	0.1137956	1.83	3.6674650	0.88
	64×64	0.0213542	2.41	1.8710598	0.97
	128×128	0.0044784	2.25	0.9423337	0.99
	256×256	0.0010532	2.09	0.4722868	1.00
	512×512	0.0002588	2.03	0.2363261	1.00
Case (b) $\beta^- = 1$, $\beta^+ = 1000$	8×8	0.8593644	–	14.189973	–
	16×16	0.1893472	2.18	6.8193067	1.06
	32×32	0.0618386	1.61	3.7891892	0.85
	64×64	0.0162176	1.93	1.8974015	1.00
	128×128	0.0040655	2.00	0.9493310	1.00
	256×256	0.0010113	2.01	0.4740717	1.00
	512×512	0.0002517	2.01	0.2367674	1.00
Case (c) $\beta^- = 1000$, $\beta^+ = 1$	8×8	0.0185411	–	0.2639868	–
	16×16	0.0039237	2.24	0.1283171	1.04
	32×32	0.0009531	2.04	0.0633187	1.02
	64×64	0.0002514	1.92	0.0315504	1.01
	128×128	0.0000665	1.92	0.0158155	1.00
	256×256	0.0000172	1.94	0.0079709	0.99
	512×512	0.0000044	1.95	0.0040528	0.98



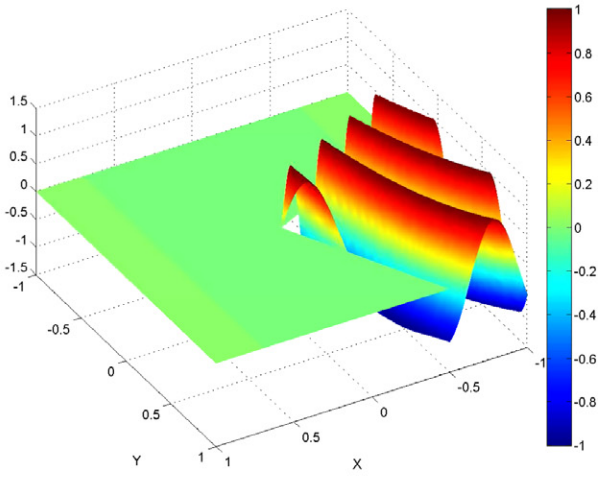
(a) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1)$



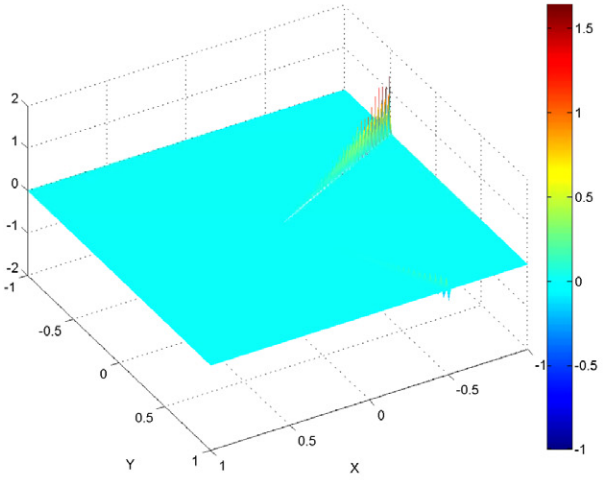
(b) Solution u_h $(\beta^- = 1,\ \beta^+ = 1)$



