Journal of Computational and Applied Mathematics 362 (2019) 540–559

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

An immersed discontinuous finite element method for the Stokes problem with a moving interface

Slimane Adjerid, Nabil Chaabane, Tao Lin, Pengtao Yue [*]

Department of Mathematics, Virginia Tech, Blacksburg, VA 24061, USA

A R T I C L E I N F O

Article history:
Received 31 December 2017
Received in revised form 27 May 2018

Keywords:
Immersed finite element
Discontinuous Galerkin
Stokes problem
Interfacial flow
Drop deformation
Surface tension

A B S T R A C T

We present a discontinuous immersed finite element (IFE) method for incompressible interfacial flows that are governed by the Stokes equations. The method is based on a Cartesian mesh with elements cut by the moving interface. On this fixed unfitted mesh, we employ an immersed Q_1/Q_0 finite element space constructed according to the location of the interface and pertinent interface jump conditions. As such, the smearing of solution across the interface is greatly reduced. The interface, represented by a sequence of marker points, is advected on the fixed background mesh by the local fluid velocity. The mesh is locally refined near the interface to further improve accuracy. Compared with the phase-field method on adaptive meshes, our method can achieve the same level of accuracy with much less degrees of freedoms. We present some numerical examples to validate and demonstrate the capability of the proposed method.

1. Introduction

The dispersions of one liquid in another immiscible liquid are known as emulsions, and the phenomenon arises in several industries such as medicine, oil recovery and material processing. The emulsions usually yield a system consisting of droplets immersed in a matrix fluid separated by interfaces. An assessment of the drop size and shape helps control the physical properties of the emulsions such as viscosity, stability and transport properties, and thus it is necessary to understand the dynamics of drop deformation [1–3].

Usually, conventional methods with body-fitted meshes can be used to solve interface problems modeling emulsions. This has been done using finite difference schemes [4], finite element methods [5,6], boundary integral and boundary element methods [7–9], and discontinuous Galerkin finite element methods [10]. In general, the solutions are satisfactory if the mesh is tailored to fit the interface. This task is relatively simple if the interface does not change shape or location. However, this is hardly the case when dealing with fluid interfaces, which always undergo significant morphological changes and even topological transitions. As a consequence, the body-fitted mesh becomes inefficient as a new mesh has to be generated for each new configuration of the interface. Furthermore the use of finite element methods, which is the subject of this manuscript, exhibits a change in the degrees of freedom as well as nodal positions, which will add an extra computational cost and complexity.

As an alternative, methods based on non-fitted fixed meshes such as the front-tracking method [11–13], the volume-of-fluid method [14–16], the level set method [17,18], and the phase-field method [19,20] have been developed. These methods differ from conventional methods in that the mesh does not conform to the interface: interface motion is tracked explicitly by marker points or implicitly by a scalar field. Usually, the interfacial tension is represented by a body force distributed over

---

[*] Corresponding author.
E-mail address: ptyue@math.vt.edu (P. Yue).

https://doi.org/10.1016/j.cam.2018.07.033

a narrow band of the interface [21], and the discontinuities of fluid properties across the interface are also smeared over the same narrow band. In this way, different fluid components can be described by a single set of governing equations. This greatly improves the computational efficiency, however, at the cost of the accuracy near the interface. In order to maintain the sharpness of the interface, different methods have been proposed, e.g., the immersed interface method [22,23], the extended finite element method [24], and the ghost fluid method [25].

In this paper, we apply an immersed finite element (IFE) method to solve the Stokes problem with a moving interface. The most important feature of the IFE method lies in the IFE basis that automatically takes care of the discontinuities in solution regardless of the interface location in a fixed Cartesian background mesh. Furthermore, having the possibility to use Cartesian meshes whenever the geometry of the domain allows is also a desirable feature for many applications. Another advantage of the IFE method is that it models the actual interfacial force instead of a body force smeared over a narrow region of the interface as in other fixed-mesh methods. However, the standard finite element spaces are not guaranteed to capture the singularities in the solutions; therefore, suitable finite element spaces such as the immersed finite element spaces are needed. Fortunately, the construction of the IFE basis is only required in the interface cells and thus does not incur too much additional computational cost. In summary, the advantage of the IFE method is a collective effect of all the benefits mentioned above. It should be noted that another "immersed" finite element method was also proposed for Stokes interface problems in [26]. But that method is based on locally refining elements such that the newly added elements fit the immersed boundary; it is thus totally different from the IFE method in this work.

The IFE basis can be used in either the traditional finite element methods or the discontinuous Galerkin methods. In this work, we show how to construct IFE spaces capable of capturing the non-smoothness of the solution and construct discontinuous Galerkin schemes that take into account the presence of surface forces along the interface. We choose the discontinuous Galerkin method because it has the desired stability with the Q_1/Q_0 finite elements for the Stokes problem and is able to handle the discontinuity of the IFE basis functions across element edges [27]. Furthermore, the discontinuous Galerkin method allows for the adaptive mesh with hanging nodes which makes the application of adaptive refinement to enhance the accuracy around the interface much easier.

The IFE method has been extensively studied for second order elliptic equations with a stationary interface in [28–39]. More recently, immersed finite methods have been proposed for interface problems in linear elasticity [40,41] and acoustics [42–44]. In [45], a parabolic problem with a moving interface was solved using an IFE method where *a priori* knowledge of the interface location was assumed. However, solving the Stokes problem with a moving fluid interface is more challenging since the interface itself is driven by the velocity and it is part of the quantities to be found. Thus we will show in this work how to model the interface and how to track it in time.

The outline of the paper is as follows: In Section 2, we state the two-dimensional Stokes interface problem with a moving interface and briefly review the construction of corresponding IFE spaces and construct their counterpart for the three-dimensional axisymmetric Stokes interface problem in Section 3. The IFE weak formulations are discussed in Section 4 and numerical examples are used to validate the new method in Section 5.

2. Two-dimensional Stokes interface problem

In this section, we treat the two-dimensional Stokes interface problem, which models two fluids separated by an interface $\Gamma(t)$. We assume that $\Gamma(t)$ separates the two-dimensional domain Ω into two subdomains $\Omega^{\pm}(t)$, where each domain contains one fluid as illustrated in Fig. 1. The fluids occupying the regions $\Omega^{+}(t)$ and $\Omega^{-}(t)$, respectively, have constant viscosities ν^{+} and ν^{-}. We follow the model presented in [46,11] and assume that the interface $\Gamma(t)$ satisfies the following system of ODEs

$$\begin{cases} \dfrac{d}{dt}\mathbf{X} & = \mathbf{u}(\mathbf{X},t), \quad t \in [0,T], \quad \forall\, \mathbf{X} \in \Gamma(t), \\ \mathbf{X}|_{t=0} & = \mathbf{X}_0, \qquad \text{with } \mathbf{X}_0 \in \Gamma(0). \end{cases} \tag{2.1a}$$

Here $\mathbf{u} = (u_1, u_2)^T$ is the solution to the transient two-dimensional Stokes interface problem:

$$-\nabla \cdot \mathbf{S}(\mathbf{u}(\mathbf{X},t), p(\mathbf{X},t)) = \mathbf{0}, \qquad \text{if } \mathbf{X} \in \Omega^{-}(t) \cup \Omega^{+}(t), \quad t > 0, \tag{2.1b}$$

$$\nabla \cdot \mathbf{u}(\mathbf{X},t) = 0, \qquad \text{if } \mathbf{X} \in \Omega^{-}(t) \cup \Omega^{+}(t), \quad t > 0, \tag{2.1c}$$

$$\mathbf{u}(\mathbf{X},t) = \mathbf{g}(\mathbf{X},t), \ \text{ on } \partial\Omega, \quad t > 0, \tag{2.1d}$$

with $\mathbf{S}$ being the stress tensor defined as

$$\mathbf{S}(\mathbf{u},p) = \nu\boldsymbol{\epsilon}(\mathbf{u}) - p\mathbf{I}, \quad \boldsymbol{\epsilon}(\mathbf{u}) = \nabla\mathbf{u} + (\nabla\mathbf{u})^T. \tag{2.1e}$$

The Stokes interface system is closed by the following jump conditions across the interface

$$[\mathbf{S}(\mathbf{u}(\mathbf{X},t), p(\mathbf{X},t))\mathbf{n}] = \boldsymbol{\sigma}(\mathbf{X},t), \quad \text{if } \mathbf{X} \in \Gamma(t), \tag{2.1f}$$

$$[\mathbf{u}(\mathbf{X},t)] = \mathbf{0}, \qquad \text{if } \mathbf{X} \in \Gamma(t), \tag{2.1g}$$

where $\mathbf{n}$ is the unit normal vector to the interface and pointing out of Ω^{+}, $\boldsymbol{\sigma}$ is the surface force defined along the interface $\Gamma(t)$, and $[\cdot] = (\cdot)|_{-}^{+}$ denotes the jump across the interface.

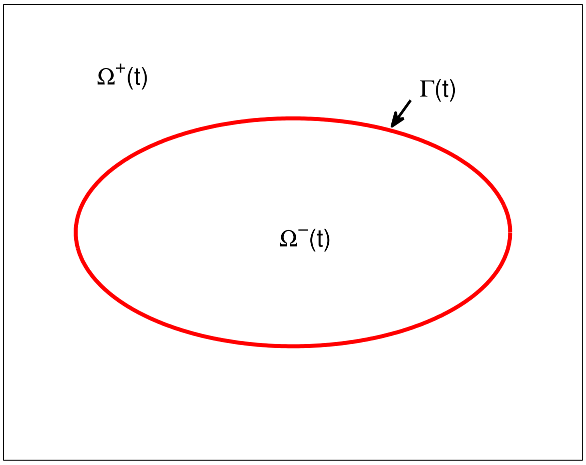


Fig. 1. A rectangular domain Ω with an immersed interface $\Gamma(t)$.

We note that the interface evolves in time according to (2.1a) and an interface tracking procedure cannot be implemented to track all the points on the interface. Instead, we assume that at the initial time t_0, the interface $\Gamma(t)$ is sampled uniformly using a set of N control points. These control points, which we shall denote $\mathcal{X}(t) = \{\mathcal{X}^{(1)}(t), \mathcal{X}^{(2)}(t), \dots, \mathcal{X}^{(N)}(t)\}$, are ordered in space, evolve in time, and are used to approximate the interface. The approximate interface is denoted by $\tilde{\Gamma}(t)$ and is defined by the piecewise linear curve interpolating the control points $\mathcal{X}(t)$.

Due to the constraint (2.1f), the velocity gradient $\nabla\mathbf{u}$ and the pressure p are not continuous across the interface. Therefore, the standard finite element basis functions cannot capture the solution singularities on interface elements in non-fitted meshes. To circumvent this problem, we apply the Q_1/Q_0 immersed finite element space for the velocity and pressure that captures the solution singularity as well as the non-homogeneous nature of the constraint (2.1f). A detailed procedure for constructing Q_1/Q_0 IFE functions using piecewise bilinear polynomial approximations for the velocity and piecewise constant approximations for the pressure are given in [27]. For the sake of completeness we will briefly discuss the main steps of the procedure.

Without loss of generality, we assume that Ω is a rectangle or a union of several rectangles. Let $\mathcal{T}_h$ be an arbitrary uniform rectangular partition of the domain Ω and let elements cut by the interface be called interface elements; otherwise we call them non-interface elements. Let $\mathcal{T}_h^i,\ \mathcal{T}_h^n$, respectively, denote the set of all interface elements and non-interface elements with $\mathcal{T}_h^n = \mathcal{T}_h \setminus \mathcal{T}_h^i$. We assume that, at time t, the approximate interface $\tilde{\Gamma}(t)$ intersects at most two edges of any given element and we consider an element to be a non-interface element if the interface intersects the element at either one vertex or two adjacent vertices. For a typical rectangular interface element $T \in \mathcal{T}_h^i$ we assume that the approximate interface $\tilde{\Gamma}(t)$ intersects two edges at D and E referred to as interface points. The interface $\widetilde{DE} = \tilde{\Gamma}(t) \cap T$ is approximated by the line segment $\overline{DE}$ which separates T into two polygonal domains T^+ and T^- such that T^+ contains vertices of T that are in Ω^+.

