



Locally conservative discontinuous bubble scheme for Darcy flow and its application to Hele-Shaw equation based on structured grids

Yoonjeong Choi¹ · Gwanghyun Jo²  · Do Y. Kwak³ · Young Ju Lee⁴

Received: 5 July 2021 / Accepted: 9 May 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

In this paper, we present an algorithm to solve the Darcy flow coupled with a transport for the interface tracking and apply the algorithm to solve a Hele-Shaw flow. The main challenge in the solution of the Hele-Shaw flow can be found at the change in the jump of the pressure along with the moving interface. We notice that such a challenge can be adequately handled by maintaining the conservation of the flux. Our algorithm employs the immersed finite element method equipped with the enrichment of piecewise constants to guarantee the conservative flux while the change of the jump condition for the pressure is handled via discontinuous bubble function, non-zero only near the interface. On the other hand, the interface motion is modeled and solved by the level set framework and WENO scheme. One important advantage of the proposed scheme is that the resulting algebraic system is efficiently handled by a proven-to-be fast and optimal algorithm in time evolution. A number of numerical tests are given to demonstrate the simplicity, efficiency and robustness of the proposed scheme.

✉ Gwanghyun Jo
gwanghyun@kunsan.ac.kr

Yoonjeong Choi
yjchoi7166@gmail.com

Do Y. Kwak
kdy@kaist.ac.kr

Young Ju Lee
yjlee@txstate.edu

¹ Urbanbase Inc, Seocho-gu, Seoul, Republic of Korea

² Department of Mathematics, Kunsan National University, Runsan-si, Republic of Korea

³ Department of Mathematical Sciences, KAIST, Daejeon, Republic of Korea

⁴ Department of Mathematics, Texas State University, San Marcos, TX, USA

Keywords Discontinuous bubble schemes · Immersed finite element method · Elliptic equation with interface · Hele-Shaw flows

Mathematics Subject Classification (2010) 65M60 · 65N30

1 Introduction

Hele-Shaw equation can be formulated as the Darcy flow coupled with the transport that describes the interface motion. It describes flows between two parallel flat plates with small gap [14, 16], where there is an interface determined by two different fluids. The challenges in the solution of Hele-Shaw equation lie in two factors: (1) due to the different properties of the fluid in the domain, the pressure is discontinuous across the interface, which is proportional to the curvature of the interface. Typically, the curvature of the interface affects the pressures and velocity, and the interface goes through big and abrupt changes in time evolution; (2) the interface motion occurs along with the Darcy velocity and it has to be carefully tracked to arrive at a reliable solution.

In this paper, we shall employ the level set equation to capture the interface motion and solve it using the technique of the weighted essentially non-oscillatory (WENO) scheme [19, 20, 41]. WENO scheme is a powerful method for solving (essentially) hyperbolic type partial differential equations (PDEs). Our observation is that the level set transport equation coupled with Darcy velocity and the Darcy velocity should be conservative to obtain robust interface tracking. Therefore, we employ the locally conservative enriched Galerkin finite element methods to solve the Darcy law.

The importance of the conservative flux has been documented in a number of literature (see [31, 43] and references cited therein) and there are a number of research performed for this purpose. For fitted grids to the interface or the mesh resolving interface, for example, mixed finite element methods (MFEMs) are most frequently used to approximate the velocity [7, 42]. Hybrid formulation of MFEMs using the Lagrangian multipliers can be found at Arnold et al. [1] for example. Another class of conservative schemes can be constructed by the discontinuous Galerkin (DG) community. However, typically DG introduces a number of degrees of freedom and the system is difficult to solve. A similar, but much efficient approach we use in this paper is the enriched Galerkin (EG) methodology [31, 43], which is constructed by enriching the conforming finite element spaces with piecewise constants. There are other types of conservative schemes based on conforming Galerkin finite elements. However, extra works have to be applied to obtain a conservative flux [3, 15]. For moving interface problem such as Hele-Shaw flow, these methods may not be adequate and it is much desirable to apply or devise the conservative methods for the unfitted grid. More precisely, handling the interface in an unfitted grid provides a number of advantages, among others, a structured grid can be used for the discretization, resulting in simple data structures. For unfitted grids, we note that finite difference method (FDM) type method, immersed interface method (IIM) was introduced by [33] and used effectively in [16, 30, 34, 35, 45]. In particular, IIM type methods were used effectively to simulate Hele-Shaw flows in [16]. Meanwhile

immersed hybrid difference methods were introduced in [17, 18]. Also, extended finite element methods (XFEMs) [4, 5, 24, 32, 40] use extra degrees of freedom (DOFs) obtained by truncating the shape function along the interface. Hence, XFEM requires more DOFs. On the other hand, immersed finite element method was (IFEM) introduced in [36, 37]. The concept of IFEM modifies the basis functions across the interface so that they satisfy the local flux continuity. One of the differences of IFEM from the XFEM is that the former does not require extra degrees of freedom. The convergence analysis was investigated in [10, 25, 26, 39] and many applications were obtained effectively, for example, in flows through the porous media [21], elasticity equation [26, 29], Poisson Boltzmann [28].

In this paper, we propose a new method based on the enriched immersed finite element method (EIFEM) recently published in [22] to tackle the Hele-Shaw flow. The novelty of the current work is the modification of EIFEM so that it can handle the abrupt change in the pressure jump along with the interface, still resulting in a locally conservative flux while keeping the numerical efficiency. To handle the pressure jump condition, we modify the discontinuous bubble IFEM (DB-IFEM) [9] framework. It is a powerful method to handle the nonhomogeneous jump conditions along the interface. Therefore, we obtain a new methodology to solve the elliptic equation with nonhomogeneous jumps by combining the EIFEM and DB-IFEM introduced in [9]. The use of discontinuous bubble function provides a framework in which the fast solver based on algebraic multigrid method can still be successfully applied just like the case of the homogeneous jump condition as discussed in [22]. The proposed scheme in this paper will be called the discontinuous bubble-enriched immersed finite element method (DB-EIFEM). To the best knowledge of authors, the proposed scheme is the first structured grids based $H(\text{div})$ flux recovery for elliptic interface problems with nonhomogeneous jump condition for the pressure. This technique is then combined with WENO scheme for solving the level set formalism to track the interface. WENO, the weighted essentially non-oscillatory (WENO) scheme [19, 20, 41] is a powerful method for solving hyperbolic type partial differential equations (PDEs).

The rest of the paper is organized as follows. In Section 2, the governing equation and the temporal discretization of Hele-Shaw equation are described. We describe the DB-EIFEM for elliptic interface problems with nonhomogeneous jump condition in Section 3. WENO scheme for level set equation is then described in Section 4. Section 5 presents a number of numerical experiments to demonstrate the performance of proposed algorithms. Finally, we conclude the paper with concluding remarks in Section 6.

2 Governing equation and its temporal discretization

We consider Hele-Shaw flows governed by Darcy laws where two different fluids meet along some interface. We shall assume that one of the fluids is injected at the

center of the region Ω in \mathbb{R}^2 whose underlying subregion is $\Omega^-(t)$. The other fluid is located on $\Omega^+(t) = \Omega / \Omega^-(t)$ and the interface is $\Gamma(t) = \partial\Omega^-(t)$. The pressure p and velocity field \mathbf{u} obey the governing equation:

$$\operatorname{div} \mathbf{u} = f, \quad \text{in } \Omega^+(t) \cup \Omega^-(t), \quad \text{and} \tag{1a}$$

$$\mathbf{u} = -\beta \nabla p, \quad \text{in } \Omega^+(t) \cup \Omega^-(t), \tag{1b}$$

subject to the interface, initial and boundary conditions:

$$[[p]]_{\Gamma(t)} = \tau \kappa(t), \quad [[\beta \nabla p \cdot \mathbf{n}]]_{\Gamma(t)} = 0, \quad p|_{t=0} = p^0, \quad \text{and} \quad p|_{\partial\Omega} = g, \tag{2}$$

where the coefficients $\beta^\pm := \beta|_{\Omega^\pm(t)}$ are positive constants and \mathbf{n} is a unit normal vector to $\Gamma(t)$, outward to $\partial\Omega^-$ so $\mathbf{n} = \mathbf{n}^-$ indeed, and bracket $[[\cdot]]_{\Gamma}$ means the jump, i.e., $[[u]]_{\Gamma} = u|_{\Omega^-} - u|_{\Omega^+}$. We assume that the source term f is zero on Ω^+ . Here, τ is the surface tension and κ is the curvature of the interface. We note that amalgamated surface tension d_0 satisfies

$$d_0 = \frac{2\tau\pi\beta^+}{\text{IR}},$$

where IR is the injection rate. Physically, this indicates that for the high injection rate, surface tension increases, leading to the high jump of pressure across the interface. Also, we observe that the curvature affects the flow since the jump is proportional to the curvature. The motion of the interface $\Gamma(t)$ at any given time level t is described by some level set of $\Phi(\mathbf{x}, t)$, where $\Phi(\mathbf{x}, t)$ satisfies

$$\frac{\partial \Phi}{\partial t} + \mathbf{u} \cdot \nabla \Phi = 0, \quad \text{in } \Omega, \quad \text{with} \quad \Gamma(0) = \{\Phi(\mathbf{x}, t = 0) = 0\}, \tag{3}$$

for which the zero-level set of Φ is set to define the interface, i.e., $\Gamma(t) = \{\Phi(\mathbf{x}, t) = 0\}$. Therefore, the governing equation for the Hele-Shaw flow we are concerned with in this paper, consists of total three equations with initial, interface, and boundary conditions, i.e., (1a), (1b), (2) and (3).