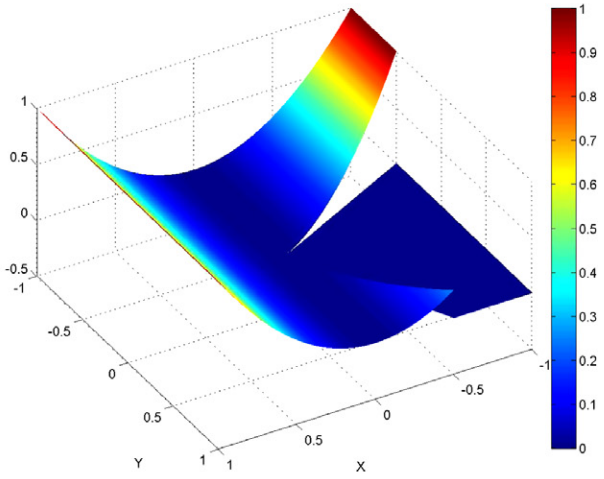
(c) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1000)$



(d) Solution u_h $(\beta^- = 1,\ \beta^+ = 1000)$

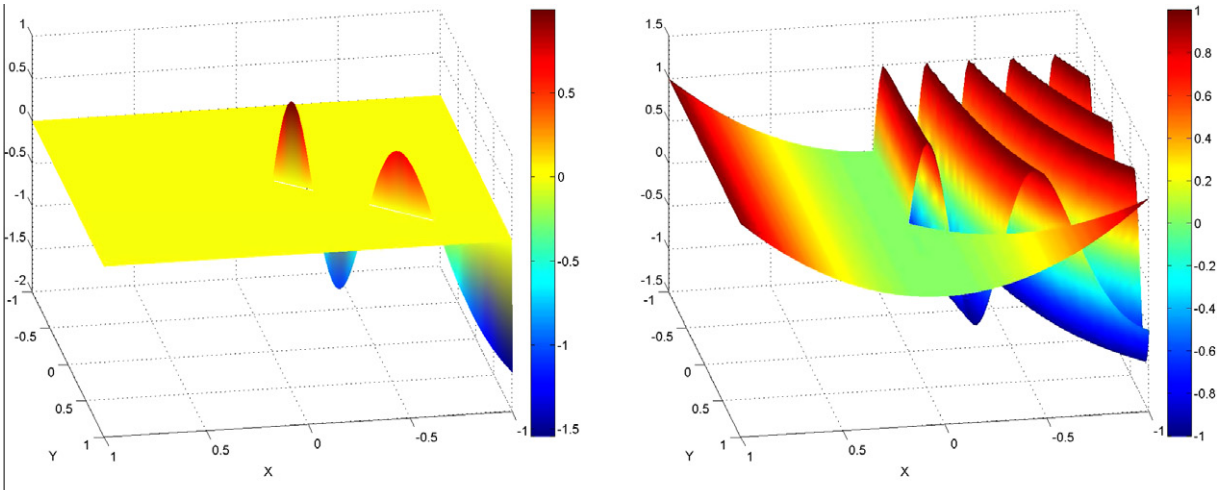


(e) DB u_h^* $(\beta^- = 1000,\ \beta^+ = 1)$

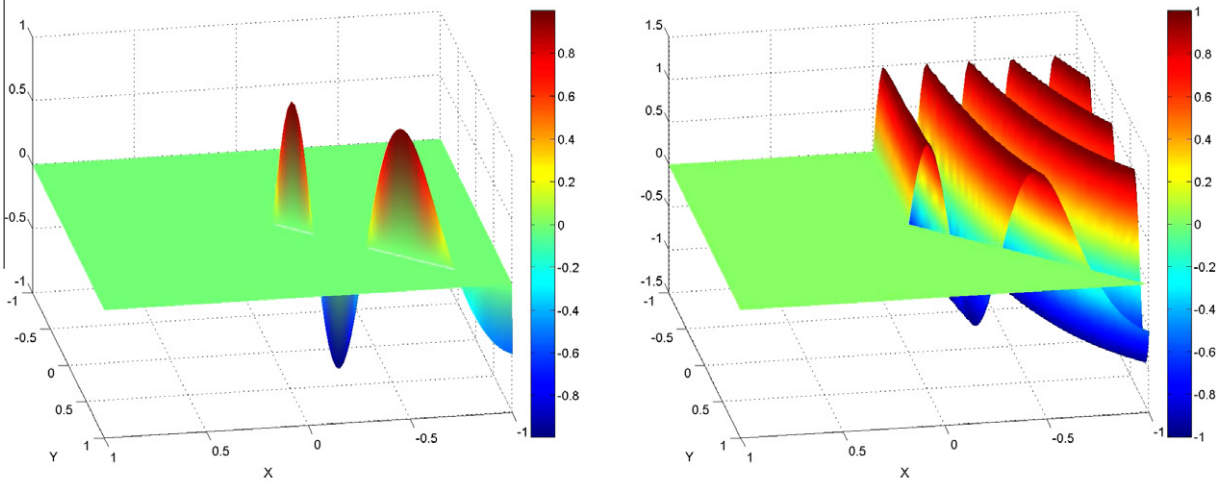


(f) Solution u_h $(\beta^- = 1000,\ \beta^+ = 1)$

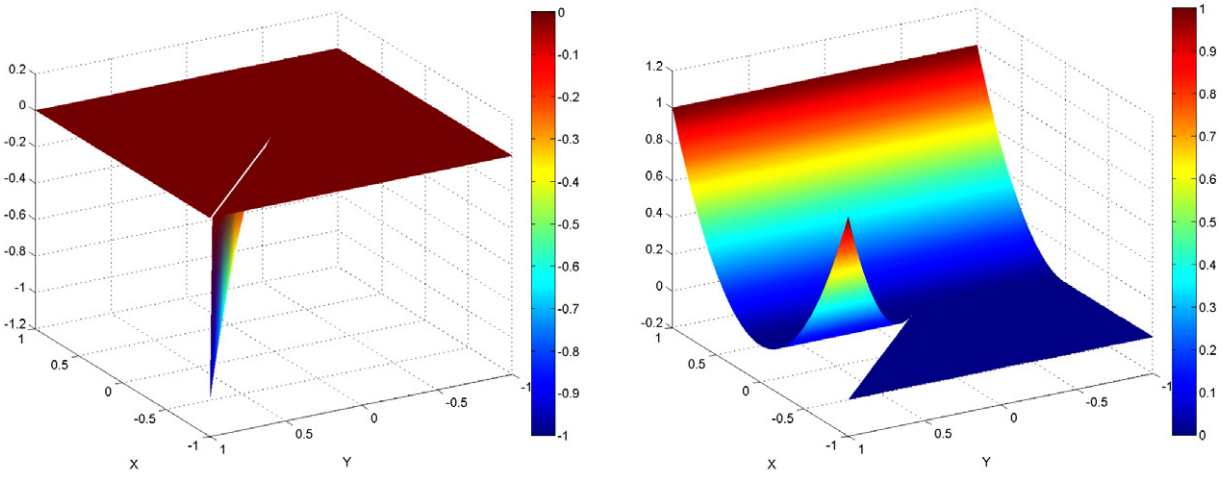
Fig. 10. A kink interface.



(a) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1)$ (b) Solution u_h $(\beta^- = 1,\ \beta^+ = 1)$



(c) DB u_h^* $(\beta^- = 1,\ \beta^+ = 1000)$ (d) Solution u_h $(\beta^- = 1,\ \beta^+ = 1000)$



(e) DB u_h^* $(\beta^- = 1000,\ \beta^+ = 1)$ (f) Solution u_h $(\beta^- = 1000,\ \beta^+ = 1)$

Fig. 11. An interface edge (constant coefficient).

Example 2 (*Heart shape*). In this example, the interface is heart shaped, but no interface vertex arises. The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

$$\begin{aligned}&\Phi(\mathbf{x}) = (3(x^2+y^2)-x)^2 - (x^2+y^2) + 0.1,\\ &(a)\ \beta^- = 1, \quad \beta^+ = 1,\\ &(b)\ \beta^- = 1, \quad \beta^+ = 10,\\ &u^- = \sin(2x^2+y^2+2)+x, \quad u^+ = \frac{\cos(1-x^2-y^2)}{\beta^+}.\end{aligned} \tag{28}$$

We again have similar optimal convergence behavior (see Table 2).

