A CutFEM method for two-phase flow problems

Susanne Claus, Pierre Kerfriden

A CutFEM method for two-phase flow problems

Susanne Claus, Pierre Kerfriden

Cardiff University, School of Engineering,
The Parade, CF243AA Cardiff, United Kingdom

June 26, 2018

Abstract

In this article, we present a cut finite element method for two-phase Navier-Stokes flows. The main feature of the method is the formulation of a unified continuous interior penalty stabilisation approach for, on the one hand, stabilising advection and the pressure-velocity coupling and, on the other hand, stabilising the cut region. The accuracy of the algorithm is enhanced by the development of extended fictitious domains to guarantee a well defined velocity from previous time steps in the current geometry. Finally, the robustness of the moving-interface algorithm is further improved by the introduction of a curvature smoothing technique that reduces spurious velocities. The algorithm is shown to perform remarkably well for low capillary number flows, and is a first step towards flexible and robust CutFEM algorithms for the simulation of microfluidic devices.

Keywords: cut finite element method, two-phase flow, micro-fluidic devices, level set technique, continuous interior penalty

1 Introduction

The development of general-purpose, robust and accurate finite element schemes for the simulation of two-phase immiscible fluids is hindered by several key scientific challenges. Firstly, the interface between the two fluids needs to be tracked whilst allowing topological changes such as droplet merging or breakup. Secondly, a contrast in fluid densities and viscosities causes a velocity kink at the interface. Similarly, surface tension forces cause the pressure field to jump at the interface. Discontinuities of these types are not easily represented in a finite element context, where fields are assumed to be piecewise smooth by construction. Finally, these difficulties, which are specific to multi-phase flows, have to

be solved in the context of general fluid dynamics, where advection and pressure-velocity coupling are well-known sources of numerical instabilities.

Two main types of numerical approaches can be distinguished to represent the interface between two fluids: sharp interface approaches and diffusive interface approaches. In diffusive interface approaches (e.g. phase field methods [2, 30], conservative level set methods [39], level set methods [43]), a smoothing region around the interface is introduced, which blends viscosities and densities smoothly from one phase to the other. In this context, surface tension forces are usually applied over the entire smoothing region (e.g. continuum surface force approach [3]). Smoothing the interface makes the solver more robust, in particular when surface tension is strong. Moreover, topological changes are handled in a straightforward manner. However, the most stringent drawback of this type of approach is the need for a high number of elements to resolve the smoothing region. In contrast, sharp interface methods, which are of interest in this article, do not introduce such a smoothing region and hence less elements are required to resolve the interface region. The most well-known sharp interface method is the ALE method [15], in which the elements of the finite element mesh are deformed to match the fluid interface. This mesh moving approach works remarkably well for small deformations but requires complex re-meshing procedures for large deformations and topological changes. Alternatively, an emerging family of methods allow the sharp interface to cut through the elements, and to move through a fixed background grid without re-meshing. In this case, the immersed interface may be tracked using either Lagrangian markers (e.g. immersed boundary method [37], front-tracking [45]) or functional representations (e.g. level-set-method [43]). Importantly, elements which are intersected by the interface need to be enriched in order to capture jumps and kinks in the solution. Failure to do so significantly deteriorates convergence with mesh refinement and introduces significant spurious velocity modes for flows involving surface tension. Finally, the most complex aspect of this family of methods is their intrinsic need for tailored stabilisation. XFEM and CutFEM are two popular frameworks proposed to systematise the process of embedding discontinuities in finite element solvers, which may be used to build two-phase flow solvers. XFEM enriches the finite element shape functions by the Partition-of-Unity method [36] and was originally introduced by [38] for crack propagation problems. XFEM was first developed for solid mechanics and was later extended to two-phase flow problems [11, 19, 16, 41]. Stabilisation in XFEM is usually achieved through a combination of pre-conditioning [31, 20, 33], deactivation of degrees of freedom [17] or more recently the use of ghost penalty stabilisation [42, 20]. The CutFEM approach [9] uses overlapping fictitious domains in combination with ghost penalty stabilisation [4] to enrich and stabilise the solution. CutFEM has been developed for, e.g. Stokes interface problems [22, 34], for the Oseen problem [48] and fluid structure interaction problems [35, 6]. An alternative to the two families of methods described previously was proposed in [26], where an unfitted discontinuous Galerkin approach was successfully developed to solve two-phase flow problems.