We now discuss how to obtain the approximate evolution of solutions to the Hele-Shaw equation. Given a time domain, $[0, T_f]$, where T_f is the final time level, we divide the time interval, uniformly with the temporal mesh size Δt , which will be specified in terms of the spatial mesh size h , so that $\Delta t = O(h^2)$. The break points of time discretization are denoted by $t^n = n\Delta t$. For the sake of simplicity, we let p^n , \mathbf{u}^n , Φ^n , κ^n and Γ^n to denote temporal approximations of $p(t^n)$, $\mathbf{u}(t^n)$, $\Phi(t_n)$, $\kappa(t_n)$, and $\Gamma(t^n)$, respectively for any given $n \geq 0$. The basic algorithm will be based on a semi-implicit scheme. More precisely, we shall perform the following time evolution algorithm:

Given $(p^n, \mathbf{u}^n, \Phi^n, \kappa^n, \Gamma^n)$, we obtain $(p^{n+1}, \mathbf{u}^{n+1}, \Phi^{n+1}, \kappa^{n+1}, \Gamma^{n+1})$ by performing the following procedures:

Step 1: Solve for p^{n+1} and \mathbf{u}^{n+1} :

$$\begin{aligned} \operatorname{div} \mathbf{u}^{n+1} &= f, & \text{in } \Omega^+(t_n) \cup \Omega^-(t_n), \\ \mathbf{u}^{n+1} &= -\beta \nabla p^{n+1}, & \text{in } \Omega^+(t_n) \cup \Omega^-(t_n), \end{aligned}$$

subject to the interface condition on Γ^n and boundary condition:

$$\llbracket p^{n+1} \rrbracket_{\Gamma^n} = \tau \kappa^n, \quad \llbracket \beta \nabla p^{n+1} \cdot \mathbf{n} \rrbracket_{\Gamma^n} = 0 \quad \text{and} \quad p^{n+1}|_{\partial \Omega} = g,$$

Step 2: Solve for Φ^{n+1} :

$$\frac{\Phi^{n+1} - \Phi^n}{\Delta t} + \mathbf{u}^{n+1} \cdot \nabla \Phi^n = 0 \quad \text{in } \Omega.$$

Step 3: Update the interface location $\Gamma(t^{n+1})$ as a zero-level set of Φ^{n+1} as well as the curvature κ^{n+1} of $\Gamma(t^{n+1})$ and go to Step 1 if $t_{n+1} < T_f$.

3 Discontinuous bubble-enriched immersed finite element method (DB-EIFEM)

The purpose of this section is to discuss how to obtain the spatial approximations, i.e., p_h^{n+1} and \mathbf{u}_h^{n+1} for given κ_h^n and Γ_h^n . For the sake of simplicity, we shall omit the superscript $n+1$ and n . Namely, we let $p := p^{n+1}$, $\mathbf{u} := \mathbf{u}^{n+1}$, $\kappa := \kappa^n$ and $\Gamma = \Gamma^n$. We also denote $\Omega^\pm := \Omega^\pm(t_n)$. From now on, we impose homogeneous boundary condition for the pressure variable for the ease of presentation, i.e., we set $g = 0$ in (2).

We shall assume that the domain Ω is of a rectangular shape. Note that this assumption is given so that one can use the structured grid to discretize the domain. If Ω is not of the rectangular shape, then it is subtle, but one can consider the domain of rectangular shape that contains Ω and then formulate the Hele-Shaw problem in an extended domain, which is the future research. Let \mathcal{T}_h be a triangulation of Ω by the squares, whose mesh size h and \mathcal{E}_h be the set of all edges. Here, $\mathcal{E}_h = \mathcal{E}_h^o \cup \mathcal{E}_h^\partial$, where \mathcal{E}_h^o is the set of interior edges while \mathcal{E}_h^∂ is the set of boundary edges. Note that there are triangles $T \in \mathcal{T}_h$ that contain the part of interface and such T will be called an interface element. In fact, since the interface is moving, the interface elements can be different with time evolution. When $T \in \mathcal{T}_h$ does not contain any part of the interface, such T will be called a noninterface element.

We begin our discussion with some notation of function spaces and norms. Note that we shall use standard Sobolev space notation. For any bounded subdomain $D \subset \Omega$, we denote $D^+ = D \cap \Omega^+$, $D^- = D \cap \Omega^-$. We shall use standard Sobolev space, i.e., $H^m(D)$ and $H_0^1(D)$ denote the ordinary Sobolev spaces of order m with the norm $\|\cdot\|_{m,D}$ and the semi-norm $|\cdot|_{m,D}$. For $m = 1, 2$, the broken Sobolev space $\tilde{H}^m(D)$ is defined as follows:

$$\tilde{H}^m(D) := H^m(D^+) \cap H^m(D^-),$$

with norms (semi-norms)

$$\|u\|_{\tilde{H}^m(D)}^2 := \|u\|_{H^m(D^+)}^2 + \|u\|_{H^m(D^-)}^2, \quad \forall u \in \tilde{H}^m(D).$$

The space $\tilde{H}_0^1(D)$ is defined by the following:

$$\tilde{H}_0^1(D) := \{u \in \tilde{H}^1(D) \mid u = 0 \text{ on } \partial D\}.$$

Step 1 is to solve the (1a) and (1b). We shall solve this equation by simply considering a pressure formulation given as follows:

$$-\operatorname{div} \beta \nabla p = f \quad \text{in } \Omega^+ \cup \Omega^-, \tag{4}$$

subject to the interface and boundary conditions:

$$[[p]]_\Gamma = \tau \kappa, \quad [[\beta \nabla p \cdot \mathbf{n}]]_\Gamma = 0, \quad \text{and} \quad p|_{\partial \Omega} = 0, \tag{5}$$

The weak formulation can be given as follows [9]: find $p \in \tilde{H}_0^1(\Omega)$ such that $[[p]]_\Gamma = \tau \kappa$ and

$$a(p, q) = (f, q), \quad \forall q \in H_0^1(\Omega), \tag{6}$$

where

$$a(p, q) := \int_{\Omega} \beta \nabla p \cdot \nabla q \, dx \quad \text{and} \quad (f, q) := \int_{\Omega} f q \, dx. \tag{7}$$

When it comes to the spatial discretization of the above equation, the issue arises in how to incorporate the nonhomogeneous jump condition, i.e., $[[p]]_\Gamma = \tau \kappa$. We propose to use the discontinuous bubble function as discussed below (see also [9]).

3.1 Discontinuous bubble-immersed finite element method (DB-IFEM)

In this section, we describe the discontinuous bubble-immersed finite element method, first introduced in [9]. The main idea is to introduce a special function $p^* \in \tilde{H}_0^1(\Omega)$ such that $[[p^*]]_\Gamma = \tau \kappa$ and to consider the following:

$$p = \hat{p} + p^*.$$

Note that if $\tau \kappa = 0$, then we can simply set $p^* = 0$ and in case $\tau \kappa \neq 0$, there are infinitely many ways to choose such p^* . To specify the particular choice of p^* , we let S_Γ be a thin tube containing the interface Γ in its interior as shown in Fig. 1. We then construct p^* so that it satisfies the jump condition $[[p^*]]_\Gamma = \tau \kappa$ and p^* is supported within S_Γ .

Such chosen p^* will be called a discontinuous bubble function since it is locally supported near the interface. Once p^* is constructed, by subtracting p^* from p in the (6), we obtain a new weak problem for \hat{p} with the homogeneous jump condition. Namely, find $\hat{p} \in \tilde{H}_0^1(\Omega)$ such that

$$a(\hat{p}, q) = (f, q) - a(p^*, q), \quad \forall q \in H_0^1(\Omega). \tag{8}$$

In the remainder of this section, we discuss how to incorporate the discrete p_h^* as well as the discrete version of the (8). We consider the Q_1 -conforming immersed finite element method introduced in [38]. Let $S_h(T)$ be a space of standard bilinear functions on T with degree of freedoms given at nodes. Note that this local space needs to be modified when T is an interface element. More precisely, let the element

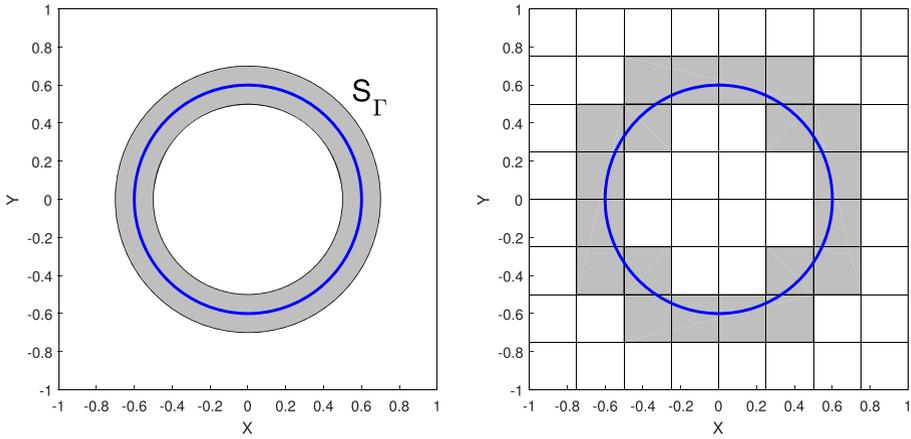


Fig. 1 Example of thin strip S_Γ (left) and discretized strip S_Γ^h (right) for the case of circular interface (blue curve) with radius $r = 0.6$

T be cut by the interface Γ at the two edges e_1 and e_3 at points E_1 and E_2 . This divides T into two parts, denoted by $T^+ = T \cap \Omega^+$ and $T^- = T \cap \Omega^-$ as shown in Fig. 2.

The basis function $\lambda \in S_h(T)$ should be modified into a piecewise linear $\hat{\lambda}$ of the following form:

$$\hat{\lambda}(x, y) := \begin{cases} \hat{\lambda}^+(x, y) = a^+ + b^+x + c^+y + d^+xy, & (x, y) \in T^+, \\ \hat{\lambda}^-(x, y) = a^- + b^-x + c^-y + d^-xy, & (x, y) \in T^-, \end{cases} \quad (9)$$

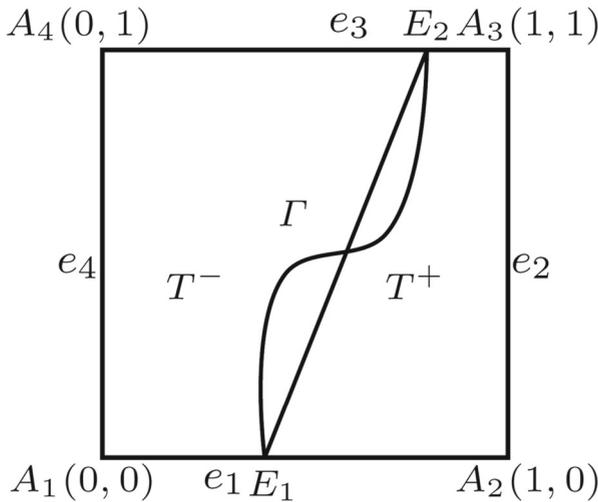


Fig. 2 An interface element T cut by interface Γ

where we impose $d^+ = d^-$ and other coefficients are determined by vertex degree of freedoms and the homogeneous jump conditions, namely,

$$\widehat{\lambda}(A_i) = \lambda(A_i), \quad i = 1, 2, 3, 4, \tag{10a}$$

$$\widehat{\lambda}^+(E_i) = \widehat{\lambda}^-(E_i), \quad i = 1, 2, \tag{10b}$$

$$\int_{\overline{E_1E_2}} \beta^+ \nabla \widehat{\lambda}^+ \cdot \mathbf{n}_{\overline{E_1E_2}} = \int_{\overline{E_1E_2}} \beta^- \nabla \widehat{\lambda}^- \cdot \mathbf{n}_{\overline{E_1E_2}}. \tag{10c}$$