Example 3 (*Starfish shape*). In this example, the interface is starfish shaped with no vertex. The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

$$\begin{aligned}&\Phi(\mathbf{x}) = \sqrt{x^2+y^2} - 0.2\sin(5\theta - \pi/5) - 0.5,\\ &\beta^- = x^2+y^2, \quad \beta^+ = 1,\\ &u^- = 6x^2+7y^2, \quad u^+ = \sin(3xy) + \cos(5x^2y^2),\end{aligned} \tag{29}$$

where $\theta = \arctan(y/x)$.

This example also show $O(h^2)$ error in L^2 and $O(h^1)$ error in H^1-norm (see Table 3).

Example 4 (*Large number of interface vertices*). The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

$$\begin{aligned}&\Phi(\mathbf{x}) = x - y,\\ &(a)\ \beta^- = 1, \quad \beta^+ = 1,\\ &(b)\ \beta^- = 1, \quad \beta^+ = 1000,\\ &(c)\ \beta^- = 1000, \quad \beta^+ = 1,\\ &u^- = \frac{\sin(10xy)}{\beta^-}, \quad u^+ = \frac{\cos(y)}{\beta^+}.\end{aligned} \tag{30}$$

In this case, we see every vertex along the diagonal line $y = x$ is on interface vertex. Hence, for $2^n \times 2^n$ ($n = 3,\ldots,9$) grid, there are $2^n + 1$ interface vertices. Also this example is different from examples in most other papers; the interface is not a closed curve, i.e., it cuts the boundary of the domain Ω. Still, our extended bubble function catches the jump discontinuity very well. The Table 4 shows perfect order of convergence, i.e., exactly 2nd order in L^2-norm and 1st order in H^1-norm (see Figs. 8 and 9).

Example 5 (*Kink interface*). In this example, we consider the case with a kink which corresponds to a vertex. The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

$$\begin{aligned}&\Phi(\mathbf{x}) = \begin{cases} y + 2x, & \text{if } y \geqslant 0,\\ x - 2y, & \text{if } y < 0,\end{cases}\\ &(a)\ \beta^- = 1, \quad \beta^+ = 1,\\ &(b)\ \beta^- = 1, \quad \beta^+ = 1000,\\ &(c)\ \beta^- = 1000, \quad \beta^+ = 1,\\ &u^- = \frac{\sin(10x(y-2))}{\beta^-}, \quad u^+ = \frac{x^2}{\beta^+}.\end{aligned}$$

The kink is (0,0) and Table 5 show 2nd order convergence in L^2 and 1st order in H^1-norm (see Figs. 10 and 11).

Example 6 (*Interface edge (constant coefficient)*). We consider the case where the interface meets two vertices of elements. In this case the interface is aligned with finite element meshes. The construction of DB is naturally obtained as a limit of non-interface edge case. The coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are same as (31). The level-set function $\Phi(\mathbf{x})$ are given as follows:

$$\Phi(\mathbf{x}) = \begin{cases} 1 + x + y, & \text{if } y \geqslant 0,\\ x, & \text{if } y < 0.\end{cases} \tag{32}$$

Table 6 shows 2nd order convergence in L^2 and 1st order in H^1-norm.

Example 7 (*Interface edge (variable coefficient)*). We consider a problem similar to Example 6, but with variable coefficient. The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

$$\begin{aligned}&\Phi(\mathbf{x}) = \begin{cases} 1 + x + y, & \text{if } y \geqslant 0,\\ x, & \text{if } y < 0,\end{cases}\\ &\beta^- = x^2 + 3, \quad \beta^+ = x^2 - y^2 + 3,\\ &u^- = x^2 + y^2 + 2, \quad u^+ = 1 - x^2 - y^2.\end{aligned} \tag{33}$$

Table 7 shows robust 2nd order convergence in L^2 and 1st order in H^1-norm.