In this article, we develop a new CutFEM solver for multiphase flows. The main novel and appealing feature of the proposed framework is the unification of the stabilisation processes. More precisely, we will use continuous interior penalty (CIP) to stabilise the

unfitted interface discretisation, which is known as ghost-penalty in the context of CutFEM, and we will also use CIP to stabilise advection and the pressure-velocity coupling. In addition to the development, implementation and validation of this unified framework, we propose a series of novel tools to improve the capabilities and performances of the CutFEM framework when addressing two-phase flows problems. These are: (i) the definition of the interface tracking problem using extended fictitious subdomains. This is to guarantee that velocities from the past time step are well-defined in the current geometry; and (ii) a curvature smoothing for the surface tension force which reduces the amplitude of spurious interface velocity modes. Consistently with our previous CutFEM developments [9, 13, 14], the different phases are described implicitly using the zero isoline of a level-set function. Finite element enrichment is obtained through the application of an overlapping domain decomposition technique that allows the pressure to jump and the velocity to kink across the fluid interface. Time integration is performed using an implicit Euler scheme, whilst the velocity and pressure fields are spatially discretised using linear continuous finite element spaces. We validate the developments in several ways. Firstly, we investigate the performance of the two-phase flow solver on the rising bubble benchmark. We show that our results agree with that of existing schemes [29]. Then, we study and discuss the effect of the parameters of the proposed unified stabilisation scheme on the shape of the rising bubble. Finally, we solve the problem of a droplet in a microfluidic 5:1:5 contraction and expansion flow, which allows us to prove the robustness of the CutFEM scheme for a surface tension dominated flow.

The paper is organised as follows: In Section 2, we introduce the strong formulation of the two-phase Navier-Stokes problem together with the level set advection problem followed by their weak formulation. In Section 3, we introduce the extended fictitious domains, describe the discretisation in time and introduce the stabilised cut finite element scheme for two-phase Navier-Stokes flows. We then discuss the stabilised curvature projection before summarising the scheme in Section 4. In Section 5, we present numerical results for the rising bubble benchmark and a droplet in a 5:1:5 contraction and expansion problem.

2 The Two-Phase Navier-Stokes Problem

Let Ω be a domain in $\mathbb{R}^d$ ($d = 2$ or 3) with a polyhedral boundary $\partial\Omega$. We assume that two immiscible incompressible Newtonian fluids occupy the time-dependent subdomains $\Omega_i = \Omega_i(t) \subset \Omega$, $i = 1, 2$, with $\Omega = \Omega_1(t) \cup \Omega_2(t)$ and that $\Gamma = \Gamma(t)$ denotes the smooth interface between them.

Two-phase Navier-Stokes Problem The two-phase flow system is described by the following Navier-Stokes problem in the time interval $t \in (0, T]$: Find the velocity $\mathbf{u} : \Omega \times (0, T] \to \mathbb{R}^d$ and the pressure $p : \Omega \times (0, T] \to \mathbb{R}$, such that

$$\begin{aligned} \rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} - \nabla\cdot\boldsymbol{\sigma} &= \rho\boldsymbol{g} && \text{in } \Omega, \\ \boldsymbol{\sigma} &= 2\eta\epsilon(\mathbf{u}) - p\mathbf{I} && \text{in } \Omega, \\ \nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega. \end{aligned} \tag{2.1}$$

Here, $\boldsymbol{\sigma}$ is the Cauchy stress tensor, $\epsilon(\mathbf{u}) = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)$ is the rate of deformation tensor, $\eta|_{\Omega_i} = \eta_i$, $i = 1, 2$, is the piecewise constant fluid viscosity on Ω_i, $\rho|_{\Omega_i} = \rho_i$, $i = 1, 2$, is the piecewise constant density on Ω_i and $\boldsymbol{g} \in [L^2(\Omega)]^d$ is a known force (e.g. gravity). Additionally, we have given initial conditions for the velocity $\mathbf{u}(\boldsymbol{x}, 0) = \mathbf{u}_0$ and boundary conditions on $\partial\Omega$ which we here assume to be of Dirichlet type for simplicity, i.e.