The reason why we set $d^+ = d^-$ is that the conditions (10a)–(10c) then uniquely determine the local basis function $\widehat{\lambda}$ [38]. We shall denote by $\widehat{S}_h(T)$ the space of modified functions that satisfy the homogeneous jump conditions. By gluing the modified local spaces, we obtain the immersed finite element method based on Q_1 -conforming space $\widehat{S}_h(\Omega)$, which is defined as follows: for $\phi \in \widehat{S}_h(\Omega)$, it holds that

$$\begin{cases} \phi|_T \in S_h(T) & \text{if } T \text{ is a noninterface element,} \\ \phi|_T \in \widehat{S}_h(T) & \text{if } T \text{ is an interface element,} \\ \phi|_{T_1}(X) = \phi|_{T_2}(X) & \text{if } T_1 \text{ and } T_2 \text{ are adjacent elements} \\ & \text{and } X \text{ is a common node of } T_1 \text{ and } T_2, \\ \phi(X) = 0 & \text{if } X \text{ is a node on the boundary edges.} \end{cases}$$

We are now in a position to construct a discontinuous bubble function p_* in a discretized domain. Basically, we shall simply define p_h^* as a piecewise bilinear function in each element, so that $(p_h^*)|_T = 0$ for any non interface element, $T \in \mathcal{T}_h$ and if T is an interface element, we define $(p_h^*)|_T$ so that it satisfies the following nonhomogeneous jump conditions:

$$\begin{aligned} p_h^*(A_i) &= 0, \quad i = 1, 2, 3, 4 \\ \llbracket p_h^* \rrbracket_\Gamma(E_i) &= \tau \kappa(E_i), \quad i = 1, 2, \\ \int_{\overline{E_1E_2}} \beta^+ \nabla p_h^*|_{T^+} \cdot \mathbf{n}_{\overline{E_1E_2}} &= \int_{\overline{E_1E_2}} \beta^- \nabla p_h^*|_{T^-} \cdot \mathbf{n}_{\overline{E_1E_2}}. \end{aligned}$$

It is remarkable that such discontinuous bubble function p_h^* can be obtained uniquely [9]. With the introduction of p_h^* , we then arrive at the following discontinuous bubble-immersed finite element method (DB-IFEM) as follows: Find $\widehat{p}_h \in \widehat{S}_h(\Omega)$ such that

$$a(\widehat{p}_h, q_h) = (f, q_h) - a(p_h^*, q_h), \quad \forall q_h \in \widehat{S}_h(\Omega). \tag{12}$$

3.2 Discontinuous bubble-enriched immersed finite element method (DB-EIFEM)

We would like to remark that DB-IFEM is shown to perform successfully for handling nonhomogeneous jump conditions [9]. On the other hand, it does not lead to the locally conservative flux and therefore, it is numerically shown that for Hele-Shaw flow of our interest, this method suffers from the difficulty in the long term computation. Namely, the interface breaks down in the convergence in the time evolution.

We begin this section by recalling the local and global conservation of the flux variable.

Definition 1 (Local conservation) Given a triangulation \mathcal{T}_h , we say that the discrete flux \mathbf{u}_h is conservative if the following holds true:

$$\int_{\partial T} \mathbf{u}_h \cdot \mathbf{n} \, ds = \int_T f \, dx, \quad \forall T \in \mathcal{T}_h, \tag{13}$$

where \mathbf{n} is the unit outward normal vector to ∂T . The corresponding global conservation is with T replaced by Ω in the equation,

We would like to remark that the conservation is dependent on the choice of triangulations, \mathcal{T}_h . The issue with the conservation has drawn a lot of attention in literatures (see [31, 43] and references cited therein). Due to the absence of the piecewise constant in the DB-IFEM, the standard and its variant IFEMs are not locally or globally conservative. We shall discuss a simple method to remedy the issue arising in DB-IFEM, leading the locally conservative flux variable. Basically, we shall consider to enrich the space $\widehat{S}_h(\Omega)$ by adding the locally defined piecewise constant functions. This concept has been introduced in prior works [31, 43]. However, it is the first time in this work to combine it with the discontinuous bubble functions to handle the nonhomogeneous jump conditions. We begin our discussion by enriching the pressure space by piecewise constants, namely, we define the enriched immersed finite element (EIFEM) space as follows:

$$E_h(\Omega) := \widehat{S}_h(\Omega) + M_h^0(\Omega), \tag{14}$$

where $M_h^0(\Omega)$ is piecewise constant functions defined on \mathcal{T}_h . Note that the dimension of the space $\widehat{S}_h(\Omega)$ is the number of interior nodes for the triangulation \mathcal{T}_h , which will be denoted by N_0 while the dimension of the space M_h^0 is the number of elements, denoted by N_E . For each $e \in \mathcal{E}_h$, we associate a unit vector \mathbf{n}_e at e . We define the jump $\llbracket \phi \rrbracket_e$ and average $\{\{\phi\}\}_e$ for $\phi \in H^1(\mathcal{T}_h)$ as follows:

$$\begin{aligned} \llbracket \phi \rrbracket_e(\mathbf{x}) &:= \lim_{\delta \rightarrow 0^+} (\phi(\mathbf{x} - \delta \mathbf{n}_e) - \phi(x + \delta \mathbf{n}_e)), \\ \{\{\phi\}\}_e(\mathbf{x}) &:= \frac{1}{2} \lim_{\delta \rightarrow 0^+} (\phi(\mathbf{x} - \delta \mathbf{n}_e) + \phi(x + \delta \mathbf{n}_e)), \end{aligned}$$

if e does not belong to $\partial\Omega$ and

$$\llbracket \phi \rrbracket_e(\mathbf{x}) := \lim_{\delta \rightarrow 0^+} (\phi(\mathbf{x} - \delta \mathbf{n}_{\partial\Omega})), \quad \{\{\phi\}\}_e(\mathbf{x}) := \lim_{\delta \rightarrow 0^+} (\phi(\mathbf{x} - \delta \mathbf{n}_{\partial\Omega}))$$

if e belongs to $\partial\Omega$.

We first, consider the space $H_h(\Omega) := \widetilde{H}^1(\Omega) + E_h(\Omega)$ and equip it with a broken H^1 -norm:

$$\|\phi\|_h := \sum_{T \in \mathcal{T}_h} \|\beta^{\frac{1}{2}} \nabla \phi\|_{0,T}^2 + \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \int_e \llbracket \phi \rrbracket_e^2 \, ds, \quad \forall \phi \in H_h(\Omega). \tag{15}$$

We then define the bilinear form $a_h(\cdot, \cdot) : H_h(\Omega) \times H_h(\Omega) \mapsto \mathbb{R}$ by, for all $v, w \in H_h(\Omega)$,

$$a_h(v, w) = \sum_{T \in \mathcal{T}_h} \int_T \beta \nabla v \cdot \nabla w \, dx - \sum_{e \in \mathcal{E}_h} \int_e \{\{\beta \nabla v \cdot \mathbf{n}_e\}\}_e \llbracket w \rrbracket_e \, ds + \mu \sum_{e \in \mathcal{E}_h} \int_e \{\{\beta \nabla w \cdot \mathbf{n}_e\}\}_e \llbracket v \rrbracket_e \, ds + \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \int_e \sigma(\beta) \llbracket v \rrbracket_e \llbracket w \rrbracket_e \, ds,$$

where $|e|$ is the measure of e , the symbol μ will be discussed below, and the symbol $\sigma(\beta)$ is to indicate that σ , the stabilization parameter is chosen depending on β in each edge $e \in \mathcal{E}_h$. Theoretically, at the interface edge, $\sigma = \zeta \bar{\beta} / \underline{\beta}$ for some $\zeta > 0$ gives sufficient stabilization.

We are now in a position to state the DB-EIFEM to solve (4) subject to the condition (5). DB-EIFEM can then be formulated as follows: find $\widehat{p}_h \in E_h$ such that

$$a_h(\widehat{p}_h, w_h) = (f, w_h) - a_h(p_*, w_h), \quad \forall w_h \in E_h, \tag{16}$$

where p_* is the discontinuous bubble function constructed in Section 3.1. The bilinear form, here is the one that corresponds to interior penalty DG method, introduced in [44]. The symbol μ is the tuning parameter, which determines the type of interior penalty method, i.e., $\mu = -1, 0$ and 1 result in SIPG, IIPG and NIPG, respectively [31, 43]. In this work, we take $\mu = -1$ to have a symmetric algebraic system. The reason for taking this parameter is that we can develop a fast solver for the symmetric positive definite system (see Section 3.2.1). We can also establish its efficiency and numerical error analysis for IIPG and NIPG. We believe their efficiency are similar to SIPG. We briefly remarked why the fast solver should perform equally well for IIPG below (see Section 3.2.1). Once we solve the aforementioned (16), we can set

$$p_h := \widehat{p}_h + p_h^*. \tag{17}$$

We state the following consistency result:

Lemma 1 *Suppose p is the solution of (4) subject to the condition (5) and p_h is defined as in (17). Then, we have*

$$a_h(p - p_h, w_h) = 0, \quad \forall w_h \in E_h. \tag{18}$$

Proof This follows from the definition of the $a_h(\cdot, \cdot)$ form. □

Now we prove the ellipticity for the completeness even though it is basically the same as [22]. We need the following lemma [22].

Lemma 2 *For all $\phi \in E_h(\Omega)$ and $T \in \mathcal{T}_h$ and edges e of T , the following trace-like inequality holds.*

$$\|\beta \nabla \phi \cdot \mathbf{n}_e\|_{0,e}^2 \leq C_t h^{-1} \|\beta \nabla \phi\|_{0,T}^2,$$

where the constant C_t is independent of both h and the location of interface.

Now, we state a lemma regarding the coerciveness of the bilinear form $a_h(\cdot, \cdot)$.

Lemma 3 *There exists some $\sigma_0 > 0$ such that the following holds whenever $\sigma(\beta) > \sigma_0$,*

$$C_\alpha \|\phi_h\|_h^2 \leq a_h(\phi_h, \phi_h), \quad \forall \phi_h \in E_h(\Omega), \tag{19}$$

for some $C_\alpha > 0$.