Table 6
(Interface edge (constant coefficient)) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
Case (a) $\beta^- = 1$, $\beta^+ = 1$	8×8	2.0062850	–	22.258706	–
	16×16	0.4676531	2.10	14.909455	0.57
	32×32	0.1173290	1.99	7.5409868	0.98
	64×64	0.0303525	1.95	3.8441953	0.97
	128×128	0.0076558	1.98	1.9314622	0.99
	256×256	0.0019182	1.99	0.9669059	0.99
	512×512	0.0004798	2.00	0.4835999	1.00
Case (b) $\beta^- = 1$, $\beta^+ = 1000$	8×8	1.5342145	–	21.148001	–
	16×16	0.4517405	1.76	14.904000	0.50
	32×32	0.1170623	1.94	7.5404862	0.98
	64×64	0.0303779	1.94	3.8440567	0.97
	128×128	0.0076666	1.98	1.9314055	0.99
	256×256	0.0019212	1.99	0.9668791	0.99
	512×512	0.0004805	2.00	0.4835867	1.00
Case (c) $\beta^- = 1000$, $\beta^+ = 1$	8×8	0.0180073	–	0.2295807	–
	16×16	0.0041460	2.11	0.1150814	0.99
	32×32	0.0010529	1.97	0.0575507	1.00
	64×64	0.0002641	1.99	0.0287850	1.00
	128×128	0.0000660	1.99	0.0143937	1.00
	256×256	0.0000165	2.00	0.0071970	1.00
	512×512	0.0000041	2.00	0.0035985	1.00

Table 7
(Interface edge (variable coefficient)) 2nd order with $\|\cdot\|_{L^2}$ and 1st order with $\|\cdot\|_{1,h}$.

	$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order
$\beta^- = x^2 + 3$, $\beta^+ = x^2 - y^2 + 3$	8×8	0.0408780	–	0.4079997	–
	16×16	0.0102287	1.99	0.2040860	0.99
	32×32	0.0025578	2.00	0.1020564	1.00
	64×64	0.0006395	2.00	0.0510302	1.00
	128×128	0.0001598	2.00	0.0255154	1.00
	256×256	0.0000399	2.00	0.0127577	1.00
	512×512	0.0000099	2.00	0.0063788	1.00

Table 8
1.8th order with $\|\cdot\|_{L^2}$, 0.8th order with $\|\cdot\|_{1,h}$ and 0.98th order with $\|\cdot\|_{L^\infty}$.

$N_x \times N_y$	$\|u-u_h\|_{L^2}$	Order	$\|u-u_h\|_{1,h}$	Order	$\|u-u_h\|_{L^\infty}$	Order
8×8	0.0988626	–	0.6765991	–	0.1062054	–
16×16	0.0250940	1.97	0.3402690	0.99	0.0356546	1.575
32×32	0.0063522	1.98	0.1719628	0.98	0.0117709	1.599
64×64	0.0016133	1.97	0.0879560	0.96	0.0046348	1.345
128×128	0.0004138	1.96	0.0459663	0.93	0.0023228	0.997
256×256	0.0001089	1.92	0.0248758	0.88	0.0011698	0.990
512×512	0.0000308	1.82	0.0141477	0.81	0.0005898	0.988

Example 8. Finally, we solve a problem from [18] and compare the results. The level-set function $\Phi(\mathbf{x})$, the coefficients $\beta^{\pm}$ and the solution $u^{\pm}$ are given as follows:

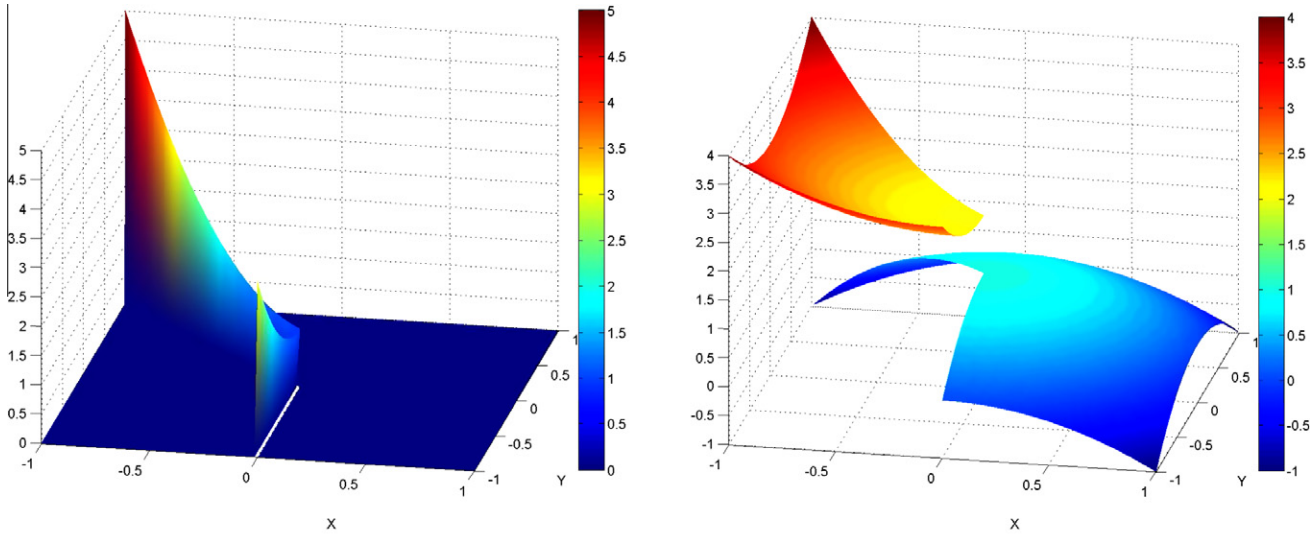
$$\begin{aligned} \Phi(\mathbf{x}) &= \begin{cases} y-2x, & \text{if } x+y>0, \\ y+\frac{1}{2}x, & \text{if } x+y\leqslant 0, \end{cases} \\ \beta^- &= 2+\sin(x+y), \quad \beta^+ = 1, \\ u^- &= (x^2+y^2)^{5/6}+\sin(x+y), \qquad u^+ = 8. \end{aligned} \tag{34}$$

In this example, we report L^2, H^1, L^∞ norm measured in the whole domain in Table 8. Comparing the order 0.88 in L^∞-norm measured away from interface available from [18], our scheme is clearly superior (0.988 in the whole domain) (see Figs. 12 and 13).

Remark 4. Finally, we mention briefly on how our scheme can be generalized to the case where the domain is partitioned into rectangular elements. We first note that the immersed finite element method for problems with no jumps was considered in [3]. For problems with jumps, one can use an equation similar to (22) to determine a bubble function which satisfies the jumps across the interface. A detailed study will be reported elsewhere.