On every interface element T we approximate the solution vector $\mathbf{U} = \begin{pmatrix}\mathbf{u}\\ p\end{pmatrix}$ by the piecewise polynomial IFE vector function $\boldsymbol{\Phi}$ having the form

$$\boldsymbol{\Phi}(x,y) = \boldsymbol{\Phi}^s(x,y) = \begin{pmatrix}\phi_1^s(x,y)\\ \phi_2^s(x,y)\\ \phi_3^s(x,y)\end{pmatrix}, \quad \text{for } (x,y) \in T^s, \quad s = +,-, \tag{2.2a}$$

where

$$\phi_j^s(x,y) = a_j^s + b_j^s x + c_j^s y + d_j^s xy, \ \ j = 1,2, \ \ s = +,-, \tag{2.2b}$$

$$\phi_3^s(x,y) = a_3^s, \ \ s = +,-. \tag{2.2c}$$

For $\boldsymbol{\sigma} = \mathbf{0}$ we find 9 IFE shape functions $\boldsymbol{\Phi}_i$, $i = 1,2,\dots,9$ of the form (2.2) where each IFE shape function is defined by 18 coefficients: $a_j^s, b_j^s, c_j^s, d_j^s, a_3^s$ for $j = 1,2$ and $s = +,-$ that are determined by imposing: *(i)* continuity of the velocity across $\overline{\mathbf{DE}}$, *(ii)* weak continuity of the normal stress across $\overline{\mathbf{DE}}$, *(iii)* continuity of the divergence of the velocity, *(iv)* Lagrange and scaling conditions. These constraints yield a linear system which has a unique solution [27].

On every non-interface element T we use standard finite element shape functions $\boldsymbol{\Psi}_i,\ i = 1,2,\dots,9$. Shape functions on all the interface elements and non-interface elements are then used together to construct the global IFE functions on Ω, at time t using the approximate interface $\tilde{\Gamma}(t)$, as follows

$$S_h(\Omega,t) = \{\mathbf{U}_h \mid \mathbf{U}_h|_T \in X_h(T)\}, \tag{2.3}$$

where

$$X_h(T) = \begin{cases} \mathrm{span}\{\boldsymbol{\Phi}_i(x,y), i = 1,2,\ldots,9\}, & \text{if } T \in \mathcal{T}_h^i, \\ \mathrm{span}\{\boldsymbol{\Psi}_i(x,y), i = 1,2,\ldots,9\}, & \text{if } T \in \mathcal{T}_h^n. \end{cases}$$

When $\boldsymbol{\sigma} \neq \mathbf{0}$ we need a set of 2 particular IFE shape functions to capture the non-homogeneous surface force term. On each interface element T, the particular IFE shape functions Υ_1 and Υ_2 are of the form (2.2) and satisfy: *(i)* continuity of the velocity across $\overline{DE}$, *(ii)* weak jump of the normal stress across $\overline{DE}$, *(iii)* continuity of the divergence of the velocity and *(iv)* homogeneous Lagrange and scaling conditions. Again, these constraints yield a linear system with a unique solution [27].

The IFE solution $\binom{\mathbf{u}_h}{p_h}$ of Stokes interface problem with a nonzero surface force $\boldsymbol{\sigma} \neq \mathbf{0}$, is sought in the set $S_h(\Omega, t) + \{\mathbf{q}_h(x,y,t)\}$, where

$$\mathbf{q}_h(x,y,t) = \begin{cases} s_1\Upsilon_1(x,y) + s_2\Upsilon_2(x,y), & \text{on T } \in \mathcal{T}_i \\ 0 & \text{elsewhere.} \end{cases} \tag{2.4}$$

Here $s_j = \int_{\overline{DE}} \sigma_{h,j} ds$ and $\boldsymbol{\sigma}_h = (\sigma_{h,1}, \sigma_{h,2})^T$ is the linear interpolation of $\boldsymbol{\sigma}$ defined by $\boldsymbol{\sigma}_h(E) = \boldsymbol{\sigma}(E)$ and $\boldsymbol{\sigma}_h(D) = \boldsymbol{\sigma}(D)$.

3. Axisymmetric Stokes interface problem

In this section, we extend the Q_1/Q_0 IFE functions for solving the three-dimensional Stokes interface problem with axisymmetry around the z-axis. That is, in the cylindrical coordinate system (r, θ, z), the θ-component of fluid velocity and all θ-derivatives vanish. The fluid domain is then project onto Ω in the r-z meridian plane, which is separated by the projected interface $\Gamma(t)$ into two subdomains $\Omega^{\pm}(t)$ containing two fluids with viscosities $\nu^{\pm}$. Thus the three-dimensional problem reduces to a two-dimensional problem in the meridian plane. As in Section 2, $\Gamma(t)$ is assumed to satisfy the following ODE

$$\begin{cases} \dfrac{d}{dt}\mathbf{X} & = \mathbf{u}(\mathbf{X}, t), \quad t \in [0, T], \quad \forall\, \mathbf{X} \in \Gamma(t), \\ \mathbf{X}|_{t=0} & = \mathbf{X}_0, \qquad \text{with } \mathbf{X}_0 \in \Gamma(0), \end{cases} \tag{3.1a}$$

where $\mathbf{u} = (u_r, u_z)^T$ is the solution to the transient Stokes interface problem:

$$-\nabla \cdot \mathbf{S}(\mathbf{u}(r,z,t), p(r,z,t)) = \mathbf{0}, \qquad \text{in } \Omega^+(t) \cup \Omega^-(t), \tag{3.1b}$$

$$\nabla \cdot \mathbf{u}(r,z,t) = 0, \qquad \text{in } \Omega^+(t) \cup \Omega^-(t), \tag{3.1c}$$

$$\mathbf{u}(r,z,t) = \mathbf{g}(r,z,t), \text{ on } \partial\Omega, \tag{3.1d}$$

where $\mathbf{g} = (g_r, g_z)^T$ is the Dirichlet boundary condition. $\mathbf{S}$ is the stress tensor defined in three-dimensional space

$$\mathbf{S}(\mathbf{u}, p) = \nu\boldsymbol{\epsilon}_3(\mathbf{u}) - p\mathbf{I}_3, \tag{3.1e}$$

where

$$\boldsymbol{\epsilon}_3(\mathbf{u}) = \nabla\mathbf{u} + (\nabla\mathbf{u})^T \tag{3.1f}$$

and $\mathbf{I}_3$ is the 3×3 identity matrix. For axisymmetric problems, the gradient of a vector $\mathbf{v} = (v_r, v_z)^T$ is

$$\nabla\mathbf{v} = \begin{pmatrix} \dfrac{\partial v_r}{\partial r} & 0 & \dfrac{\partial v_r}{\partial z} \\ 0 & \dfrac{v_r}{r} & 0 \\ \dfrac{\partial v_z}{\partial r} & 0 & \dfrac{\partial v_z}{\partial z} \end{pmatrix} \tag{3.1g}$$

and the divergence of a vector is

$$\nabla \cdot \mathbf{v} = \frac{1}{r}\frac{\partial(r v_r)}{\partial r} + \frac{\partial v_z}{\partial z}. \tag{3.1h}$$

The jump conditions across the projected interface $\Gamma(t)$ become

$$[\mathbf{u}(r,z,t)]|_{\Gamma(t)} = \mathbf{0}, \tag{3.1i}$$

$$[\mathbf{S}_2(\mathbf{u}(r,z,t), p(r,z,t))\mathbf{n}]|_{\Gamma(t)} = \boldsymbol{\sigma}(r,z,t), \tag{3.1j}$$

where $\mathbf{n} = (n_r, n_z)^T$ is the unit vector normal to the interface $\Gamma(t)$ and $\boldsymbol{\sigma} = (\sigma_r, \sigma_z)^T$ is the surface force. $\mathbf{S}_2$ is the two-dimensional stress tensor defined as

$$\mathbf{S}_2(\mathbf{u}, p) = \nu\boldsymbol{\epsilon}_2(\mathbf{u}) - p\mathbf{I}_2, \tag{3.1k}$$

where $\mathbf{I}_2$ is the 2×2 identity matrix and

$$\boldsymbol{\epsilon}_2(\mathbf{u}) = \begin{pmatrix} 2\dfrac{\partial u_r}{\partial r} & \dfrac{\partial u_r}{\partial z} + \dfrac{\partial u_z}{\partial r} \\ \dfrac{\partial u_r}{\partial z} + \dfrac{\partial u_z}{\partial r} & 2\dfrac{\partial u_z}{\partial z} \end{pmatrix}. \tag{3.1l}$$

As in Section 2, we approximate the interface $\Gamma(t)$ using a set of control points $\mathcal{X}(t)$ to obtain a piecewise linear approximate interface $\tilde{\Gamma}(t)$. Since the solution to the Stokes interface problem is not smooth, we propose to extend the IFE functions described in Section 2 to the axisymmetric Stokes interface problem (3.1) for capturing the solution singularities. The detailed construction of the axisymmetric IFE spaces is given in Appendix.

4. IFE methods for Stokes interface problem

In this section, we present the discontinuous Galerkin immersed finite element (DG-IFE) methods for solving the Stokes interface problem. Note that, due to its complexity, the Stokes problem has to be solved with suitably chosen finite element spaces and special finite element functions have to be used in order to get satisfactory approximations. Here we consider the Q_1/Q_0 finite element functions with discontinuous Galerkin methods used in [47] . This choice is motivated by two major reasons:

- The primal finite element method with the Q_1/Q_0 FE space is not stable [48], while the discontinuous Galerkin method leads to a stable formulation [47].
- The discontinuous Galerkin formulation can naturally handle the discontinuity of the global IFE functions across edges of interface elements.

4.1. DG-IFE scheme for two-dimensional problem

The DG-IFE method for the two-dimensional Stokes interface problem (2.1b), (2.1c), (2.1d) consists of finding $\begin{pmatrix}\mathbf{u}_h\\ p_h\end{pmatrix} \in S_h(\Omega, t) + \{\mathbf{q}_h(x, y, t)\}$ such that

$$\begin{cases} A(\mathbf{u}_h, \mathbf{v}_h) + B(\mathbf{v}_h, p_h) & = L_h(\mathbf{v}_h) \\ B(\mathbf{u}_h, q_h) & = l(q_h), \end{cases} \quad \forall \begin{pmatrix}\mathbf{v}_h\\ q_h\end{pmatrix} \in S_{h,0}(\Omega, t), \tag{4.1a}$$

subject to the boundary condition

$$\mathbf{u}_h(R) = \mathbf{g}(R), \quad \text{for all mesh vertices } R \in \partial\Omega,$$

where

$$A(\mathbf{w}, \mathbf{v}) = \int_\Omega \nu \boldsymbol{\epsilon}(\mathbf{w}) : \nabla \mathbf{v} dx - \sum_{e\in\mathcal{E}_h} \int_e \nu\{\boldsymbol{\epsilon}(\mathbf{w})\mathbf{n}\} \cdot [\mathbf{v}] ds$$

$$- \sum_{e\in\mathcal{E}_h} \int_e \{\nu\boldsymbol{\epsilon}(\mathbf{v})\mathbf{n}\} \cdot [\mathbf{w}] ds + \sum_{e\in\mathcal{E}_h} \frac{\alpha}{h_e} \int_e \nu[\mathbf{u}] \cdot [\mathbf{v}] ds, \tag{4.1b}$$

$$B(\mathbf{v}, q) = -\int_\Omega q \nabla \cdot \mathbf{v} dx + \sum_{e\in\mathcal{E}_h} \int_e \{q\}[\mathbf{v}] \cdot \mathbf{n} ds, \tag{4.1c}$$

$$l(q) = \sum_{e\subset\partial\Omega} \int_e \{q\}[\mathbf{g}] \cdot \mathbf{n} ds, \tag{4.1d}$$

$$L_h(\mathbf{v}) = \sum_{T\in\mathcal{T}_h^i} \int_{\overline{DE}} \boldsymbol{\sigma}_h \cdot \mathbf{v} ds - \sum_{e\subset\partial\Omega} \int_e \{\nu\boldsymbol{\epsilon}(\mathbf{v})\mathbf{n}\} \cdot [\mathbf{g}] ds + \sum_{e\subset\partial\Omega} \frac{\alpha}{h_e} \int_e \nu[\mathbf{g}] \cdot [\mathbf{v}] ds. \tag{4.1e}$$

Here $\mathcal{E}_h$ is the collection of all edges in $\mathcal{T}_h$, α is a positive constant which is chosen to be 10 in the following simulations, h_e is the edge length, $\{\cdot\}$ and $[\cdot]$ denote the average and jump of quantities across element edge, and

$$S_{h,0}(\Omega, t) = \{\begin{pmatrix}\mathbf{v}_h\\ q_h\end{pmatrix} \in S_h(\Omega, t) : \ \mathbf{v}_h(R) = \mathbf{0}, \quad \text{for all mesh vertices } R \in \partial\Omega\}.$$

More details can be found in [27].