Proof Let ϕ_h be an arbitrary function in $E_h(\Omega)$. First, the following comes from the Cauchy’s inequality:

$$\begin{aligned} & \sum_{e \in \mathcal{E}_h} \int_e |\{\{\beta \nabla \phi_h \cdot \mathbf{n}_e\}\} \llbracket \phi_h \rrbracket| ds \\ & \leq \left(h \sum_{e \in \mathcal{E}_h} \|\{\{\beta \nabla \phi_h \cdot \mathbf{n}_e\}\}\|_{0,e}^2 \right)^{\frac{1}{2}} \left(h^{-1} \sum_{e \in \mathcal{E}_h} \|\llbracket \phi_h \rrbracket\|_{0,e}^2 \right)^{\frac{1}{2}}. \end{aligned} \tag{20}$$

Let T_e^+ and T_e^- be two neighboring elements of the edge e . By applying the Lemma 2 and using the fact each edge is included at most two elements, we have that

$$\begin{aligned} h \sum_{e \in \mathcal{E}_h} \|\{\{\beta \nabla \phi_h \cdot \mathbf{n}_e\}\}\|_{0,e}^2 & \leq h \sum_{e \in \mathcal{E}_h} \left(\|(\beta \nabla \phi_h)|_{T_e^+} \cdot \mathbf{n}_e\|_{0,e}^2 + \|(\beta \nabla \phi_h)|_{T_e^-} \cdot \mathbf{n}_e\|_{0,e}^2 \right) \\ & \leq C_t \sum_{e \in \mathcal{E}_h} \|\beta \nabla \phi_h\|_{0,T_e^+ \cup T_e^-}^2 \\ & \leq 2C_t \bar{\beta} \sum_{T \in \mathcal{T}_h} \|\beta^{\frac{1}{2}} \nabla \phi_h\|_{0,T}^2. \end{aligned} \tag{21}$$

Invoking Young’s inequality and using (20) and (21), we have that

$$\begin{aligned} & \sum_{e \in \mathcal{E}_h} \int_e |\{\{\beta \nabla \phi_h \cdot \mathbf{n}_e\}\} \llbracket \phi_h \rrbracket| ds \\ & \leq \frac{\delta}{2} \sum_{T \in \mathcal{T}_h} \|\beta^{\frac{1}{2}} \nabla \phi_h\|_{0,T}^2 + \frac{2C_t \bar{\beta}}{2\delta} \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \|\llbracket \phi_h \rrbracket\|_{0,e}^2, \end{aligned} \tag{22}$$

for every $\delta > 0$. By the definition of $a_h(\cdot, \cdot)$ and by the inequalities (22), we have

$$\begin{aligned} a_h(\phi_h, \phi_h) & = \sum_{T \in \mathcal{T}_h} \int_T \beta \nabla \phi_h \cdot \nabla \phi_h dx \\ & \quad - 2 \sum_{e \in \mathcal{E}_h} \int_e \{\{\beta \nabla \phi_h \cdot \mathbf{n}_e\}\}_e \llbracket \phi_h \rrbracket_e ds + \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \int_e \sigma(\beta) \llbracket \phi_h \rrbracket_e^2 ds \\ & \geq (1 - \delta) \sum_{T \in \mathcal{T}_h} \|\beta^{\frac{1}{2}} \nabla \phi_h\|_{0,T}^2 + \left(\min_{e \in \mathcal{E}_h} \sigma(\beta) - \frac{2C_t \bar{\beta}}{\delta} \right) \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \|\llbracket \phi_h \rrbracket\|_{0,e}^2. \end{aligned}$$

Here, we let $\delta = 1/2$. Choosing σ_0 large enough such that

$$\sigma_0 > 4C_t \bar{\beta},$$

we obtain the desired result when $\sigma(\beta) > \sigma_0$ with

$$C_\alpha = \min\left(\frac{1}{2}, \sigma_0 - 4C_t\bar{\beta}\right).$$

This completes the proof. □

The following optimal convergence result for SIPG version of DB-EIFEM can be established:

Theorem 1 *Suppose p is the solution of (4) subject to the condition (5) and p_h is defined as in (17). The following error estimate holds true :*

$$\|p - p_h\|_{0,\Omega} + h\|p - p_h\|_h \leq Ch^2\|f\|_{0,\Omega}. \tag{23}$$

Proof When $\llbracket p \rrbracket_\Gamma = 0$, it can be established as in [22]. For the nonhomogeneous case, the result can be shown similarly to [27]. This completes the proof. □

3.2.1 Fast solver for EIFEM based on algebraic multigrid method

In this section, we give a description of preconditioning techniques based on fictitious or auxiliary spaces as discussed in [31]. We let

$$\widehat{S}_h(\mathcal{T}_h) = \text{span}\{\phi_j\}_{j=1}^{N_0} \quad \text{and} \quad M_h^0(\mathcal{T}_h) = \text{span}\{\psi_j\}_{j=1}^{N_e},$$

where ϕ_j is a nodal basis for the IFEM and ψ_j is the element-wise constant function defined by $\psi_j|_{T_\ell} = \delta_{j\ell}$. Here, N_0 is the number of total nodes in \mathcal{T}_h , which excludes the Dirichlet nodes and N_e is the number of elements in \mathcal{T}_h . As discussed in [22], we have imposed the strong Dirichlet boundary condition for the piecewise linear finite element while the weakly imposed zero boundary condition is given for piecewise enrichment to remove the zero eigen-mode for the resulting system. The system arising from (12) is written in N by N ($N = N_0 + N_e$) system

$$\underset{\approx}{A} p \underset{\approx}{=} f, \tag{24}$$

where the matrix can be written as

$$\underset{\approx}{A} = \begin{pmatrix} \underset{\approx}{A} & \underset{\approx}{A} \\ \underset{\approx}{A}_{11} & \underset{\approx}{A}_{12} \\ \underset{\approx}{A} & \underset{\approx}{A} \\ \underset{\approx}{A}_{21} & \underset{\approx}{A}_{22} \end{pmatrix}.$$

The right-hand side of (24) is $(f, \phi_i) - a_h(p_h^*, \phi_i)$ for the first N_0 -entries and $(f, \psi_i) - a_h(p_h^*, \psi_i)$ for the next N_e -entries. We shall now describe the block matrices denoted by $\underset{\approx}{A}_{11}$, $\underset{\approx}{A}_{12}$, $\underset{\approx}{A}_{21}$ and $\underset{\approx}{A}_{22}$. We recall that the bilinear form is given as follows (with $\mu = -1$):

$$\begin{aligned} a_h(v, w) &= \sum_{T \in \mathcal{T}_h} \int_T \beta \nabla v \cdot \nabla w dx - \sum_{e \in \mathcal{E}_h} \int_e \{\{\beta \nabla v \cdot \mathbf{n}_e\}\}_e \llbracket w \rrbracket_e ds \\ &\quad - \sum_{e \in \mathcal{E}_h} \int_e \{\{\beta \nabla w \cdot \mathbf{n}_e\}\}_e \llbracket v \rrbracket_e ds + \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \int_e \sigma(\beta) \llbracket v \rrbracket_e \llbracket w \rrbracket_e ds. \end{aligned}$$

The first block is obtained by setting $v = \phi_j$ and $w = \phi_i$, both of which are from $\widehat{S}_h(\mathcal{T}_h)$. Noticing that $v|_e = w|_e = 0$ for $e \in \mathcal{E}_h^\partial$, we have that

$$A_{\approx 11}(i, j) = a_h(\phi_j, \phi_i) = \sum_{T \in \mathcal{T}_h} \int_T \beta \nabla \phi_j \cdot \nabla \phi_i dx - \sum_{e \in \mathcal{E}_h^o} \int_e \{\{\beta \nabla \phi_j \cdot \mathbf{n}_e\}\}_e \llbracket \phi_i \rrbracket_e ds$$

$$- \sum_{e \in \mathcal{E}_h^o} \int_e \{\{\beta \nabla \phi_i \cdot \mathbf{n}_e\}\}_e \llbracket \phi_j \rrbracket_e ds + \sum_{e \in \mathcal{E}_h^o} \frac{1}{|e|} \int_e \sigma(\beta) \llbracket \phi_j \rrbracket_e \llbracket \phi_i \rrbracket_e ds.$$

We now consider the submatrix $A_{\approx 12}$. This is obtained by setting $v = \psi_j$ with $\psi_j \in M_h^0(\mathcal{T}_h)$ while $w = \phi_i$ with $\phi_i \in \widehat{S}_h(\mathcal{T}_h)$. Again, considering the strongly imposed boundary condition for ϕ_i , we have that

$$A_{\approx 12}(i, j) = - \sum_{e \in \mathcal{E}_h^o} \int_e \{\{\beta \nabla \phi_i \cdot \mathbf{n}_e\}\}_e \llbracket \psi_j \rrbracket_e ds + \sum_{e \in \mathcal{E}_h^o} \frac{1}{|e|} \int_e \sigma(\beta) \llbracket \psi_j \rrbracket_e \llbracket \phi_i \rrbracket_e ds.$$

Similarly $A_{\approx 21}$ is obtained by setting $v = \phi_j$ and $w = \psi_i$. We see that $A_{\approx 21}(i, j) = A_{\approx 12}(j, i)$. Finally, for $A_{\approx 22}$, we set $v = \psi_j$ and $w = \psi_i$ with $\psi_j, \psi_i \in M_h^0(\mathcal{T}_h)$. We then arrive at the following:

$$A_{\approx 22}(i, j) = \sum_{e \in \mathcal{E}_h} \frac{1}{|e|} \int_e \sigma(\beta) \llbracket \psi_j \rrbracket_e \llbracket \psi_i \rrbracket_e ds.$$

We consider the auxiliary space preconditioner which consists of the following three steps: step i) pre-smoothing step ii) solving each diagonal block system, $A_{\approx 11}$ and $A_{\approx 22}$, which are the restrictions of A to $\widehat{S}_h(\mathcal{T}_h)$ and $M_h^0(\mathcal{T}_h)$, respectively, and step iii) post-smoothing (for symmetrization).

We apply the following in each iteration:

1. Gauss Seidel N_{GS} .
2. Compute Residual. $R = R_1 + R_2$.
3. Precondition for each submatrix S_{11} and S_{22} : $z_1 = AMG(S_{11})R_1$ and $z_2 = AMG(S_{22})R_2$.
4. Update corrections: $x = (x_1 + z_1) + (x_2 + z_2)$.
5. Backward Gauss-Seidel N_{GS} .

We shall state the main result for the proposed solver without proof since it is established in [22]. Numerical test results confirming the theory are presented in Section 5.