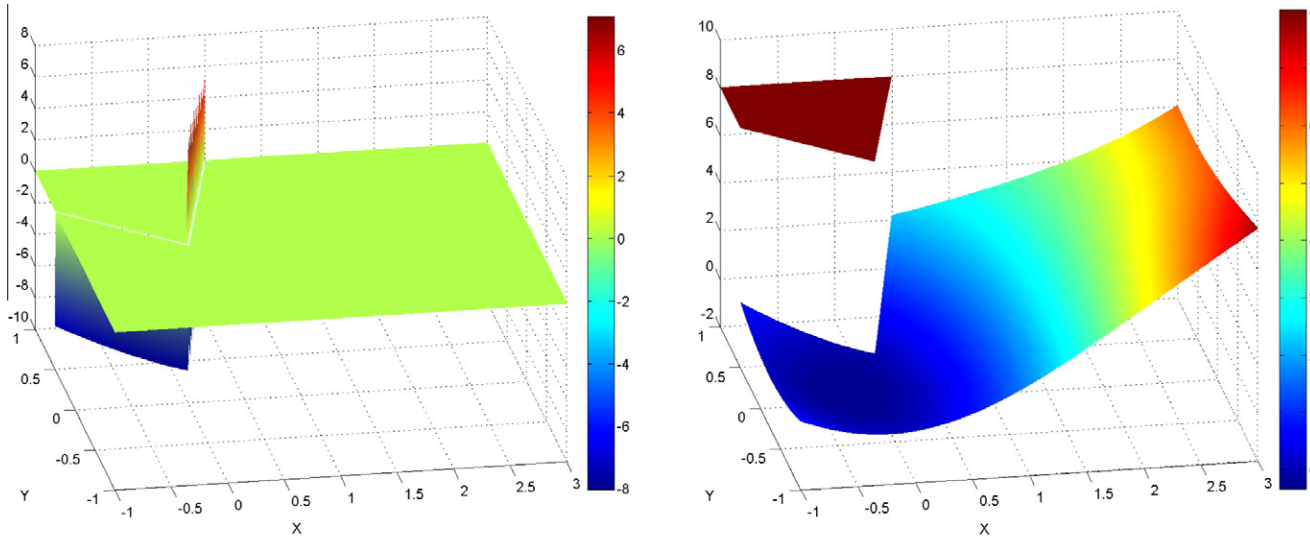
5. Conclusion

In this paper, we have introduced a new numerical method of solving elliptic interface problems having jump discontinuity in the solution and flux. The basic idea is to construct a piecewise linear function having small support near the interface which satisfies



(a) DB u_h^* $(\beta^- : x^2+3,\ \beta^+ : x^2-y^2+3)$ (b) Solution u_h $(\beta^- : x^2+3,\ \beta^+ : x^2-y^2+3)$

Fig. 12. An interface edge (variable coefficient).



(a) DB u_h^* $(\beta^- : 2+\sin(x+y),\ \beta^+ : 1)$ (b) Solution u_h $(\beta^- : 2+\sin(x+y),\ \beta^+ : 1)$

Fig. 13. The example 18 in [18].

the jump conditions. Then, removing it from the original variational form, we obtain an interface problem with homogeneous jumps. The equation is then solved with IFEM introduced in [2,3] which is shown to be effective in many situations [1,24,25]. Hence it is cheap to implement and very useful in a situation where one has to solve the same problem with different jump conditions many times. Moreover, to the authors' knowledge, our scheme is the first successful attempt to handle the case where the interface meets with one or more vertices. Also, our scheme for the interface vertices (edges) can be interpreted as the limit of interior interface. Numerical experiments show exactly second-order in L^2-norm and first order in H^1-norm, robust for all examples above.