4.2. DG-IFE scheme for axisymmetric problem

The DG-IFE method for the axisymmetric Stokes interface problem consists of finding $\begin{pmatrix}\mathbf{u}_h\\ p_h\end{pmatrix} \in S_h^s(\Omega, t) + \{\mathbf{q}_h^s(r, z, t)\}$such that

$$\begin{cases} A(\mathbf{u}_h, \mathbf{v}_h) + B(\mathbf{v}_h, p_h) & = L_h(\mathbf{v}_h) \\ B(\mathbf{u}_h, q_h) & = l(q_h), \end{cases} \quad \forall \begin{pmatrix}\mathbf{v}_h\\ q_h\end{pmatrix} \in S_{h,0}^s(\Omega, t), \tag{4.2a}$$

subject to the boundary condition

$$\mathbf{u}_h(R) = \mathbf{g}(R), \quad \text{for all mesh vertices } R \in \partial\Omega,$$

where

$$A(\mathbf{w},\mathbf{v}) = \int_\Omega \nu \boldsymbol{\epsilon}_s(\mathbf{w}) : \nabla \mathbf{v} r dr dz - \sum_{e\in\mathcal{E}_h}\int_e \nu\{\boldsymbol{\epsilon}_2(\mathbf{w})\mathbf{n}\}\cdot[\mathbf{v}]r ds$$
$$- \sum_{e\in\mathcal{E}_h}\int_e \{\nu\boldsymbol{\epsilon}_2(\mathbf{v})\mathbf{n}\}\cdot[\mathbf{w}]r ds + \sum_{e\in\mathcal{E}_h}\frac{\alpha}{h_e}\int_e \nu[\mathbf{w}]\cdot[\mathbf{v}]r ds, \tag{4.2b}$$

$$B(\mathbf{v},q) = -\int_\Omega q\nabla\cdot\mathbf{v} r dr dz + \sum_{e\in\mathcal{E}_h}\int_e \{q\}[\mathbf{v}]\cdot\mathbf{n} r ds, \tag{4.2c}$$

$$l(q) = \sum_{e\subset\partial\Omega}\int_e \{q\}[\mathbf{g}]\cdot\mathbf{n} r ds, \tag{4.2d}$$

$$L_h(\mathbf{v}) = \sum_{T\in\mathcal{T}_h^i}\int_{\overline{DE}} \boldsymbol{\sigma}_h\cdot\mathbf{v} r ds - \sum_{e\subset\partial\Omega}\int_e\{\nu\boldsymbol{\epsilon}_2(\mathbf{v})\mathbf{n}\}\cdot[\mathbf{g}]r ds + \sum_{e\subset\partial\Omega}\frac{\alpha}{h_e}\int_e \nu[\mathbf{g}]\cdot[\mathbf{v}]r ds, \tag{4.2e}$$

and

$$S_{h,0}^s(\Omega,t) = \{\begin{pmatrix}\mathbf{v}_h\\ q_h\end{pmatrix} \in S_h^s(\Omega,t) : \ \mathbf{v}_h(R) = \mathbf{0}, \ \text{for all mesh vertices } R\in\partial\Omega\}.$$

Remark. The DG-IFE methods (4.1)–(4.2) were studied in [27,49], where it was shown numerically that the velocity is second order convergent and the pressure is first order convergent. These results correspond to the order of convergence obtained in [10] where fitted meshes are used. It is worthwhile to note that the schemes presented here correspond to the standard symmetric interior penalty Galerkin (SIPG) scheme [47]. Another popular scheme, referred to as nonsymmetric interior penalty Galerkin (NIPG) scheme, was also studied in [27,49], where it was shown that NIPG possesses the same order of convergence.

4.3. Moving interface

Since the interface evolves in time according to the ODEs (3.1a)–(2.1a), we track the interface using approximate velocity $\mathbf{u}_h$ and control points (a.k.a. markers) $\mathcal{X}^{(i)}(t)$, $1 \le i \le N$. The tracking procedure can therefore be written as

$$\begin{cases} \dfrac{d}{dt}\mathcal{X}^{(i)}(t) & = \mathbf{u}_h(\mathcal{X}^{(i)}(t),t), \quad t\in[0,T] \\ \mathcal{X}^{(i)}(0) & = \mathcal{X}_0^{(i)}, \end{cases} \quad i = 1,2,\dots,N, \tag{4.3}$$

where $\mathbf{u}_h(\mathcal{X}^{(i)}(t),t)$ is the IFE velocity of the fluid at the control point $\mathcal{X}^{(i)}(t)$ and N denotes the total number of control points.

To solve the Stokes problem with a moving interface, we use the following algorithm:

- *Select a time step Δt and the number of control points N*
- *Determine the initial control points $\mathcal{X}(0)$*
- *Construct an initial uniform mesh for Ω.*
- *For $k = 0, 1, 2, \dots$, loop until a stopping criterion is met.*
    1. *If needed, refine the mesh near the interface.*
    2. *Based on $\mathcal{X}(t_k)$, where $t_k = k\Delta t$, compute the surface force $\boldsymbol{\sigma}$ at the interface points, and construct the IFE basis functions and particular functions.*
    3. *Solve Stokes problem and determine the velocity $\mathbf{u}_h(\mathcal{X}^{(i)}(t_k), t_k)$ at the control points $\mathcal{X}^{(i)}(t_k)$, $i = 1, 2, \dots, N$.*
    4. *Update the control points $\mathcal{X}(t_{k+1})$ by integrating ODE (4.3) numerically. Check the distance between control points and redistribute if necessary.*

It should be noted that Δt and N can also be chosen adaptively, based on the CFL condition and the morphology of interface.

5. Numerical simulations

In this section, we present three simulation examples: drop retraction, drop deformation in shear flow, and drop deformation in extensional flow. The first two are in two dimensions and the last one is axisymmetric three dimensions. It should be noted that it is not mandatory to use adaptive mesh in our method. However, to better resolve the interfacial flow without dramatically increasing the number of unknowns, the mesh is locally refined near the interface. The procedure

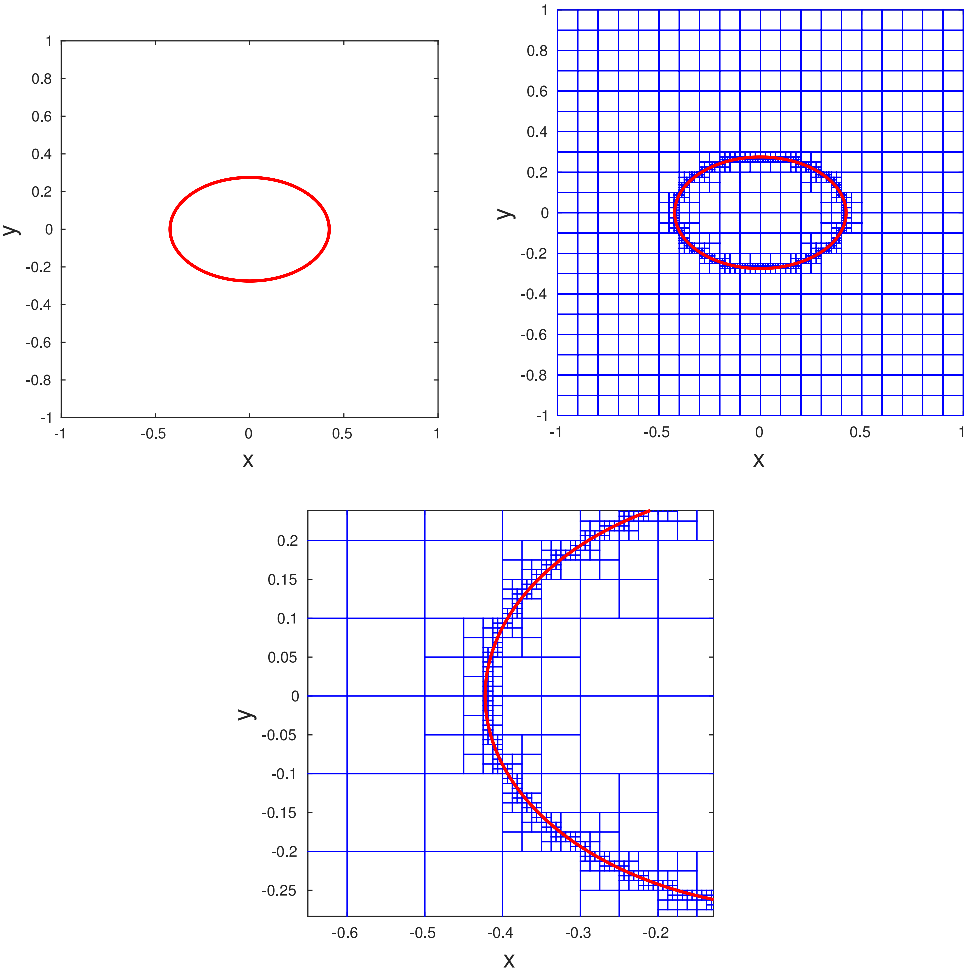


Fig. 2. Computational setup for Example 5.1: initial state (upper left), computational mesh (upper right), and a close view of the mesh near the interface (bottom).

consists of subdividing every interface element into four congruent elements, and may be carried out several times [50]. In this manuscript, we use at most five levels of refinement near the interface.

The surface force $\boldsymbol{\sigma}$ is computed using the following expression [15,51,52]

$$\boldsymbol{\sigma}(\mathbf{X}, t) = \sigma\,(\kappa_1(\mathbf{X}, t) + \kappa_2(\mathbf{X}, t))\,\mathbf{n},$$

where $\mathbf{n}$ is a unit normal vector to the interface at the point $\mathbf{X}$, $\sigma > 0$ is the surface tension, and $\kappa_1(\mathbf{X}, t)$ and $\kappa_2(\mathbf{X}, t)$ are the principal curvatures of the interface. In this work, κ_1 at each control point is computed using the quadratic interpolation based on that control point and its two neighbors. In two-dimensional problems, $\kappa_2(\mathbf{X}, t) = 0$; in axisymmetric problems, $\kappa_2(\mathbf{X}, t) = \frac{1}{L_1}$, where L_1 is the distance between X and where the line normal to the interface at X intersects the z-axis. Initially, the control points are uniformly distributed along the interface with spacing approximately equal to the cell size at the interface. No redistribution of the control points is performed except in Example 5.3.

Example 5.1. Drop retraction.

It is well known that, under surface tension, drops tend to retract to a spherical shape in 3D or circular shape in 2D. In this experiment, we consider the square domain $[-1, 1]^2$ and an initially elliptical drop centered at the origin with major axis $a = \sqrt{3}/4.1$ and minor axis $b = \sqrt{3}/6.3$. The fluid viscosity is $\nu^- = 1$ inside and outside the drop. The surface tension is chosen to be $\sigma = 2$. We use a mesh with approximately 3000 elements, obtained by partitioning the domain using a 20 × 20 background mesh and using four levels of local refinement applied on each interface element as shown in Fig. 2. Therefore the coarsest element size is $\frac{1}{20}$ and the finest element size is $\frac{1}{320}$. We use homogeneous velocity boundary conditions and zero body forces so that the motion of the interface is solely driven by surface tension.

The Stokes interface problem is solved using the DG-IFE method (4.1) at every time step t_k in order to obtain the approximate velocity $\mathbf{u}_h(\mathcal{X}_k^{(i)}, t_k)$, $i = 1, 2, \dots, N$ at the control points. In this simulation, we use Euler's method for (4.3) with a fixed time step $\Delta t = 2 \times 10^{-3}$.

We show the DG-IFE pressure and the mesh at $t = 0, 0.2, 2.102$ in Fig. 3. We stop the simulation once the L^2-norm of the velocity, $\|\mathbf{u}_h\| = \left(\int_\Omega |\mathbf{u}_h|^2 d\Omega\right)^{\frac{1}{2}}$, drops below 10^{-3}, as shown in Fig. 4. This is achieved when the drop is nearly circular at $t = 2.102$, i.e., after 1051 time steps. Since we are dealing with incompressible flows, the volume of the drop should remain unchanged with respect to time. This has also been validated in Fig. 4 where the normalized volume V/V_0 (V_0 is the initial volume of the drop) is shown. Furthermore, the steady state pressure shown in Fig. 3(f) satisfies the Young–Laplace equation

$$[p]|_\Gamma = \frac{\sigma}{R}, \tag{5.1}$$

where R is the radius of the drop. The surface tension $\sigma = 2$ is used in this experiment, and the radius of the steady state drop is $R = 0.3408$. According to (5.1), the jump of the pressure across the interface should be $[p]|_\Gamma = 5.8686$. For the example demonstrated in Fig. 3(f), the pressure on each subdomain is constant and the jump of the DG-IFE pressure across the interface is $[p]|_\Gamma = 5.856$ which agrees with the expected value.