Theorem 2 [22] *The algorithm produces preconditioner that works independent of mesh size and jumps whenever the block preconditioner is chosen to work independent of mesh size and jumps.*

We note that the block matrix $A \approx_{22}$ is M-matrix and weakly diagonally dominant and therefore, it can be easily solved by a classical algebraic multigrid method [22, 31]. We remark that our solver is based on the block diagonal preconditioning. Even if it is not the scope of this paper, especially for IIPG, the off-diagonal block $A \approx_{12}$ is almost negligible, which means that the system is almost tridiagonal system. The tridiagonal system can be easily handled if one can handle $A \approx_{11}$ and $A \approx_{22}$. Heuristically, this observation can explain the effectiveness of our solver for IIPG system.

3.2.2 Conservative flux reconstruction and its error analysis

In this section, we discuss the flux reconstruction. Basically, we shall apply $H(\text{div})$ projection of the numerical flux to lowest order Raviart-Thomas (RT) space [42] and this will be shown to be conservative. Let $V_h(T)$ be the local RT space on an element T and $V_h(\Omega)$ be the global RT space defined on \mathcal{T}_h . Note that $H(\text{div})$ -flux reconstruction of DG were developed and analyzed in, e.g., [15]. The corresponding EG-flux reconstruction was introduced in [31, 43]. Note that these prior works assumed that the jump discontinuity is aligned with the mesh. In this case, the flux recovery can be done as follows:

$$\mathbf{u}_h \cdot \mathbf{n}_e = -\{\{\beta \nabla p_h \cdot \mathbf{n}_e\}\}_e + \frac{\sigma(\beta)}{|e|} \llbracket p_h \rrbracket_e.$$

On the other hand, for the case when the discontinuity is allowed within an element, we modify this as follows, which will be coined as ‘‘EIFEM-flux recovery.’’ Namely, for each edge $e \in \mathcal{E}_h$, we define

$$\mathbf{u}_h \cdot \mathbf{n}_e := \frac{1}{|e|} \int_e \left(-\{\{\beta \nabla p_h \cdot \mathbf{n}_e\}\}_e + \frac{\sigma(\beta)}{|e|} \llbracket p_h \rrbracket_e \right) ds, \tag{25}$$

We remark that such definition produces the flux contained in the space $V_h(\Omega)$ and the computation is completely local, and so, it is inexpensive. Furthermore, as desired, the following local and global conservation can be shown to hold:

Proposition 1 *The flux \mathbf{u}_h defined through (25) satisfies the local and global conservation, namely,*

$$\int_{\partial T} \mathbf{u}_h \cdot \mathbf{n} ds = \int_T f dx, \quad \forall T \in \mathcal{T}_h, \tag{26}$$

where \mathbf{n} is the unit outward normal to ∂T , and

$$\int_{\partial \Omega} \mathbf{u}_h \cdot \mathbf{n} ds = \int_{\Omega} f dx,$$

where \mathbf{n} is the unit outward normal to $\partial \Omega$.

We note that the error estimate can be established for both $\|\mathbf{u} - \mathbf{u}_h\|_0$ and $\|\text{div}(\mathbf{u} - \mathbf{u}_h)\|_0$ as follows:

Theorem 3 Let \mathbf{u} be the solution of (4) with condition (5) and \mathbf{u}_h be the EIFEM-flux computed by (25). If $\mathbf{u} \in (H^1(\Omega))^2$, then the following holds.

$$\|\mathbf{u} - \mathbf{u}_h\|_{L^2(\Omega)} \leq Ch\|f\|_{L^2(\Omega)}.$$

Suppose further that for each $T \in \mathcal{T}_h$, $f \in H^1(T)$, then following holds.

$$\sum_{T \in \mathcal{T}_h} \|\text{div}(\mathbf{u} - \mathbf{u}_h)\|_{0,T} \leq Ch\|f\|_{1,h}.$$

Remark 1 We can apply a posteriori error estimates developed in the literature [2, 6, 8, 13, 23] to DB-EIFEM and use it for the adaptive mesh refinement. The recovery-based posteriori error estimates are studied for mixed finite element method in [2, 6, 13, 23], while those for discontinuous Galerkin are studied in [8]. However, due to the presence of interface and nonhomogeneous jump conditions, we leave the technical details for a future investigations.

4 Numerical methods for the evolution of $\Gamma(t)$

In this section, we discuss how to evolve the interface $\Gamma(t)$. The proposed technique is based on the evolution of the level set function Φ . We also discuss how to obtain the zero-level set, or the updated interface from Φ . Basically, once we obtain the velocity $\mathbf{u}_h = (u_h, v_h)$ at time step, $T = t^n$ from DB-EIFEM scheme, we shall obtain the updated level set function using the fifth-order WENO scheme and then obtain the zero-level set using a certain projection step. Throughout this section, we denote ∂_ξ by the partial derivative with respect to the variable ξ .

4.1 WENO scheme for level set equation

As discussed in [16], we recall that the level set (3) is a Hamilton-Jacobi type, we rewrite (3) as

$$\Phi_t + H(\partial_x \Phi, \partial_y \Phi) = 0, \tag{27}$$

where $H(\phi, \psi) = u_h \phi + v_h \psi$. To solve the (27), we propose to apply the fifth-order WENO scheme with the local Lax-Friedrichs (LLF) flux as introduced in [19, 20]. Note that we use the lowest order RT element for the velocity whose degrees of freedom are defined on edges. Therefore, it is more convenient to define DOFs of Φ on the cell centers (see Fig. 3). Throughout this section, for any given function, say Ψ dependent on time and space, by $\Psi_{i,j}^n$ we mean the cell average of Ψ^n on the (i, j) -th cell $[i \Delta x, (i + 1) \Delta x] \times [j \Delta y, (j + 1) \Delta y]$, whose cell center coordinate will be denoted by (x_i, y_j) with $\Psi^n = \Psi(t_n)$.

The WENO scheme ([19, 41]) to approximate the Jacobi-Hamilton (27) can be formulated as follows:

$$\Phi_{i,j}^{n+1} = \Phi_{i,j}^n - \Delta t \widehat{H}((\partial_x \Phi)_{i,j}^-, (\partial_x \Phi)_{i,j}^+, (\partial_y \Phi)_{i,j}^-, (\partial_y \Phi)_{i,j}^+), \tag{28}$$

where \widehat{H} is the local Lax-Friedrichs (LLF) flux, $(\partial_x \Phi)_{i,j}^\pm$ and $(\partial_y \Phi)_{i,j}^\pm$ are fifth-order WENO approximations of $\partial_x \Phi^n$ and $\partial_y \Phi^n$ at (x_i, y_j) , respectively. More precisely,

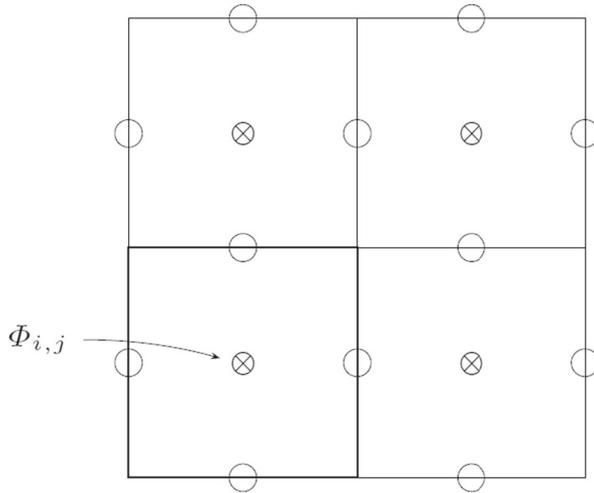


Fig. 3 Degree of freedoms of Φ_h are located at cell center, while those for $\mathbf{u}_h = (u_h, v_h)$ are located at edges

the LLX flux, \widehat{H} is given as follows:

$$\widehat{H}(\phi^-, \phi^+, \psi^-, \psi^+) = H\left(\frac{\phi^+ + \phi^-}{2}, \frac{\psi^+ + \psi^-}{2}\right) - \alpha(\phi^+, \phi^-) \left(\frac{\phi^+ - \phi^-}{2}\right) - \beta(\psi^+, \psi^-) \left(\frac{\psi^+ - \psi^-}{2}\right),$$

where with $I(a^\pm) = [\min(a^+, a^-), \max(a^+, a^-)]$,

$$\alpha(\phi^+, \phi^-) = \max_{\phi \in I(\phi^\pm), \psi \in I(\psi^\pm)} |\partial_{(\partial_x \Phi)} H(\phi, \psi)|, \quad \text{and}$$

$$\beta(\psi^+, \psi^-) = \max_{\psi \in I(\psi^\pm), \phi \in I(\phi^\pm)} |\partial_{(\partial_y \Phi)} H(\phi, \psi)|.$$

We now discuss approximations of the derivatives of Φ^n at the cell edges, i.e., $(\partial_x \Phi)_{i,j}^\pm$ and $(\partial_y \Phi)_{i,j}^\pm$. Note that we only describe $(\partial_x \Phi)_{i,j}^-$ since the others can be obtained in a similar fashion (see [19] for detailed discussions). We let $\phi(x) := (\partial_x \Phi)^n(x, y_j)$. We denote $\phi_k = \phi(x_k)$ and define the left-biased difference of ϕ 's at x_i as $\Delta^- \phi_i = \phi_i - \phi_{i-1}$. The third-order approximations of $\partial_x \phi_i$ based on the difference at the stencils- $\mathcal{S}_r(i) := \{x_{i-2+r}, x_{i-1+r}, x_{i+r}\}$ ($r = 0, 1, 2$) are given as follows:

$$\begin{aligned} (\partial_x \phi)_i^{-,0} &= \frac{1}{3} \frac{\Delta^- \phi_{i-2}}{\Delta x} - \frac{7}{6} \frac{\Delta^- \phi_{i-1}}{\Delta x} + \frac{11}{6} \frac{\Delta^- \phi_i}{\Delta x}, \\ (\partial_x \phi)_i^{-,1} &= -\frac{1}{6} \frac{\Delta^- \phi_{i-1}}{\Delta x} + \frac{5}{6} \frac{\Delta^- \phi_i}{\Delta x} + \frac{1}{3} \frac{\Delta^- \phi_{i+1}}{\Delta x}, \\ (\partial_x \phi)_i^{-,2} &= \frac{1}{3} \frac{\Delta^- \phi_i}{\Delta x} + \frac{5}{6} \frac{\Delta^- \phi_{i+1}}{\Delta x} - \frac{1}{6} \frac{\Delta^- \phi_{i+2}}{\Delta x}. \end{aligned}$$