References

[1] Z. Li, T. Lin, X. Wu, New Cartesian grid methods for interface problems using the finite element formulation, Numer. Math. 96 (2003) 61–98.
[2] Z. Li, T. Lin, Y. Lin, R.C. Rogers, An immersed finite element space and its approximation capability, Numer. Methods. Partial Differ. Equat. 20 (2004) 338–367.
[3] X. He, T. Lin, Y. Lin, Approximation capability of a bilinear immersed finite element space, Inc. Num. Methods Partial Differ. Equat. 24 (2008) 1265–1300.
[4] R.V. Davalosa, B. Rubinskya, L.M. Mirb, Theoretical analysis of the thermal effects during in vivo tissue electroporation, Bioelectrochemistry 61 (2003) 99–107.
[5] I. Lackovic, R. Magjarevic, D. Miklavcic, Analysis of tissue heating during electroporation based therapy: a 3D FEM model for a pair of needle electrodes, IFMBE Proc. 16 (2007) 631–634.
[6] C.S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (3) (1977) 220–252.
[7] C.S. Peskin, Lectures on mathematical aspects of physiology, Lect. Appl. Math. 19 (1981) 69–107.
[8] C. Tu, C.S. Peskin, Stability and instability in the computation of flows with moving immersed boundaries: a comparison of three methods, SIAM J. Sci. Stat. Comput. 13 (1992) 1361–1376.
[9] Ming-Chih Lai, Charles Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2) (2000) 705–719.
[10] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (1994) 1019–1044.
[11] T.Y. Hou, Z.L. Li, S. Osher, H. Zhao, A hybrid method for moving interface problems with application to the HeleShaw flow, J. Comput. Phys. 134 (1997) 236–252.
[12] R.J. LeVeque, Z. Li, Immersed interface method for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput. 18 (1997) 709–735.
[13] R.J. LeVeque, C. Zhang, Immersed interface methods for wave equations with discontinuous coefficients, Wave Motion 25 (1997) 237–263.
[14] L. Lee, R.J. Leveque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (3) (2003) 832–856.
[15] Z. Li, Immersed interface method for moving interface problems, Numer. Algorithms 14 (1997) 269–293.
[16] Petter Andreas Berthelsen, A decomposed immersed interface method for variable coefficient elliptic equations with non-smooth and discontinuous solutions, J. Comput. Phys. 197 (2004) 364–386.
[17] Y. Gong, B. Li, Z. Li, Immersed-interface finite-element methods for elliptic interface problems with nonhomogeneous jump conditions, SIAM J. Numer. Anal. 46 (1) (2008) 472–495.
[18] S. Hou, X. Liu, A numerical method for solving variable coefficient elliptic equation with interfaces, J. Comput. Phys. 202 (2) (2005) 411–445.
[19] M. Lai, Z. Li, X. Lin, Fast solvers for 3D Poisson equations involving interfaces in a finite or the infinite domain, J. Comput. Appl. Math. 191 (1) (2006) 106–125.
[20] Z. Li, A fast iterative algorithm for elliptic interface problems, SIAM J. Numer. Anal. 35 (1998) 230–254.
[21] X. Liu, T.C. Sideris, Convergence of the ghost fluid method for elliptic equations with interfaces, Math. Comput. 72 (244) (2003) 1731–1746.
[22] Z. Qiao, Z. Li, T. Tang, A finite difference scheme for solving the nonlinear Poisson–Boltzmann equation modeling charged spheres, J. Comput. Math. 24 (3) (2006) 252–264.
[23] T.Y. Hou, Z. Li, S. Osher, H. Zhao, A hybrid method for moving interface problems with application to the Hele–Shaw flow, J. Comput. Phys. 123 (1997) 236–252.
[24] K.T. Wee, The immersed Interface Method for Elliptic Equations with Discontinuous Coefficients, Ph. D. Thesis, Korea Advanced Institute of Science and Technology, 2007.
[25] S.H. Chou, D.Y. Kwak, K.T. Wee, Optimal convergence aanalysis of an immersed interface finite element method, Adv. Comput. Math., (in press).
[26] G.J. Wagner, S. Ghosal, W.K. Liu, Particulate flow simulations using lubrication theory solution enrichment, Int. J. Numer. Methods Engrg. 56 (9) (2003) 1261–1289.
[27] A. Gerstenberger, W.A. Wall, An extended finite element method/Lagrnage multiplier based approach for fluid-structure interaction, Comput. Methods Appl. Mech. Engrg. 197 (2008) 1699–1714.
[28] J.A. Roitberg, Z.G. Seftel, A homeomorphism theorem for elliptic systems, and its applicaitons, Mat, Sb. 78 (1969) 446–472.