Example 5.2. Drop deformation in shear flow

We validate our DG-IFE method by studying the behavior of a drop in shear flow. The domain of interest consists of a drop with initial radius $a = \frac{1}{4}$ centered in a rectangle $\Omega = [-\pi, \pi] \times [-1, 1]$ and subjected to a shear flow, as illustrated in Fig. 5. The boundary conditions are defined as $\mathbf{u} = \mathbf{g} = (\dot{\epsilon} y, 0)^T$ on $\partial\Omega$ where $\dot{\epsilon}$ is the shear rate. We define the capillary parameter $Ca = \frac{\nu a \dot{\epsilon}}{\sigma}$ where ν is the viscosity of the drop and σ is the surface tension. The deformation of the drop depends strongly on Ca and was extensively studied in [53]. The deformation is computed using the following formula:

$$D = \frac{L - W}{L + W}, \tag{5.2}$$

where L is the length of the deformed drop and W is its width, as illustrated in Fig. 6.

We use a 90×50 background mesh with 5 levels of refinement such that the largest mesh size is $\frac{\pi}{45}$ and the smallest mesh size is $\frac{1}{800}$, which amounts to no more than 8200 elements. The simulations are carried out using the DG-IFE method (4.1) and the embedded Runge–Kutta method of orders 2 and 3 is used for (4.3). A typical time step is around 5×10^{-3}. We choose $\nu^- = \nu^+ = \nu$ and $\sigma = 1$, and test different capillary numbers $Ca = \frac{3}{32}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}$ by varying $\dot{\epsilon}$ and ν.

We plot the evolution of the deformation D for different values of Ca in Fig. 7(a). We also plot the steady state deformation D versus the capillary number in Fig. 7(b), and compare our results to those obtained by Zhou and Pozrikidis [53] and Yue et al. [54] Fig. 7(b) shows excellent agreement. However, the simulations in [54] were carried out using the phase-field method discretized by the spectral methods on a much finer Cartesian mesh that has 2048×1024 grid points or more than 2 million elements.

Example 5.3. Drop deformation in extensional flow

We validate our method by studying the behavior of a three-dimensional drop subjected to a uniaxial extensional flow, as shown in Fig. 8. The flow is axisymmetric about the z-axis and symmetric with respect to the xy-plane. Therefore, we can reduce it to a two-dimensional problem in the upper half of the meridian plane. We use the same parameters as in [52] with initial drop radius R_0 and domain size $H = W = 10R_0$. The boundary conditions are $\mathbf{u}(r, z) = (u_r, u_z) = (-0.5\dot{\epsilon} r, \dot{\epsilon} z)$ along $r = W$ and $z = H$, where $\dot{\epsilon} > 0$ is the extension rate. On the boundaries defined by $r = 0$ and $z = 0$, we impose symmetry conditions:

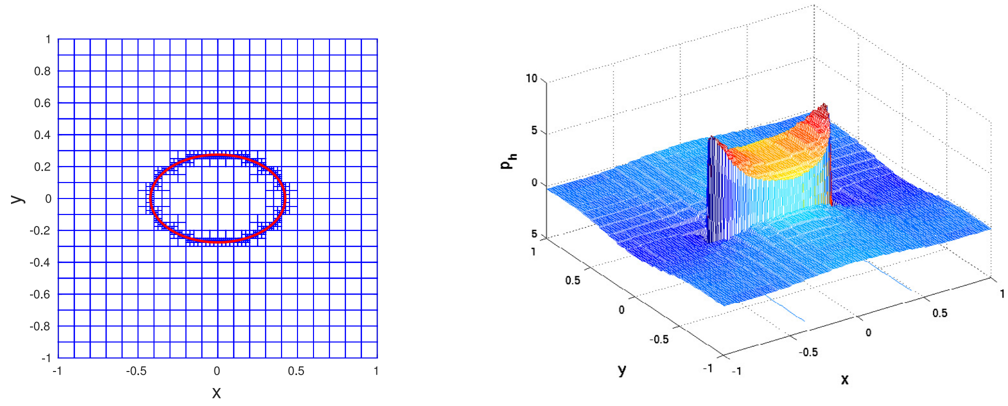
$$\begin{cases} u_r = 0, \ \dfrac{\partial}{\partial r} u_z = 0, & \text{if } r = 0 \\ u_z = 0, \ \dfrac{\partial}{\partial z} u_r = 0, & \text{if } z = 0. \end{cases}$$

We denote by ν^+ and ν^- the viscosities of the matrix (*i.e.* the fluid outside the drop) and the drop, respectively. The capillary number is then defined as $Ca = \frac{\nu^+ \dot{\epsilon} R_0}{\sigma}$ and the viscosity ratio is defined as $\beta = \frac{\nu^-}{\nu^+}$.

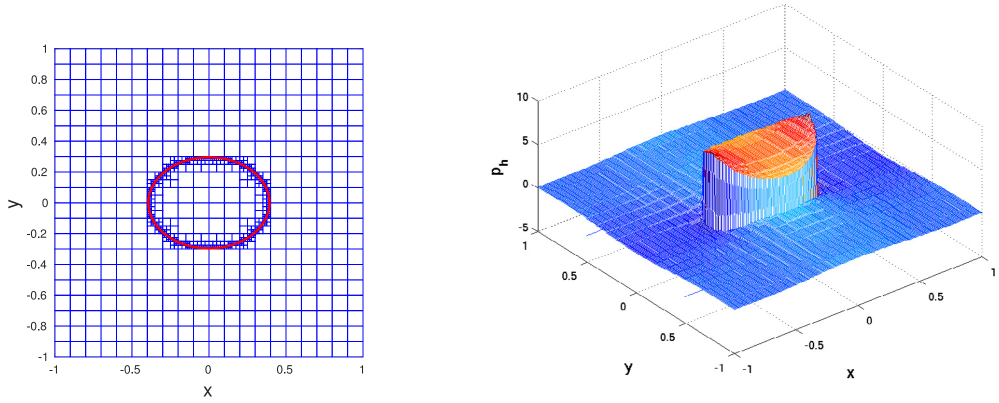
The simulation is carried out using the algorithm described in Section 4.3 with the DG-IFE method (4.2). The explicit trapezoidal rule with uniform time step $\Delta t \dot{\epsilon} = 5 \times 10^{-3}$ is used to integrate (4.3). The computational mesh is obtained from a 61×61 uniform background mesh with four levels of local refinement; this amounts to at most 4500 elements.

The interface shapes and computational meshes at different time instants for $Ca = 0.1$ and $\beta = 0.5$ are plotted in Figs. 9 and 10, respectively. The transient behavior of drop deformation is investigated by plotting the time history of $\frac{L}{R_0}$ in Fig. 11, where L is the length of the deformed drop shown in Fig. 8. The comparison shows good agreement with [52] for $t\dot{\epsilon} < 1.5$. However, spurious oscillations appear at later times.

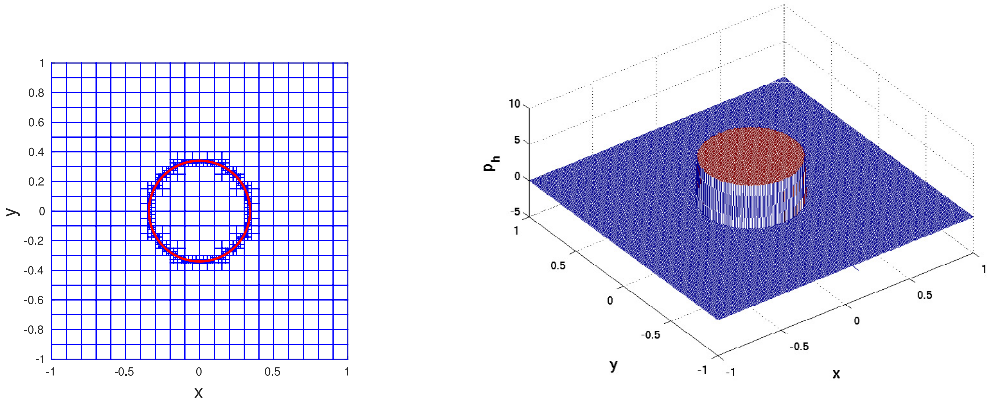
After some investigation, we find that these spurious oscillations are caused by the clustering of the control points near the upper tip of the drop as shown in Fig. 12, which makes the curvature calculation very sensitive to errors in the control point locations. Actually, it is well-known that decreasing the distance between control points does not necessarily improve



(a) $t = 0$



(b) $t = 0.2$



(c) $t = 2.102$ (steady state)

Fig. 3. Simulation of Example 5.1. Left column shows the interface and the computational mesh; right column shows the pressure.

accuracy in front-tracking methods [12]. The optimal distance between control points is on the same order as h, the cell size at the interface [12,55]. Following [55], we monitor the distance between control points after each interface update. If the

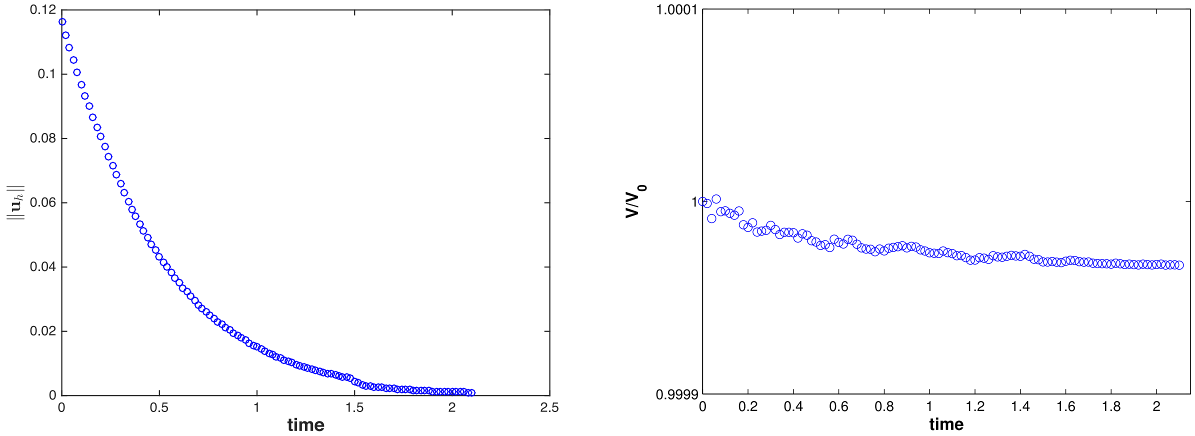


Fig. 4. Velocity norm $\|\mathbf{u}_h\|$ of the system versus time (left). Normalized volume V/V_0 of the drop versus time (right).

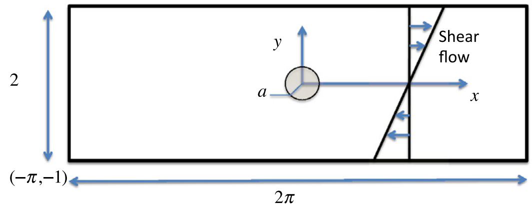


Fig. 5. The computational domain for drop deformation in shear flow.