The idea of WENO reconstruction for $(\partial_x \phi)_i^-$ is then to take a linear combination of the above approximations as follows:

$$(\partial_x \phi)_i^- = \sum_{k=0}^2 \omega_k (\partial_x \phi)_i^{-,k},$$

where weights ω_k are defined as $\omega_k = \frac{\alpha_k}{\sum_{s=0}^2 \alpha_s}$ for $k = 0, 1, 2$. The parameters α_k 's are given as $\alpha_k = \frac{d_k}{(\epsilon + \beta_k)^2}$ with d_k 's and β_k 's being given as follows:

$$\begin{aligned} d_0 &= \frac{1}{10}, \quad d_1 = \frac{3}{5}, \quad d_2 = \frac{3}{10}, \\ \beta_0 &= \frac{13}{12} \left(\frac{\Delta^- \phi_{i-2} - 2\Delta^- \phi_{i-1} + \Delta^- \phi_i}{\Delta x} \right)^2 + \frac{1}{4} \left(\frac{\Delta^- \phi_{i-2} - 4\Delta^- \phi_{i-1} + 3\Delta^- \phi_i}{\Delta x} \right)^2, \\ \beta_1 &= \frac{13}{12} \left(\frac{\Delta^- \phi_{i-1} - 2\Delta^- \phi_i + \Delta^- \phi_{i+1}}{\Delta x} \right)^2 + \frac{1}{4} \left(\frac{\Delta^- \phi_{i-1} - \Delta^- \phi_{i+1}}{\Delta x} \right)^2, \\ \beta_2 &= \frac{13}{12} \left(\frac{\Delta^- \phi_i - 2\Delta^- \phi_{i+1} + \Delta^- \phi_{i+2}}{\Delta x} \right)^2 + \frac{1}{4} \left(\frac{3\Delta^- \phi_i - 4\Delta^- \phi_{i+1} + \Delta^- \phi_{i+2}}{\Delta x} \right)^2. \end{aligned}$$

Finally, the parameter ϵ is chosen typically as a small positive number, which is to avoid overflow when β_k is close to zero. In our experiments, we set ϵ at $\epsilon = 10^{-6}$.

4.2 Interface reconstruction from level set function

In this section, we describe how to obtain Γ_h^{n+1} as a zero-level set of Φ_h^{n+1} , following the technique introduced in [16]. Note that the level set function is evaluated at centers of elements, we need to find points on the interface with these values.

We call a cell center coordinate (x_i, y_j) of an element, *control point* if $\Phi_{i,j}^n < 0$ and there exists at least one adjacent element at which Φ^n takes positive value. We then reconstruct the interface by projecting these control points on the interface, called the *interface point*. With $\Phi := \Phi^n$, let $X = (x_i, y_j)$ be a control point. We let \mathbf{n} be the unit gradient vector of Φ , i.e.,

$$\mathbf{n} = -\frac{\nabla \Phi}{\|\nabla \Phi\|}.$$

Then we can find the interface point $X^* = (x_i^*, y_j^*)$, which is the projection of (x_i, y_j) along \mathbf{n} . Namely,

$$X^* = X + t\mathbf{n},$$

where t is the positive solution of the quadratic equation given as follows:

$$\frac{1}{2}(\mathbf{n}^T He(\Phi)\mathbf{n})t^2 + \|\nabla \Phi(X)\|t + \Phi(X) = 0.$$

Note that $He(\Phi)$ is the Hessian matrix of Φ evaluated at the point X . Finally, the curvature κ^n of the interface can be approximated by the following formula:

$$\kappa = \nabla \cdot \mathbf{n} = -\frac{\phi_{xx}\phi_y^2 - 2\phi_{xy}\phi_x\phi_y + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}}. \tag{29}$$

Once $\phi_x, \phi_y, \phi_{xx}, \phi_{xy}, \phi_{yy}$ are approximated by the central difference schemes, we obtain curvature by substituting them in (29). Note that we only need to compute the curvature for the interface elements (Fig. 4).

5 Numerical experiments

In this section, we demonstrate the performance of the proposed method by performing a couple of numerical experiments of interface problems with nonhomogeneous jump conditions. The experiments are conducted with a variety of interfaces Γ and with coefficients β^\pm with different jumps.

For all the examples, we fix the domain $\Omega = (-2, 2)^2$. The domain Ω is then triangulated uniformly with rectangles of size h . The triangulation will be denoted by \mathcal{T}_h . The computational setting is similar to that presented in [16]. Let $\Omega_r = \{(x, y) \in \Omega : \sqrt{x^2 + y^2} \leq r\}$ for any $r \in \mathbb{R}$. Let α be a fixed positive real number. The source term f and boundary condition for p are given as follows:

$$f(x, y) = \begin{cases} \frac{6V_0}{\alpha^2}(\alpha - \sqrt{x^2 + y^2}), & \text{if } (x, y) \in \Omega_\alpha \\ 0, & \text{otherwise,} \end{cases}$$

$$p(x, y) = -\frac{V_0\alpha}{\beta^+} \log(\sqrt{x^2 + y^2}), \quad \text{on } \partial\Omega,$$

where V_0 is a controlling parameter. For all the experiments, we set the injection rate IR as $IR = 2\pi V_0\alpha$. We remark that surface tension τ is then determined by the

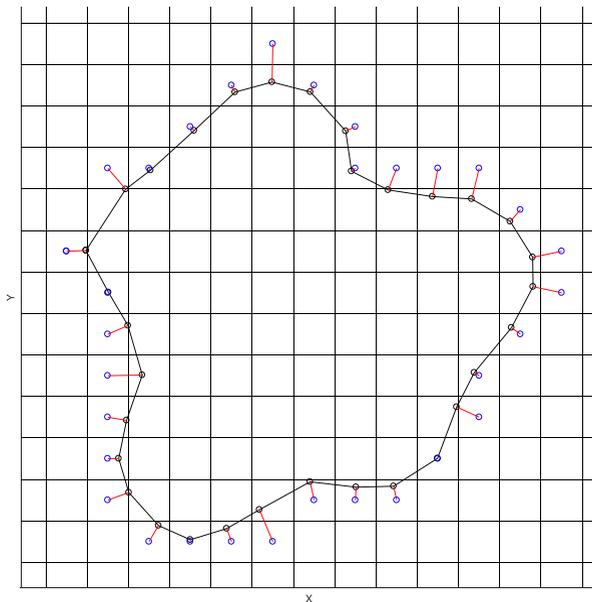


Fig. 4 Blue circles are control points and black circles are the projected points along the red lines

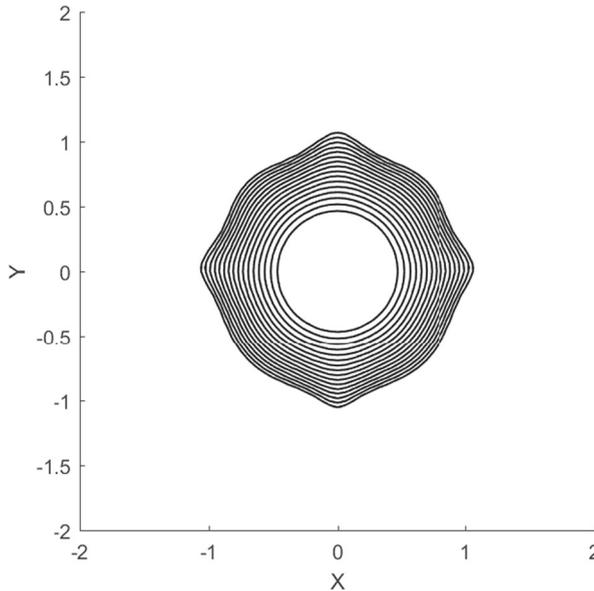


Fig. 5 Inner interface shows the interface at $t = 0$ and outer interface shows the interface at $t = 8$ for Example 1

following equation:

$$\tau = \frac{d_0 \text{IR}}{2\pi\beta^+}. \tag{30}$$

All the numerical experiments below are conducted on PC with an Intel(R) Core(TM) i7-3770 CPU@ 3.40GHz processor. The DB-EIFEM system is solved by applying the solver discussed in Section 3.2.1. For the implementation of AMG preconditioner, C++ open software library AMGCL developed by Demidov [11, 12] were employed.

We begin with an example for which analytic solution is given. The main purpose of this setting is to consider the convergence of the method.

Example 1 We shall consider the benchmark problem [16], formulated with an analytic solution and the interface of initial circle shape given as follows:

$$p(x, y) = \begin{cases} \frac{V_0}{\beta^-} \left(\frac{2\sqrt{x^2+y^2}^3}{3\alpha^2} - \frac{3\sqrt{x^2+y^2}^2}{2\alpha} \right) + C_1, & \text{if } 0 \leq \sqrt{x^2 + y^2} \leq \alpha \\ -\frac{V_0\alpha}{\beta^-} \log(\sqrt{x^2 + y^2}) + C_0 & \text{if } \alpha < \sqrt{x^2 + y^2} \leq r_\Gamma \\ -\frac{V_0\alpha}{\beta^+} \log(\sqrt{x^2 + y^2}), & \text{otherwise,} \end{cases}$$

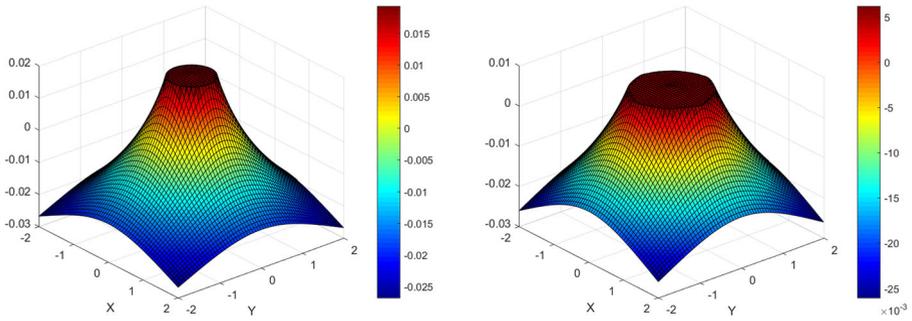


Fig. 6 Pressure field at $T = 0$ and at $T = 16$, respectively for Example 1

where $r_\Gamma = \sqrt{2\alpha V_0 t + r_0^2}$ with r_0 the radius of the initial interface of the circle shape, $V_0 = 0.25$, $\beta^+ = 1$, $\beta^- = 100$, $\alpha = 0.1$, $d_0 = 2.5E-3$ and

$$C_0 = \frac{\tau}{r_\Gamma} + V_0 \alpha \log(r_\Gamma) \left(\frac{1}{\beta^-} - \frac{1}{\beta^+} \right),$$

$$C_1 = C_0 - \frac{V_0 \alpha \log(\alpha)}{\beta^-} + \frac{5V_0 \alpha}{6\beta^-}.$$

Note that the initial interface is given as $\Gamma(0) = \{x^2 + y^2 = r_0^2\}$ with $r_0 = 0.41$. Note that Fig. 5 shows the evolution of the interface from the time level $t = 0$ to the time level $t = 16$ in the collage images.