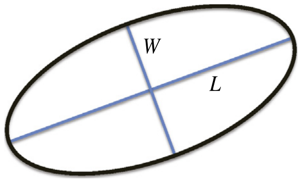
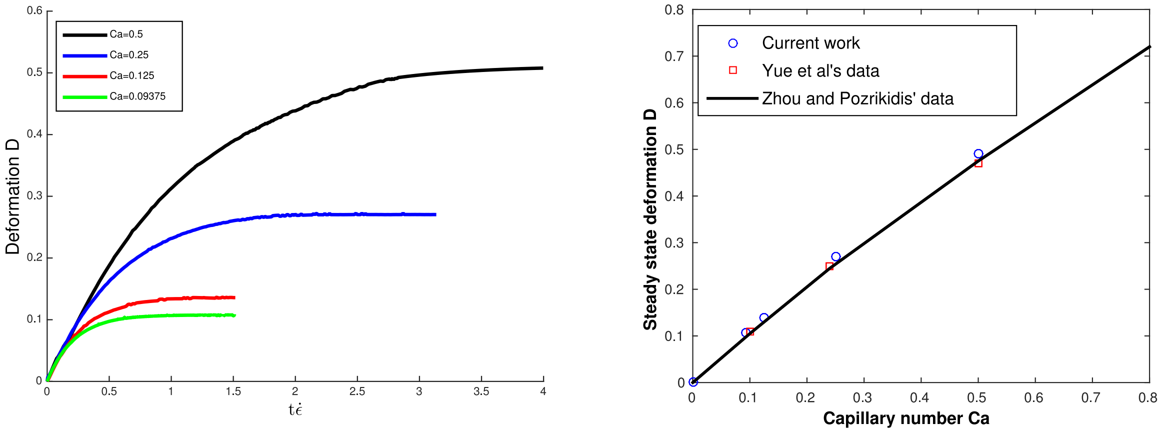


Fig. 6. The length L and width W of a deformed drop.



(a) Drop deformation D versus dimensionless time.

(b) Steady state deformation D versus the Capillary number Ca.

Fig. 7. Simulation of Example 5.2.

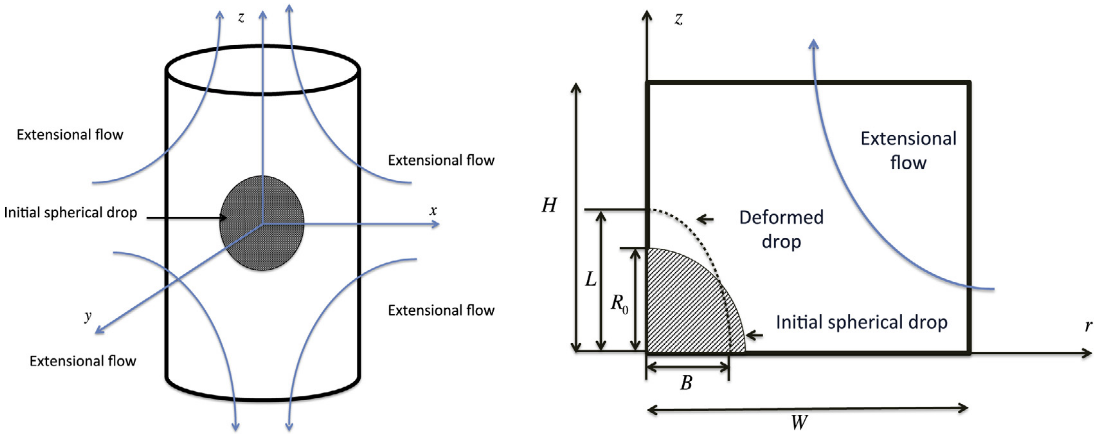


Fig. 8. The computational domain for transient drop deformation under extensional flow (left) and its projection onto the meridian plane (right).

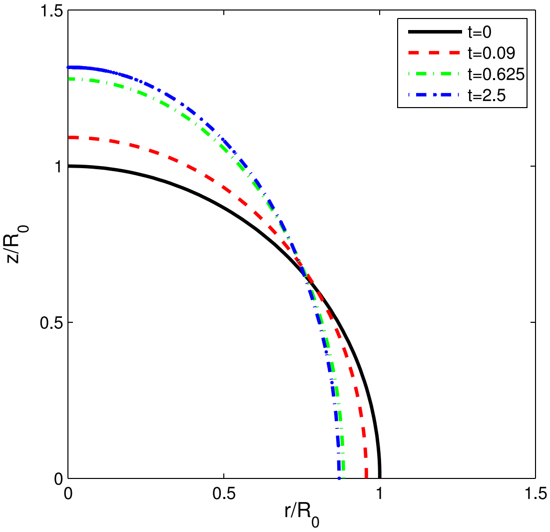


Fig. 9. Interface shapes for drop deformation in extensional flow. $Ca = 0.1$, $\beta = 0.5$.

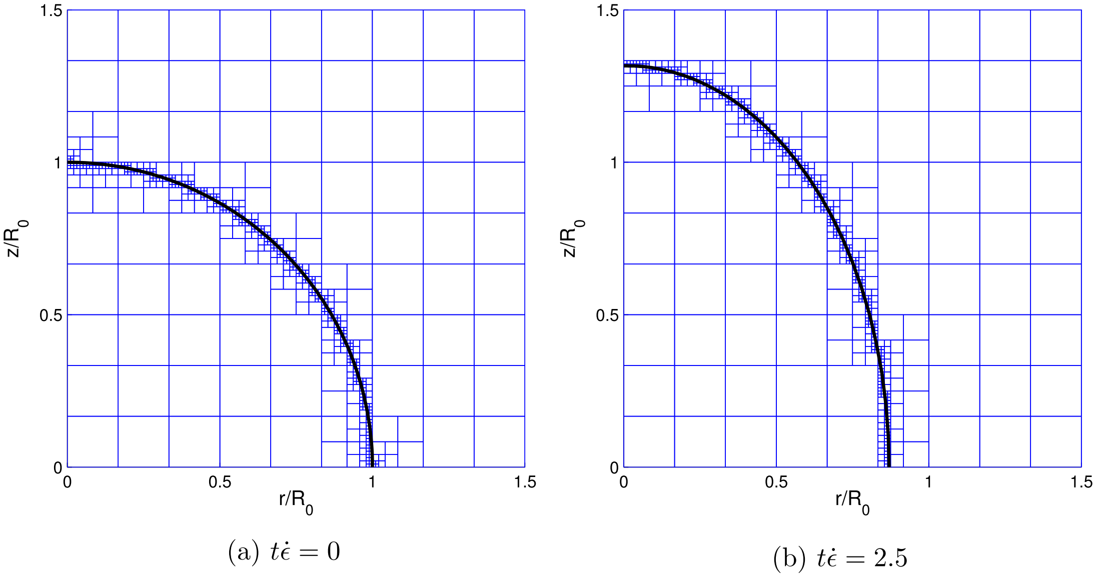
minimum distance is below $0.5h$ or the maximum distance is above $1.5h$, we redistribute the control points uniformly along a parametric spline reconstruction of the interface.

We then rerun the simulation for $\beta = 0.5$ and 1 using the same parameters as above. We plot the deformation versus time together with results obtained by Yue et al. [52] in Fig. 13, which shows excellent agreement without any spurious oscillations. It should be noted that Yue et al. used more than 6000 P1/P2 elements. We further verify the conservation of mass property in Fig. 14 by plotting the volume of the drop versus time to observe that the relative error of drop volume is below 0.05%. In terms of mass conservation, the current method outperforms the phase-field method whose mass loss is typically at the order of 1% depending on the interfacial thickness [52,56].

6. Conclusion

In this manuscript, we present a DG-IFE method for simulating Stokes interfacial flows. We use the Q_1/Q_0 finite element space, which has been shown in [27] to achieve second order convergence in velocity and first order convergence in pressure. We construct new IFE shape functions for the 3D axisymmetric Stokes problem and track the moving interface by a front-tracking technique. This extends the work in [27] that only dealt with the 2D Stokes problem with a fixed interface.

We demonstrate the capability of our DG-IFE method by computing three problems: drop retraction, drop deformation in shear flow, and drop deformation in extensional flow. The IFE basis functions are constructed such that the numerical solution automatically satisfies the velocity continuity condition and the stress jump condition (i.e., the Young–Laplace



(a) $t\dot{\epsilon} = 0$ (b) $t\dot{\epsilon} = 2.5$

Fig. 10. Close view of mesh near the interface for drop deformation in extensional flow. $Ca = 0.1$, $\beta = 0.5$.

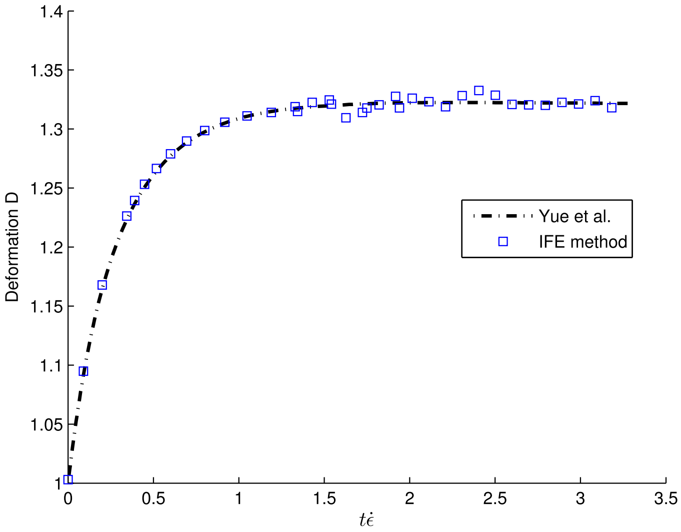


Fig. 11. Drop deformation in extensional flow versus time for Example 5.3.

equation) across the interface. As a result, we do not need to distribute the interfacial force into a narrow band like in many other fixed-mesh methods. This improves the numerical accuracy at the interface. Compared with the phase-field method, our proposed DG-IFE method has a better mass conservation property and requires much less degrees of freedom.

It should be noted that there is no principal difficulty to couple the DG-IFE method with other fixed-mesh methods such as volume-of-fluid and level-set methods, as long as we can identify the interface location inside each interface element. However, if the fluid equations are time-dependent, such as the transient Navier–Stokes equations, the time derivatives in the IFE shape functions have to be considered as well [45,57]. In the future, we intend to couple the DG-IFE method with the level-set method and extend it to Navier–Stokes moving interface problems.

Acknowledgments

This research was partially supported by the National Science Foundation, United States under DMS-1016313. PY also acknowledges the support by the National Science Foundation, United States under DMS-1522604.

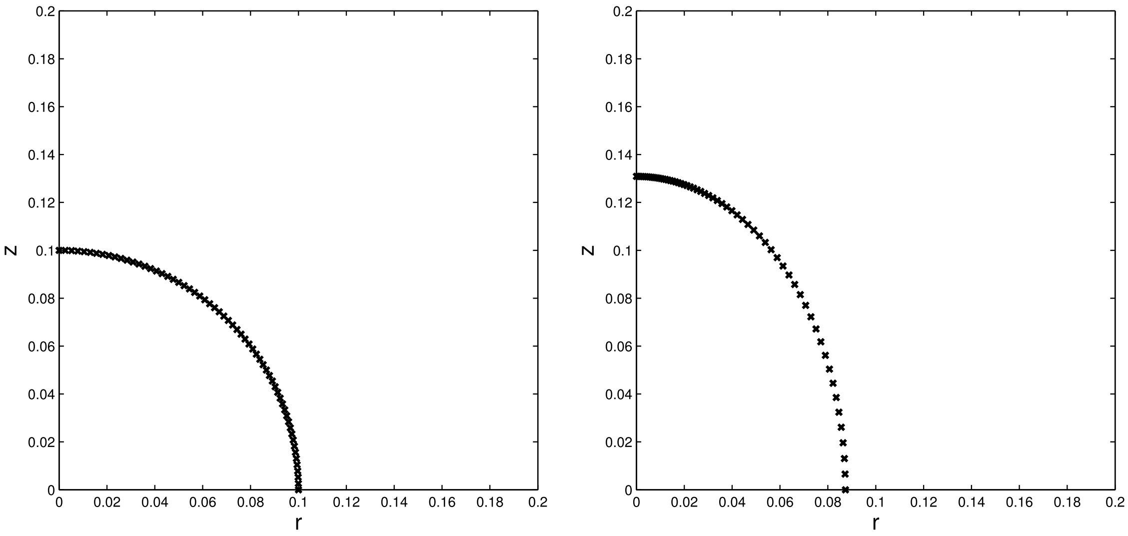


Fig. 12. Control points at $t = 0$ (left) and $t = 1$ (right) for Example 5.3 without redistribution.

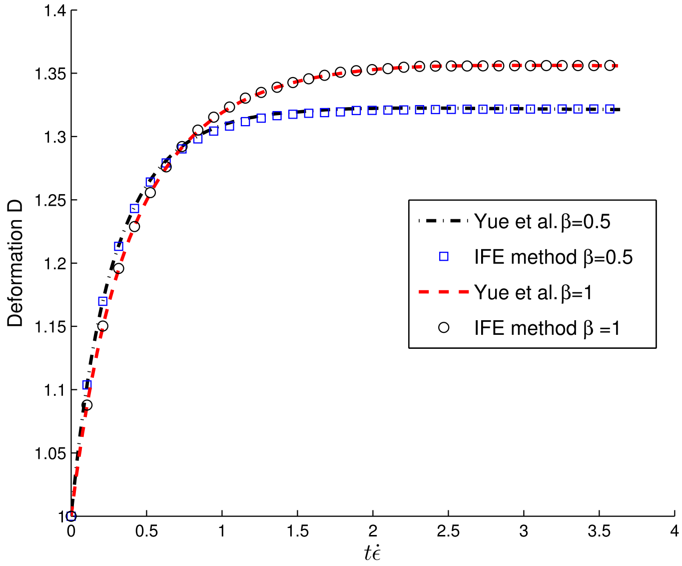


Fig. 13. Drop deformation in extensional flow versus time for Example 5.3 with control point redistribution.

Appendix. Axisymmetric IFE shape functions

We first construct axisymmetric IFE shape functions without interface force and then we include the interface force.

A.1. Q_1/Q_0 axisymmetric IFE shape functions without surface force

We start by describing a procedure to construct the IFE shape functions for the axisymmetric Stokes interface problem on a typical interface element $T = \square A_1A_2A_3A_4 \in \mathcal{T}_h^i$ with vertices $A_j = (r_j, z_j)$, $j = 1, 2, 3, 4$, at time t. We assume that the approximate interface $\tilde{\Gamma}(t)$ intersects T at $D = (r_D, z_D)$ and $E = (r_E, z_E)$, and the interface $\widetilde{DE} = \tilde{\Gamma}(t) \cap T$ is approximated by the line segment $\overline{DE}$ which separates T into two polygonal domains T^+ and T^- such that T^+ contains vertices of T that are in Ω^+. Topologically, there are two types of interface elements. Type I interface elements are those with two adjacent edges cut by the interface and Type II interface elements have two opposite edges cut by the interface.

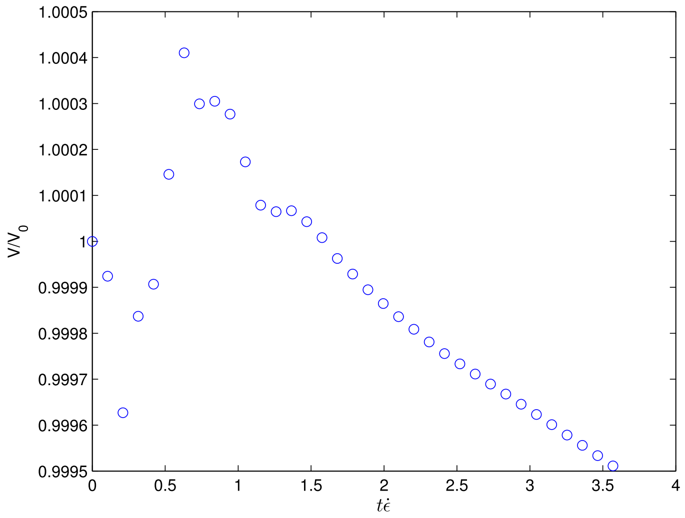


Fig. 14. Drop volume versus time for Example 5.3.

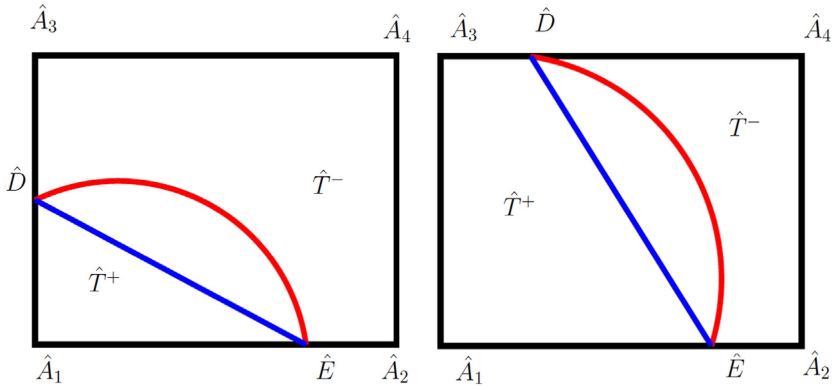


Fig. A.15. Reference interface elements of Type I (left) and Type II (right).

The construction of the finite element shape functions are performed on the reference element $\hat{T} = \Box\hat{A}_1\hat{A}_2\hat{A}_3\hat{A}_4$ with vertices $\hat{A}_1 = (0,0)^T$, $\hat{A}_2 = (1,0)^T$, $\hat{A}_3 = (0,1)^T$, $\hat{A}_4 = (1,1)^T$. Let $Y = (r,z)^T$ and $\hat{Y} = (\hat{r},\hat{z})$ and let

$$\hat{Y} = F(Y) = MY + B, \tag{A.1}$$

be the standard affine mapping from an arbitrary element T to the reference element $\hat{T}$ such that $\hat{A}_j = F(A_j)$, $j = 1,2,3,4$. We further note that each interface element T of Type I (Type II) is mapped into a reference element of Type I (Type II) shown in Fig. A.15 where $\hat{E} = F(E)$ and $\hat{D} = F(D)$ and $\overline{\hat{D}\hat{E}} = F(\overline{DE})$. The interface points $\hat{E}$ and $\hat{D}$ can be written as

$$\hat{D} = (0,\hat{d}), \quad \hat{E} = (\hat{e},0), \tag{A.2}$$

for an element of Type I and

$$\hat{D} = (\hat{d},1), \quad \hat{E} = (\hat{e},0), \tag{A.3}$$

for an element of Type II, where $0 < \hat{d},\hat{e} < 1$.

Now we are ready to describe our procedure for constructing the IFE shape functions on the reference element for the axisymmetric interface problem. As usual, a function $\hat{f}(\hat{r},\hat{z})$ defined for $(\hat{r},\hat{z}) \in \hat{T}$ leads to a function $f(r,z) = \hat{f}(F(r,z))$ for $(r,z) \in T$ by the affine mapping between the reference element $\hat{T}$ and element T. We first note that the velocity $\mathbf{u} = (u_r,u_z)^T$ and the pressure p are coupled through the jump condition (3.1j) which requires the design of vector-valued shape functions for both $\mathbf{u}$ and p. This means we plan to approximate the solution vector $\hat{\mathbf{U}} = \binom{\hat{\mathbf{u}}}{\hat{p}}$ by an IFE function $\hat{\boldsymbol{\Phi}}$ which is a piecewise polynomial vector function of the form

$$\hat{\boldsymbol{\Phi}}(\hat r,\hat z) = \hat{\boldsymbol{\Phi}}^s(\hat r,\hat z) = \begin{pmatrix}\hat\phi_1^s(\hat r,\hat z)\\ \hat\phi_2^s(\hat r,\hat z)\\ \hat\phi_3^s(\hat r,\hat z)\end{pmatrix}, \quad \text{for } (\hat r,\hat z)\in \hat T^s, \quad s=+,-, \tag{A.4a}$$

where

$$\hat\phi_j^s(\hat r,\hat z) = a_j^s + b_j^s\hat r + c_j^s\hat z + d_j^s\hat r\hat z, \;\; j=1,2, \;\; s=+,-, \tag{A.4b}$$

$$\hat\phi_3^s(\hat r,\hat z) = a_3^s, \;\; s=+,-. \tag{A.4c}$$

Next we let $\hat{\boldsymbol{\Theta}} = (\hat\phi_1,\hat\phi_2)^T$ and $\hat{\boldsymbol{\Theta}}|_{T^s}(\hat r,\hat z) = \hat{\boldsymbol{\Theta}}^s(\hat r,\hat z) = \begin{pmatrix}\hat\phi_1^s(\hat r,\hat z)\\ \hat\phi_2^s(\hat r,\hat z)\end{pmatrix}$, $s=+,-$.

We then discuss the construction of IFE shape functions that will be used to form the local IFE space on the reference element $\hat T$. According to (A.4) each IFE shape function is defined by 18 coefficients $a_j^s, b_j^s, c_j^s, d_j^s, j = 1{,}2, s = +,-$ and $a_3^s,\; s=+,-$. Hence, we can define IFE shape functions $\hat{\boldsymbol{\Phi}}_i, i = 1,2,\dots,9$ whose coefficients are uniquely determined by the following 18 conditions:

- continuity of the velocity component across $\overline{\hat D\hat E}$ for $\hat{\boldsymbol{\Theta}}_i = (\hat\phi_{1,i},\hat\phi_{2,i})^T$

$$\hat{\boldsymbol{\Theta}}_i^-(\hat E) = \hat{\boldsymbol{\Theta}}_i^+(\hat E), \quad \hat{\boldsymbol{\Theta}}_i^-(\hat D) = \hat{\boldsymbol{\Theta}}_i^+(\hat D), \quad \frac{\partial^2\hat{\boldsymbol{\Theta}}_i^-}{\partial\hat r\partial\hat z} = \frac{\partial^2\hat{\boldsymbol{\Theta}}_i^+}{\partial\hat r\partial\hat z} \tag{A.5a}$$

- weak continuity of the normal stress (3.1j) across $\overline{\hat D\hat E}$

$$\int_{\overline{\hat D\hat E}} [\mathbf{S}_2(\hat{\boldsymbol{\Theta}}_i,\hat\phi_{3,i})\cdot\mathbf{n}_{\overline{\hat D\hat E}}]ds = 0 \tag{A.5b}$$

$\mathbf{n}_{\overline{\hat D\hat E}} = (n_r,n_z)^T$ is a unit vector normal to the approximate interface $\overline{\hat D\hat E}$ and $\mathbf{S}_2$ is the stress tensor defined in (3.1k).

- continuity of the divergence of the velocity

$$\nabla\cdot\hat{\boldsymbol{\Theta}}_i^+\left(\frac{\hat D+\hat E}{2}\right) = \nabla\cdot\hat{\boldsymbol{\Theta}}_i^-\left(\frac{\hat D+\hat E}{2}\right) \tag{A.5c}$$

- Lagrange and scaling conditions

$$\hat\phi_{1,i}(\hat A_j) = \delta_{i,j}, \;\; \hat\phi_{2,i}(\hat A_j) = \delta_{i,j+4}, \; j=1,2,3,4, \text{ and } \frac{1}{|\hat T|}\int_{\hat T}\hat\phi_{3,i}dX = \delta_{i,9}. \tag{A.5d}$$

Since D and E do not lie on the z-axis simultaneously, the midpoint of $\hat D$ and $\hat E$ is used in Eq. (A.5c) to avoid the singularity associated with $1/r$ in the divergence. Note that the continuity of the second derivatives in (A.5a) is equivalent to $d_1^+ = d_1^-$ and $d_2^+ = d_2^-$ which by using $d_1^+ = d_1^- = d_1$ and $d_2^+ = d_2^- = d_2$ in (A.4a)–(A.4b) reduces the number of unknown coefficients to 16.