Figures 6 and 7 show the pressure and velocity fields at time level $t = 0$ and $t = 16$ as snap shots. We present two tables, one for error analysis of the pressure (see Table 1) and the other for error analysis for the velocity (see Table 2). These results show the optimal convergence rate, which is predicted by the theory.

The performance of the preconditioner proposed in Section 3.2.1 for resulting system is reported in Table 3. As demonstrated, the optimal performance of the solver

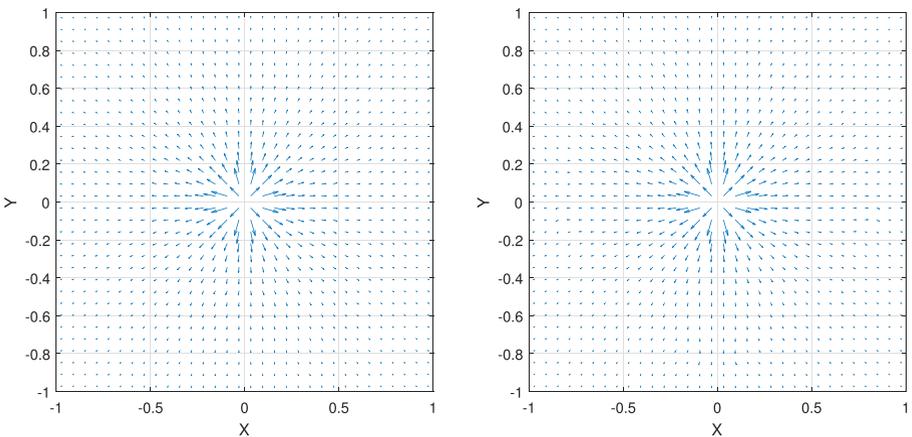


Fig. 7 Velocity field at $T = 0$ and at $T = 16$ (Example 1)

Table 1 L^2 and Energy norm error for pressure approximation for Example 1. Both errors are computed at the time level $T = 1$ and the time step size is set at $\Delta t = (1/16)h^2$

# of elements	$\ p - p_h\ _{0,\Omega}$	Order	$\ p - p_h\ _{1,h}$	Order
16^2	$7.014E-3$		$1.221E-2$	
32^2	$2.233E-3$	1.651	$4.750E-3$	1.362
64^2	$4.539E-4$	2.299	$1.901E-3$	1.321
128^2	$1.446E-5$	4.972	$8.776E-4$	1.115

is established. Note that the maximum number of AMG iterations for block matrices, $A_{\approx 11}$ and $A_{\approx 22}$ were set at the fixed number 10 with one Gauss-seidel iteration. We note that DB-EIFEM scheme provides a setting where the system solved changes only due to the location of interface and therefore, the performance of the proposed preconditioner stays the same with the different time levels.

In the following, we shall present more challenging case where the interface conditions are complicated. For the next two examples, we shall consider the standard cylindrical coordinate of $(x, y) \in \Omega$, which will be denoted by $(\rho, \theta) := (\rho(x, y), \theta(x, y))$. This is convenient to express the interface conditions.

Example 2 We consider the problem with a symmetric star-shaped interface (Figs. 8, 9 and 10). The initial interface is given as follows:

$$\Gamma(0) = \{(x, y) \in \Omega : \rho(x, y) = r_0 + 0.1 \sin(5\theta(x, y))\},$$

where $r_0 = 0.7$. Parameters used are $V_0 = 0.25, \beta^+ = 1, \beta^- = 100, \alpha = 0.3$, and $d_0 = 2.3E-3$ and $r_0 = 0.7$.

Example 3 We consider the problem with two different initial asymmetric interfaces (Fig. 11), which are given as follows, respectively:

$$\Gamma_1(0) = \{(x, y) \in \Omega : \rho = r_0 + 0.05(\sin(2\theta) + \cos(3\theta))\}$$

$$\Gamma_2(0) = \{(x, y) \in \Omega : \rho = r_0 + 0.05(\sin(5\theta) + \cos(3\theta))\},$$

where $r_0 = 0.55$. The parameters used are $V_0 = 0.25, \beta^+ = 1, \beta^- = 100, \alpha = 0.3$, $d_0 = 2.1E-3$.

Table 2 L^2 norm error for velocity approximation for Example 1. It is computed at the time level $T = 1$ and the time step size is set at $\Delta t = (1/16)h^2$

# of elements	$\ \mathbf{u} - \mathbf{u}_h\ _{0,\Omega}$	Order	$\ \nabla \cdot (\mathbf{u} - \mathbf{u}_h)\ _{0,\Omega}$	Order
16^2	$5.317E-2$		$1.204 \cdot E-0$	
32^2	$3.534E-2$	0.589	$8.709 \cdot E-1$	0.467
64^2	$1.197E-2$	0.562	$4.302 \cdot E-1$	1.018
128^2	$5.500E-3$	1.122	$2.316 \cdot E-1$	0.893

Table 3 Performance of auxiliary space preconditioner-PCG for Example 1

# of elements	# of iterations	CPU time
16^2	10	0.16
32^2	12	0.57
64^2	11	2.60
128^2	12	8.59
256^2	12	25.49
512^2	12	101.46

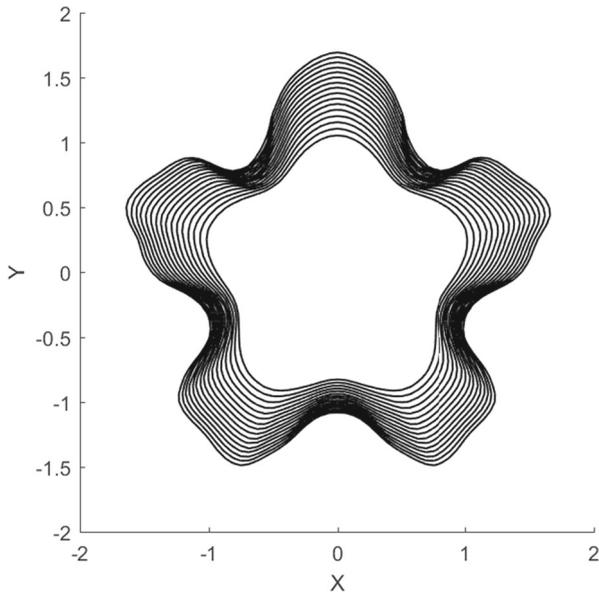


Fig. 8 Inner interface shows the interface at $T = 0$ and outer interface shows the interface at $T = 8$ for Example 2

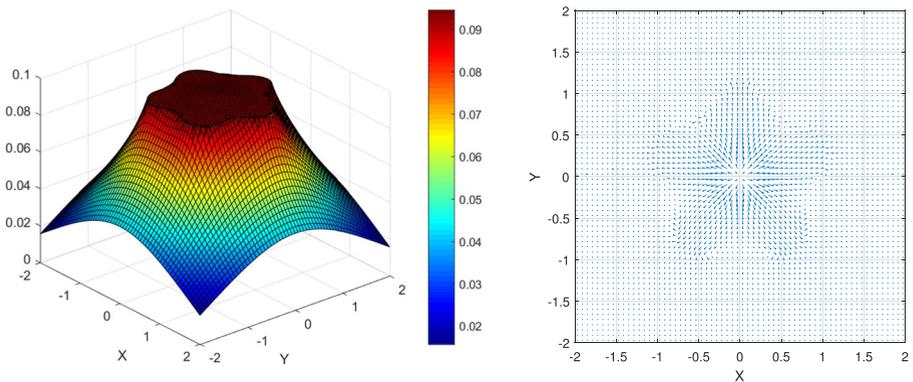


Fig. 9 Pressure (left column) and velocity field (right column) for $T = 0$ for Example 2

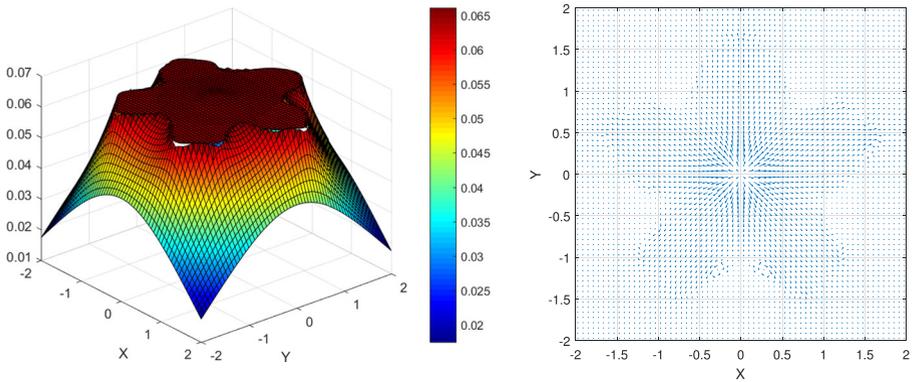


Fig. 10 Pressure (left column) and velocity field (right column) for $T = 8$ for Example 2

Figure 12 shows the importance of flux and its conservative property. We have attempted to use flux obtained from the DB-IFEM [9] method, which is not locally conservative instead of the flux we proposed to use. It is noticed that the computation breaks down at the time level $t = 0.25$.