Conditions (A.5a)–(A.5d) lead to a linear system $\mathbf{M}\mathbf{c}_i = \mathbf{b}_i$ for the coefficients

$$\mathbf{c}_i = (a_1^+, b_1^+, c_1^+, d_1, a_1^-, b_1^-, c_1^-, a_2^+, b_2^+, c_2^+, d_2, a_2^-, b_2^-, c_2^-, a_3^-, a_3^+)^T.$$

The matrix **M** for a reference element of Type I is

$$\begin{pmatrix}
1 & \hat{e} & 0 & 0 & -1 & -\hat{e} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 0 & \hat{d} & 0 & -1 & 0 & -\hat{d} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \hat{e} & 0 & 0 & -1 & -\hat{e} & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \hat{d} & 0 & -1 & 0 & -\hat{d} & 0 & 0\\
0 & m_{52} & m_{53} & m_{54} & 0 & m_{56} & \nu^-\hat{e} & 0 & -\nu^+\hat{e} & 0 & m_{511} & 0 & \nu^-\hat{e} & 0 & -\hat{d} & \hat{d}\\
0 & 0 & m_{63} & m_{64} & 0 & 0 & \nu^-\hat{d} & 0 & -\nu^+\hat{d} & m_{610} & m_{611} & 0 & \nu^-\hat{d} & m_{614} & -\hat{e} & \hat{e}\\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_{1515} & m_{1516}\\
-1 & -\hat{e} & -\frac{\hat{d}}{2} & 0 & 1 & \hat{e} & \frac{\hat{d}}{2} & 0 & 0 & -\frac{\hat{e}}{2} & 0 & 0 & 0 & \frac{\hat{e}}{2} & 0 & 0
\end{pmatrix}, \quad \text{(A.6)}$$

where

$$m_{52} = -2\nu^+\hat{d},\;\; m_{53} = -\nu^+\hat{e},\;\; m_{54} = \frac{1}{2}(\nu^- - \nu^+)\left(\hat{e}^2 + 2\hat{d}^2\right), m_{56} = 2\nu^-\hat{d},$$

$$m_{511} = \frac{1}{2}(\nu^- - \nu^+)\hat{e}\hat{d},$$

$$m_{63} = -\nu^+\hat{d},\;\; m_{64} = \frac{1}{2}(\nu^- - \nu^+)\hat{e}\hat{d},\;\; m_{610} = -2\nu^+\hat{e},$$

$$m_{611} = \frac{1}{2}(\nu^- - \nu^+)\left(2\hat{e}^2 + \hat{d}^2\right), m_{614} = 2\nu^-\hat{e},$$

$$m_{1515} = \frac{1}{2}\hat{e}\hat{d},\;\; m_{1516} = 1 - \frac{1}{2}\hat{e}\hat{d}.$$

For a reference element of Type II, the matrix **M** is

$$\begin{pmatrix}
1 & \hat{e} & 0 & 0 & -1 & -\hat{e} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & d & 1 & 0 & -1 & -\hat{d} & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \hat{e} & 0 & 0 & -1 & -\hat{e} & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \hat{d} & 1 & 0 & -1 & -\hat{d} & -1 & 0 & 0\\
0 & -2\nu^+ & m_{53} & m_{54} & 0 & 2\nu^- & m_{57} & 0 & m_{59} & 0 & m_{510} & 0 & m_{513} & 0 & -1 & 1\\
0 & 0 & -\nu^+ & m_{64} & 0 & 0 & \nu^- & 0 & -\nu^+ & m_{610} & m_{611} & 0 & \nu^- & m_{614} & \hat{d}-\hat{e} & m_{616}\\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\hat{d}+\hat{e}}{2} & m_{1516}\\
-1 & -\hat{e} & -\frac{\hat{d}}{2} & 0 & 1 & \hat{e} & \frac{\hat{d}}{2} & 0 & 0 & -\frac{\hat{e}}{2} & 0 & 0 & 0 & \frac{\hat{e}}{2} & 0 & 0
\end{pmatrix}, \quad \text{(A.7)}$$

where

$$m_{53} = \nu^+(\hat{d} - \hat{e}),\; m_{54} = -\frac{1}{2}(\nu^- - \nu^+)\left(-2 + \hat{d}^2 - \hat{e}^2\right),\;\; m_{57} = \nu^-(-\hat{d} + \hat{e}),\;\; m_{5,9} = \nu^+(\hat{d} - \hat{e}),$$

$$m_{511} = -\frac{1}{2}(\nu^- - \nu^+)(\hat{d} - \hat{e}),\;\; m_{513} = \nu^-(-\hat{d} + \hat{e}),$$

$$m_{64} = \frac{1}{2}(\nu^- - \nu^+)(\hat{d} + \hat{e}), m_{610} = 2\nu^+(\hat{d} - \hat{e}),\;\; m_{611} = -\frac{1}{2}(\nu^- - \nu^+)\left(-1 + 2\hat{d}^2 - 2\hat{e}^2\right),$$

$$m_{614} = 2\nu^-(-\hat{d}+\hat{e}), m_{1516} = \frac{1}{2}(2-\hat{d}-\hat{e}).$$

The vector $\mathbf{b}_i$ for both Type I and Type II elements is

$$\mathbf{b}_i = (0, 0, 0, 0, 0, 0, \delta_{i,1}, \delta_{i,2}, \delta_{i,3}, \delta_{i,4}, \delta_{i,5}, \delta_{i,6}, \delta_{i,7}, \delta_{i,8}, \delta_{i,9}, 0)^T.$$

We note that the matrices $\mathbf{M}$ in (A.6) and (A.7) are quite similar to those used to construct the IFE shape functions for the two-dimensional Stokes interface problem, see [27,49] for more details, but the last row which corresponds to the conservation of mass constraint (3.6c) is different. This implies that the shape functions constructed in this section are essentially different from those described in Section 2 despite several similarities. Once these IFE shape functions are constructed on the reference element, the standard affine mapping is applied to obtain the corresponding vector IFE shape functions on an interface element T as $\mathbf{\Phi}_i(r, z) = \hat{\mathbf{\Phi}}_i(F(r, z)),\ \ i = 1, 2, \dots, 9$.

On every non-interface element T we use the standard finite element shape functions $\mathbf{\Psi}_i,\ i = 1, 2, \dots, 9$. Fig. A.16 presents illustrations for the shape functions $\mathbf{\Psi}_1$ and $\mathbf{\Phi}_1$. Unlike $\mathbf{\Psi}_1$ used in the standard Q_1/Q_0 finite element space, the components of the IFE shape function $\mathbf{\Phi}_1$ cannot be decoupled, *i.e.,* its second and third components are generally not zero.

Then, the two-dimensional shape functions defined above are used to construct the global IFE space on Ω for the axisymmetric Stokes interface problem, at time t, as follows:

$$S_h^s(\Omega, t) = \{\mathbf{U}_h \mid \mathbf{U}_h|_T \in X_h^s(T)\}, \tag{A.8}$$

where

$$X_h^s(T) = \begin{cases} \text{span}\{\mathbf{\Phi}_i(r, z), i = 1, 2, \dots, 9\}, & \text{if } T \in \mathcal{T}_h^i, \\ \text{span}\{\mathbf{\Psi}_i(r, z), i = 1, 2, \dots, 9\}, & \text{if } T \in \mathcal{T}_h^n. \end{cases}$$

A.2. Q_1/Q_0 particular axisymmetric IFE functions with surface force

In the case where the jump condition (3.1j) is such that $\boldsymbol{\sigma} \neq \mathbf{0}$, we use the same idea presented in Section 2 to construct the IFE particular functions. On each interface element T, the two particular IFE functions are defined as

$$\Upsilon_j = \begin{pmatrix} \mathbf{\Lambda}_j \\ \psi_j \end{pmatrix}, \ j = 1, 2,$$

whose velocity component is

$$\mathbf{\Lambda}_j(r, z) = \begin{cases} \mathbf{\Lambda}_j^+(r, z) & \text{on } T^+ \\ \mathbf{\Lambda}_j^-(r, z) & \text{on } T^-, \end{cases} \tag{A.9}$$

with

$$\mathbf{\Lambda}_j^s(r, z) = \begin{pmatrix} \Lambda_{1,j}^s(r, z) \\ \Lambda_{2,j}^s(r, z) \end{pmatrix}, \ \text{ on } T^s, \ \ s = +, -, \tag{A.10}$$

and the pressure component is a piecewise constant function such that

$$\psi_j = \begin{cases} \psi_j^+(r, z), & \text{on } T^+ \\ \psi_j^-(r, z), & \text{on } T^- \end{cases}, \ \ \psi_j^s(r, z) = a_3^s, \ \ s = +, -.$$

We then define the velocity component of the particular IFE shape functions to be piecewise polynomials

$$\Lambda_{i,j}^s(r, z) = a_{i,j}^s + b_{i,j}^s r + c_{i,j}^s z + d_{i,j}^s rz, \ \ j = 1, 2, \ \ i = 1, 2, \ \ s = +, -. \tag{A.11}$$

We further require that each particular IFE shape function Υ_j satisfies the following 18 constraints:

$$\mathbf{\Lambda}_j(A_i) = 0, \quad i = 1, 2, 3, 4, \tag{A.12a}$$

$$\mathbf{\Lambda}_j^-(E) = \mathbf{\Lambda}_j^+(E), \quad \mathbf{\Lambda}_j^-(D) = \mathbf{\Lambda}_j^+(D), \quad \frac{\partial^2 \mathbf{\Lambda}_j^-}{\partial r \partial z} = \frac{\partial^2 \mathbf{\Lambda}_j^+}{\partial r \partial z}, \tag{A.12b}$$

$$\int_{\overline{DE}} [\mathbf{S}_1(\mathbf{\Lambda}_j, \psi_j)\mathbf{n}_{\overline{DE}}] ds = \mathbf{e}_j, \tag{A.12c}$$

$$\frac{1}{|T|}\int_T \psi_j dX = 0, \ \ \nabla \cdot \mathbf{\Lambda}_j^+\left(\frac{D+E}{2}\right) = \nabla \cdot \mathbf{\Lambda}_j^-\left(\frac{D+E}{2}\right), \ \ j = 1, 2, \tag{A.12d}$$

where $\mathbf{n}_{\overline{DE}}$ is the unit normal vector used in (A.5b), $\mathbf{e}_j$ is the canonical vector in $\mathbb{R}^2$ and $\mathbf{S}_1$ is defined in (3.1k). Note that these conditions lead to a linear system for determining the parameters of a particular IFE function, and the matrices of this linear

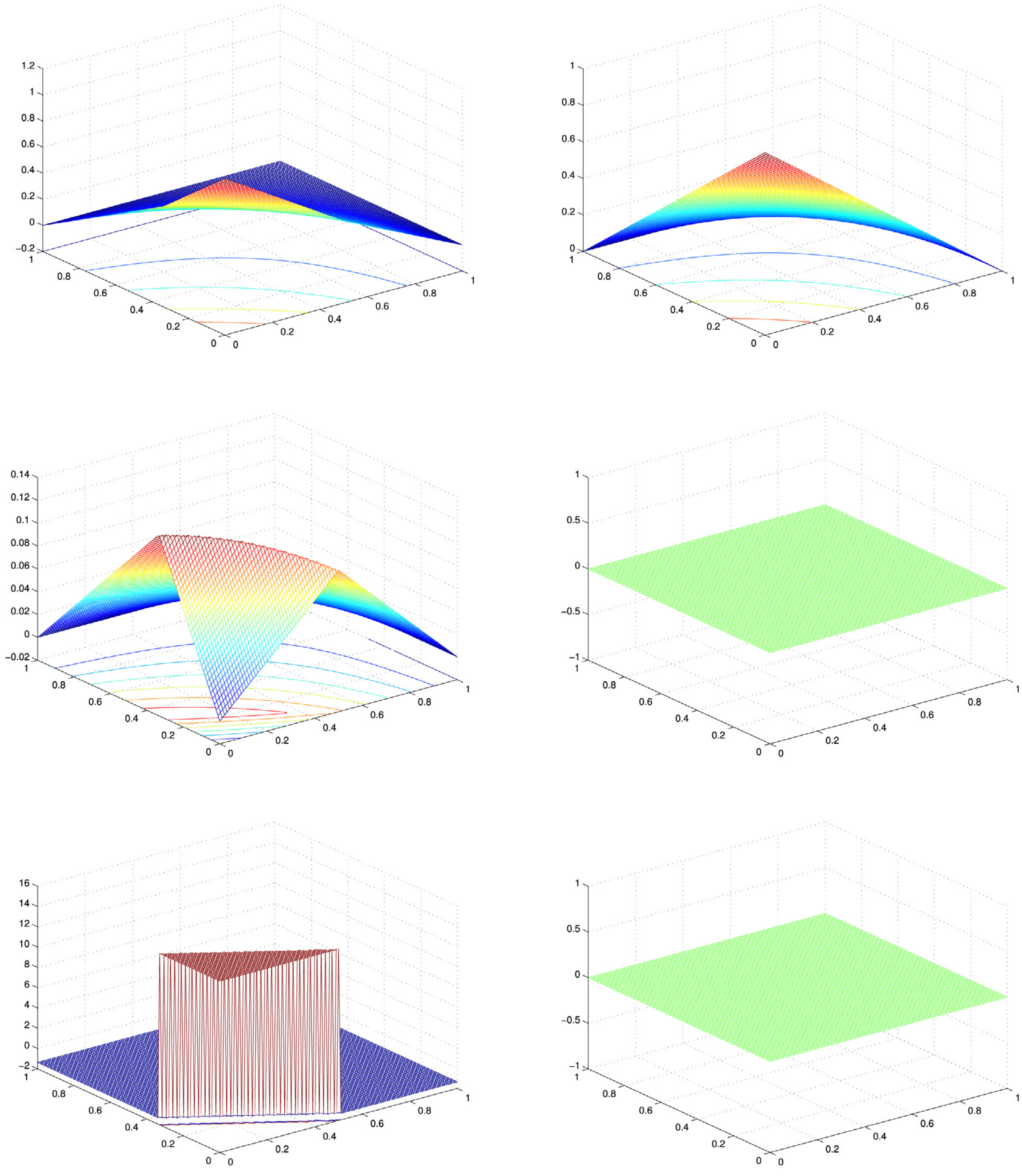


Fig. A.16. The (u_r, u_z, p) components of the IFE shape function Φ_1 (left) and standard shape function Ψ_1 (right) on a reference interface element of Type I.