6 Conclusions

We presented a locally conservative discontinuous bubble scheme for interface problem with nonhomogeneous interface problem coupled with transports of level set formulation for evolution of the interface. For simulating a coupled-flow and transport, particularly the Hele-Shaw equation, we demonstrate the importance of the locally conservative flow scheme. We also demonstrate the newly proposed scheme

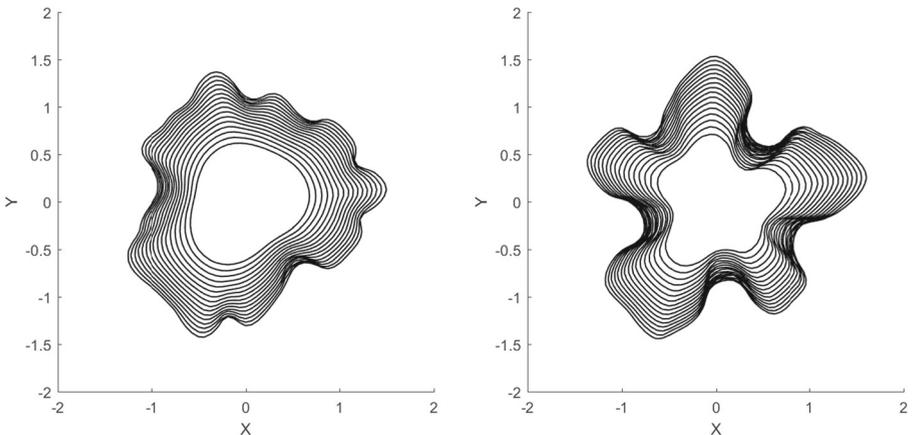


Fig. 11 Evolution of interfaces for $\Gamma_1(t)$ for $t \geq 0$ (left) and $\Gamma_2(t)$ for $t \geq 0$ (right). For both cases, inner interface shows the interface at $T = 0$ and outer interface shows the interface at $T = 8$ for Example 3

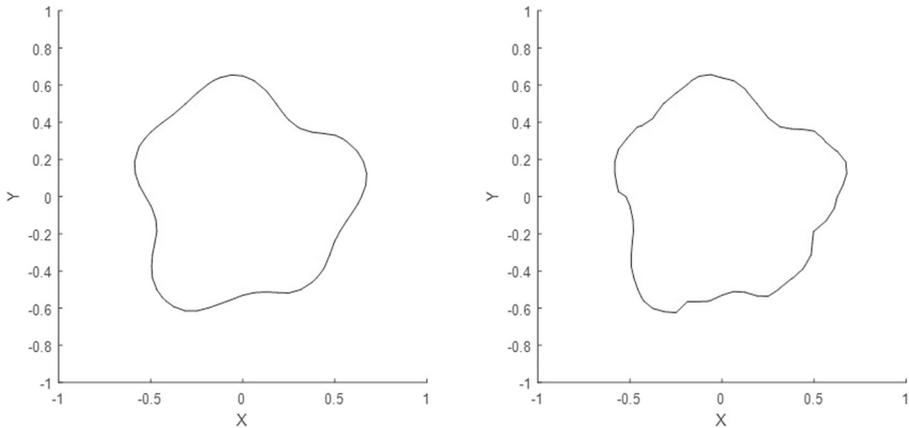


Fig. 12 Interface at time $t = 0.25$ where initial interface is $\Gamma_2(0)$. Left is a numerical result by DB-EIFEM and right is a numerical result by DB-IFEM

is ideal for the evolution problem since a fast solver based on the algebraic multigrid method can be incorporated.

Funding The second author is supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2020R1C1C1A01005396).

Data availability The datasets generated during the current study are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Arnold, D.N., Brezzi, F.: Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates. *ESAIM: Math. Model. Numer. Anal.* **19**, 7–32 (1985)
2. Bank, R.E., Li, Y.: Superconvergent recovery of Raviart–Thomas mixed finite elements on triangular grids. *J. Sci. Comput.* **81**, 1882–1905 (2019)
3. Bastian, P., Rivière, B.: Superconvergence and $H(\text{div})$ projection for discontinuous galerkin methods. *Int. J. Numer. Methods Fluids* **42**, 1043–1057 (2003)
4. Belytschko, T., Black, T.: Elastic crack growth in finite elements with minimal remeshing. *Int. J. Numer. Methods Eng.* **45**, 601–620 (1999)
5. Belytschko, T., Parimi, C., Moës, N., Sukumar, N., Usui, S.: Structured extended finite element methods for solids defined by implicit surfaces. *Int. J. Numer. Methods Eng.* **56**, 609–635 (2003)
6. Brandts, J.H.: Superconvergence for triangular order $k = 1$ Raviart–Thomas mixed finite elements and for triangular standard quadratic finite element methods. *Appl. Numer. Math.* **34**, 39–58 (2000)
7. Brezzi, F., Fortin, M.: *Mixed and Hybrid Finite Element Methods*, vol. 15. Springer, New York (1991)
8. Cai, Z., Ye, X., Zhang, S.: Discontinuous Galerkin finite element methods for interface problems: a priori and a posteriori error estimations. *SIAM J. Numer. Anal.* **49**, 1761–1787 (2011)
9. Chang, K.S., Kwak, D.Y.: Discontinuous bubble scheme for elliptic problems with jumps in the solution. *Comput. Methods Appl. Mech. Eng.* **200**, 494–508 (2011)

10. Chou, S.H., Kwak, D.Y., Wee, K.T.: Optimal convergence analysis of an immersed interface finite element method. *Adv. Comput. Math.* **33**, 149–168 (2010)
11. Demidov, D.: Amgcl: a c++ library for solution of large sparse linear systems with algebraic multigrid method. <https://github.com/ddemidov/amgcl> (2017)
12. Demidov, D.: amgcl: an efficient, flexible, and extensible algebraic multigrid implementation. *Lobachevskii J. Math.* **40**, 535–546 (2019)
13. Dupont, T.F., Keenan, P.T.: Superconvergence and postprocessing of fluxes from lowest-order mixed methods on triangles and tetrahedra. *SIAM J. Sci. Comput.* **19**, 1322–1332 (1998)
14. Entov, V., Etingof, P.: On a generalized two-fluid hele-shaw flow. *Eur. J. Appl. Math.* **18**, 103–128 (2007)
15. Ern, A., Nicaise, S., Vohralík, M.: An accurate $H(\text{div})$ flux reconstruction for discontinuous galerkin approximations of elliptic problems. *C. R. Math.* **345**, 709–712 (2007)
16. Hou, T.Y., Li, Z., Osher, S., Zhao, H.: A hybrid method for moving interface problems with application to the Hele–Shaw flow. *J. Comput. Phys.* **134**, 236–252 (1997)
17. Jeon, Y., Tran, M.L.: Numerical analysis of interface hybrid difference methods for elliptic interface equations. *J. Comput. Appl. Math.* **377**, 112869 (2020)
18. Jeon, Y., Shin, D.: Immersed hybrid difference methods for elliptic boundary value problems by artificial interface conditions. *Electron. Res. Arch.* **29**, 3361 (2021)
19. Jiang, G.-S., Peng, D.: Weighted ENO schemes for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.* **21**, 2126–2143 (2000)
20. Jiang, G.-S., Shu, C.-W.: Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126**, 202–228 (1996)
21. Jo, G., Kwak, D.Y.: An IMPES scheme for a two-phase flow in heterogeneous porous media using a structured grid. *Comput. Methods Appl. Mech. Eng.* **317**, 684–701 (2017)
22. Jo, G., Young, K.D., Lee, Y.-J.: Locally conservative immersed finite element method for elliptic interface problems. *J. Sci. Comput.* **87** (2021)
23. Kim, K.-Y.: Guaranteed and asymptotically exact a posteriori error estimator for lowest-order Raviart–Thomas mixed finite element method. *Appl. Numer. Math.* **165**, 357–375 (2021)
24. Krysl, P., Belytschko, T.: An efficient linear-precision partition of unity basis for unstructured meshless methods. *Commun. Numer. Methods Eng.* **16**, 239–255 (2000)
25. Kwak, D.Y., Wee, K.T., Chang, K.S.: An analysis of a broken p_1 -nonconforming finite element method for interface problems. *SIAM J. Numer. Anal.* **48**, 2117–2134 (2010)
26. Kwak, D.Y., Jin, S., Kyeong, D.: A stabilized p_1 -nonconforming immersed finite element method for the interface elasticity problems. *ESAIM: Math. Model. Numer. Anal.* **51**, 187–207 (2017)
27. Kwon, I., Jo, G.: A consistent discontinuous bubble scheme for elliptic problems with interface jumps. *J. Korean Soc. Ind. Appl. Math.* **24**, 143–159 (2020)
28. Kwon, I., Kwak, D.Y.: Discontinuous bubble immersed finite element method for Poisson–Boltzmann equation. *Commun. Comput. Phys.* **25**, 928–946 (2019)
29. Kyeong, D., Kwak, D.Y.: An immersed finite element method for the elasticity problems with displacement jump. *Adv. Appl. Math. Mech.* **9**, 407–428 (2017)
30. Lee, L., LeVeque, R.J.: An immersed interface method for incompressible Navier–Stokes equations. *Wave Motion* **25**, 832–856 (2003)
31. Lee, S., Lee, Y., Wheeler, M.: A locally conservative enriched Galerkin approximation and efficient solver for elliptic and parabolic problems. *SIAM J. Sci. Comput.* **38**, A1404–A1429 (2016)
32. Legrain, G., Moes, N., Verron, E.: Stress analysis around crack tips in finite strain problems using the extended finite element method. *Int. J. Numer. Methods Eng.* **63**, 290–314 (2005)
33. LeVeque, R.J., Li, Z.: The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. *SIAM J. Numer. Anal.* **31**, 1019–1044 (1994)
34. LeVeque, R.J., Li, Z.: Immersed interface methods for stokes flow with elastic boundaries or surface tension. *SIAM J. Sci. Comput.* **18**, 709–735 (1997)
35. Li, Z.: Immersed interface methods for moving interface problems. *Numer. Algorithms* **14**, 269–293 (1997)
36. Li, Z., Lin, T., Wu, X.: New cartesian grid methods for interface problems using the finite element formulation. *Numer. Math.* **96**, 61–98 (2003)
37. Li, Z., Lin, T., Lin, Y., Rogers, R.C.: An immersed finite element space and its approximation capability. *Numer. Methods Partial Differ. Equ.* **20**, 338–367 (2004)

38. Lin, T., Lin, Y., Rogers, R., Ryan, M.L.: A rectangular immersed finite element space for interface problems. *Adv. Comput. Theory Pract.* **7**, 107–114 (2001)
39. Lin, T., Lin, Y., Zhang, X.: Partially penalized immersed finite element methods for elliptic interface problems. *SIAM J. Numer. Anal.* **53**, 1121–1144 (2015)
40. Moës, N., Dolbow, J., Belytschko, T.: A finite element method for crack growth without remeshing. *Int. J. Numer. Methods Eng.* **46**, 131–150 (1999)
41. Osher, S., Shu, C.-W.: High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations. *SIAM J. Numer. Anal.* **28**, 907–922 (1991)
42. Raviart, P.A., Thomas, J.M.: A mixed finite element method for 2-nd order elliptic problems. *Mathematical aspects of finite element methods*, pp. 292–315 (1977)
43. Sun, S., Liu, J.: A locally conservative finite element method based on piecewise constant enrichment of the continuous galerkin method. *SIAM J. Sci. Comput.* **31**, 2528–2548 (2009)
44. Wheeler, M.F.: An elliptic collocation-finite element method with interior penalties. *SIAM J. Numer. Anal.* **15**, 152–161 (1978)
45. Zhang, C., LeVeque, R.J.: The immersed interface method for acoustic wave equations with discontinuous coefficients. *Wave Motion* **25**, 237–263 (1997)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.