system, after mapping of (A.12a)–(A.12d) to the reference element, are equal to those used to find the IFE shape functions in Appendix A.1. However, the right-hand side $\mathbf{b}_j$ is different and is given by

$$\mathbf{b}_j = (0, 0, 0, 0, \delta_{j,1}, \delta_{j,2}, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)^T.$$

For the axisymmetric Stokes interface problem with a nonzero surface force $\boldsymbol{\sigma} \neq \mathbf{0}$, we look for an IFE solution $\begin{pmatrix}\mathbf{u}_h\\ p_h\end{pmatrix}$ in $S_h^s(\Omega,t)+\{\mathbf{q}_h^s(r,z,t)\}$ where $S_h^s(\Omega,t)$ is the IFE space defined in Appendix A.1 and the vector function $\mathbf{q}_h^s(r,z,t)$ is such that

$$\mathbf{q}_h^s(r,z,t)=\begin{cases} s_1\Upsilon_1(r,z)+s_2\Upsilon_2(r,z), & \text{on } T\in\mathcal{T}_h^i\\ 0, & \text{elsewhere,}\end{cases} \tag{A.13}$$

where $s_1=\int_{\overline{DE}}\sigma_{h,r}ds$, $s_2=\int_{\overline{DE}}\sigma_{h,z}ds$ and $\boldsymbol{\sigma}_h=(\sigma_{h,r},\sigma_{h,z})^T$ is the linear interpolation of $\boldsymbol{\sigma}=(\sigma_r,\sigma_z)^T$ defined by $\boldsymbol{\sigma}_h(E)=\boldsymbol{\sigma}(E)$ and $\boldsymbol{\sigma}_h(D)=\boldsymbol{\sigma}(D)$. The procedure above guarantees that the particular solution $\mathbf{q}_h^s$ satisfies the stress jump condition (3.1j) in the discrete sense. In fact, the IFE solution on an interface element will take the form

$$\begin{pmatrix}\mathbf{u}_h\\ p_h\end{pmatrix}=\sum_{i=1}^{9}c_i\boldsymbol{\Phi}_i+s_1\Upsilon_1+s_2\Upsilon_2, \tag{A.14}$$

where $\mathbf{u}_h=(u_{r,h},u_{z,h})^T$.

The existence and uniqueness of these shape functions were established in [49], together with their approximation capability. It was shown in [49] that the IFE velocity is second order convergent and the IFE pressure is first order convergent.

References

[1] T. Mason, J. Bibette, Emulsification in viscoelastic media, Phys. Rev. Lett. 77 (1996) 3481–3484.
[2] J. Ottino, P. DeRoussel, S. Hansen, D. Khakhar, Mixing and dispersion of viscous liquids and powdered solids, Adv. Chem. Eng. 25 (1999) 105–204.
[3] C. Lin, L. Guo, Experimental study of drop deformation and breakup in simple shear flows, Chin. J. Chem. Eng. 15 (2007) 1–5.
[4] S. Ramaswamy, L. Leal, The deformation of a viscoelastic drop subjected to steady uniaxial extensional flow of a Newtonian fluid, J. Non-Newton. Fluid Mech. 85 (1999) 127–163.
[5] R. Hooper, M. Toose, C. Macosko, J. Derby, A comparison of boundary element and finite element methods for modeling axisymmetric polymeric drop deformation., Internat. J. Numer. Methods Fluids 37 (2001) 837–864.
[6] H. Hu, N. Patankar, M. Zhu, Direct numerical simulations of fluid-solid systems using the arbitrary Lagrangian-Eulerian technique., J. Comput. Phys. 169 (2001) 427–462.
[7] V. Cristini, J. Blawzdziewicz, M. Loewenberg, Drop breakup in three-dimensional viscous flows, Phys. Fluids 10 (8) (1998) 1781–1783.
[8] R. Khayat, Three-dimensional boundary-element analysis of drop deformation for Newtonian and viscoelastic systems, Internat. J. Numer. Methods Fluids 34 (2000) 241–275.
[9] E. Toose, B. Geurts, J. Kuerten, A boundary integral method for two-dimensional (non)-Newtonian drops in slow viscous flow, Non-Newton. Fluid Mech. 60 (1995) 129–154.
[10] V. Girault, B. Riviere, M. Wheeler, A discontinuous Galerkin method with nonoverlapping domain decomposition for the Stokes and Navier-Stokes problems, Math. Comp. 74 (249) (2005) 53–84.
[11] S. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, J. Comput. Phys. 100 (1992) 25–37.
[12] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, Internat. J. Numer. Methods Fluids 30 (1999) 775–793.
[13] J. Du, B. Fix, J. Glimm, X. Jia, X. Li, Y. Li, L. Wu, A simple package for front tracking, J. Comput. Phys. 213 (2006) 613–628.
[14] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, S. Zaleski, Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows, J. Comput. Phys. 152 (1999) 423–456.
[15] J. Li, Y. Renardy, Numerical study of flows of two immiscible liquids at low Reynolds number, SIAM Rev. 42 (3) (2000) 417–439.
[16] S. Afkhami, S. Zaleski, M. Bussmann, A mesh-dependent model for applying dynamic contact angles to VOF simulations, J. Comput. Phys. 228 (15) (2009) 5370–5389.
[17] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, J. Comput. Phys. 114 (1994) 146–159.
[18] Y. Chang, T. Hou, B. Merriman, S. Osher, A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, J. Comput. Phys. 124 (1996) 449–464.
[19] D.M. Anderson, G.B. McFadden, A.A. Wheeler, Diffuse-interface methods in fluid mechanics, Annu. Rev. Fluid Mech. 30 (1) (1998) 139–165.
[20] D. Jacqmin, Calculation of two-phase navierstokes flows using phase-field modeling, J. Comput. Phys. 155 (1) (1999) 96–127.
[21] J.U. Brackbill, D.B. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys. 100 (1992) 146–159.
[22] R.J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal. 31 (4) (1994) 1019–1044.
[23] Z. Li, M.-C. Lai, The immersed interface method for the navierstokes equations with singular forces, J. Comput. Phys. 171 (2) (2001) 822–842.
[24] J. Chessa, H. Wang, T. Belytschko, An extended finite element method for two-phase fluids, J. Appl. Mech. Trans. ASME 70 (1) (2003) 10–17.
[25] X.-D. Liu, R.P. Fedkiw, M. Kang, A boundary condition capturing method for poisson's equation on irregular domains, J. Comput. Phys. 160 (1) (2000) 151–178.
[26] F. Auricchio, F. Brezzi, A. Lefieux, A. Reali, An “immersed” finite element method based on a locally anisotropic remeshing for the incompressible Stokes problem, Comput. Methods Appl. Mech. Engrg. 294 (Supplement C) (2015) 428–448.
[27] S. Adjerid, N. Chaabane, T. Lin, An immersed discontinuous finite element method for Stokes interface problems, Comput. Methods Appl. Mech. Engrg. 293 (2015) 170–190.
[28] S. Adjerid, M. Ben-Romdhane, T. Lin, High-order interior penalty immersed finite element method for second-order elliptic interface problems, Int. J. Numer. Anal. Model. 11 (2014) 541–566.
[29] S. Adjerid, T. Lin, Higher-order immersed discontinuous Galerkin methods, Int. J. Inf. Syst. Sci. 3 (2007) 558–565.
[30] S. Adjerid, T. Lin, A p^{th}-degree immersed finite element method for boundary value problems with discontinuous coefficients, Appl. Numer. Math. 59 (2009) 1303–1321.
[31] S. Chou, D. Kwak, K. Wee, Optimal convergence analysis of an immersed interface finite element method, Adv. Comput. Math. 33 (2010) 149–168.
[32] 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.
[33] X. He, Bilinear Immersed Finite Elements for Interface Problems (Ph.D dissertation), Virginia Tech, 2009.
[34] X. He, T. Lin, Y. Lin, Approximation capability of a bilinear immersed finite element space, Numer. Methods Partial Differential Equations 24 (2008) 1265–1300.

[35] X. He, T. Lin, Y. Lin, A bilinear immersed finite volume element method for the diffusion equation with discontinuous coefficient, Commun. Comput. Phys. 6 (1) (2009) 185–202.
[36] X. He, T. Lin, Y. Lin, Immersed finite element methods for elliptic interface problems with non-homogeneous jump conditions, Int. J. Numer. Anal. Model. 8 (2) (2011) 284–301.
[37] D. Kwak, K. Wee, K. Chang, An analysis of a broken P_1-nonconforming finite element method for interface problems, SIAM J. Numer. Anal. 48 (6) (2010) 2117–2134.
[38] Z. Li, T. Lin, X. Wu, New Cartesian grid methods for interface problems using the finite element formulation, Numer. Math. 96 (2003) 61–98.
[39] T. Lin, Y. Lin, R. Rogers, L. Ryan, A rectangular immersed finite element space for interface problems, Adv. Comput. Theory Pract. 7 (2001) 107–114.
[40] T. Lin, D. Sheen, X. Zhang, A locking-free immersed finite element method for planar elasticity interface problems, J. Comput. Phys. 247 (2013) 228–247.
[41] T. Lin, X. Zhang, Linear and bilinear immersed finite elements for planar elasticity interface problems, J. Comput. Appl. Math. 236 (18) (2012) 4681–4699.
[42] S. Adjerid, K. Moon, A higher order immersed discontinuous Galerkin finite element method for the acoustic interface problem, in: A. Ansari, H. Temimi (Eds.), Advances in Applied Mathematics, Vol. 18, Springer, New York, 2014, pp. 57–69.
[43] S. Adjerid, K. Moon, An immersed discontinuous Galerkin method for acoustic wave propagation in inhomogeneous media, SIAM J. Numer. Anal. (2018) submitted for publication.
[44] K. Moon, Immersed Discontinuous Galerkin Methods for Acoustic Wave Propagation in Inhomogeneous Media (Ph.D dissertation), Virginia Tech, 2016.
[45] X. Zhang, Nonconforming Immersed Finite Element Methods for Interface Problems (Ph.D dissertation), Virginia Tech, 2013.
[46] R. Beyer, A computational model of the Cochlea using the immersed boundary method, J. Comput. Phys. 98 (1992) 145–162.
[47] B. Riviere, Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations, SIAM, Philadelphia, 2008.
[48] P. Bochev, C. Dohrmann, M. Gunzberger, Stabilization of low-order mixed finite elements for the Stokes equations, SIAM J. Numer. Anal. 44 (1) (2006) 82–101.
[49] N. Chaabane, Immersed and Discontinuous Galerkin Finite Element Methods (Ph.D dissertation), Virginia Tech, 2015.
[50] X. He, T. Lin, Y. Lin, Interior penalty bilinear IFE discontinuous Galerkin methods for elliptic equations with discontinuous coefficient, J. Syst. Sci. Complex. 23 (2010) 467–483.
[51] J. Li, Y. Renardy, Shear-induced rupturing of a viscous drop in a Bingham liquid, J. Non-Newton. Fluid Mech. 95 (2000) 235–251.
[52] P. Yue, C. Zhou, J. Feng, C. Ollivier-Gooch, H. Hu, Phase-field simulations of interfacial dynamics in viscoelastic fluids using finite elements with adaptive meshing, J. Comput. Phys. 219 (2006) 47–67.
[53] H. Zhou, C. Pozrikidis, The flow suspension in channels: Single files of drops, Phys. Fluids A 5 (1993) 311–324.
[54] P. Yue, J. Feng, C. Liu, J. Shen, A diffuse-interface method for simulating two-phase flows of complex fluids, J. Fluid Mech. 515 (2004) 293–317.
[55] H. Udaykumar, R. Mittal, W. Shyy, Computation of solid-liquid phase fronts in the sharp interface limit on fixed grids, J. Comput. Phys. 153 (1999) 535–574.
[56] P. Yue, C. Zhou, J. Feng, Spontaneous shrinkage of drops and mass conservation in phase-field simulations, J. Comput. Phys. 223 (2007) 1–9.
[57] X. He, T. Lin, Y. Lin, X. Zhang, Immersed finite element methods for parabolic equations with moving interface, Numer. Methods Partial Differential Equations 29 (2) (2013) 619–646.