$$\mathbf{u} = \mathbf{u}_D \text{ on } \partial\Omega \tag{2.2}$$

with given function $\mathbf{u}_D \in [H^{1/2}(\partial\Omega)]^d$. At the interface between the two fluids, we enforce a zero jump condition on the velocity, $[\![\mathbf{u}]\!] = (\mathbf{u}_1 - \mathbf{u}_2)|_\Gamma = 0$, $\mathbf{u}_i = \mathbf{u}|_{\Omega_i}$, $i = 1, 2$, and the jump of the traction vector over the interface is given by the surface tension, i.e.

$$\begin{aligned} [\![\mathbf{u}]\!] &= 0 && \text{on } \Gamma(t), \\ [\![\boldsymbol{\sigma}\cdot\boldsymbol{n}_\Gamma]\!] &= -\gamma\kappa\boldsymbol{n}_\Gamma && \text{on } \Gamma(t). \end{aligned} \tag{2.3}$$

Here, γ is the surface tension coefficient, κ is the curvature of the interface and $\boldsymbol{n}_\Gamma$ is the outward unit normal on the interface pointing from $\Omega_1(t)$ to $\Omega_2(t)$.

Interface Evolution To capture the evolution of the interface $\Gamma(t)$, we will use a level set function. The level set function is a scalar function on Ω, such that $\phi(\boldsymbol{x}, t) < 0$ for $\boldsymbol{x} \in \Omega_1(t)$, $\phi(\boldsymbol{x}, t) > 0$ for $\boldsymbol{x} \in \Omega_2(t)$ and $\phi(\boldsymbol{x}, t) = 0$ for $\boldsymbol{x} \in \Gamma(t)$. The level set function will be moved with the fluid velocity $\mathbf{u}$ by solving the following advection problem: Find the level set function $\phi : \Omega \times (0, T] \to \mathbb{R}$, such that

$$\frac{\partial\phi}{\partial t} + \mathbf{u}\cdot\nabla\phi = 0 \tag{2.4}$$

with given initial condition $\phi(\boldsymbol{x}, 0) = \phi_0$ and inflow boundary conditions $\phi = \phi_{in}$ at the inflow boundary $\partial\Omega_{in}(t) := \{\boldsymbol{x} \in \partial\Omega(t) : \mathbf{u}(\boldsymbol{x}, t)\cdot\boldsymbol{n} < 0\}$. Here, $\boldsymbol{n}$ is the outward normal to Ω.

2.1 Weak Formulation

Let us denote

$$\begin{aligned}(u,v)_{\Omega_i} &= \int_{\Omega_i} u_i v_i \, dx \\ (\llbracket u \rrbracket, \llbracket v \rrbracket)_\Gamma &= \int_\Gamma \llbracket u \rrbracket \llbracket v \rrbracket \, ds\end{aligned} \tag{2.5}$$

and

$$(u,v)_\Omega = \sum_i (u_i, v_i)_{\Omega_i}, \quad i = 1,2. \tag{2.6}$$

And let us introduce the following Sobolev space

$$L^2_0(\Omega) = \left\{ p \in L^2(\Omega) : \int_\Omega p = 0 \right\}. \tag{2.7}$$

Weak Formulation of the Two-Phase Navier-Stokes Problem We seek the solution to the two-phase Navier-Stokes problem for $t \in (0,T]$, for $U(t) := (\mathbf{u}(t), p(t)) \in [H^1(\Omega)]^d \times L^2_0(\Omega)$, such that

$$A(U(t), V) = L(V) \tag{2.8}$$

for all $V = (\mathbf{v}, q) \in [H^1(\Omega)]^d \times L^2_0(\Omega)$.
Here,

$$\begin{aligned}A(U(t),V) = &\left(\rho \frac{\partial \mathbf{u}(t)}{\partial t}, \mathbf{v}\right)_\Omega + \mathcal{N}(\mathbf{u}(t), \mathbf{u}(t), \mathbf{v}) \\ &+ a(\mathbf{u}(t), \mathbf{v}) + b(p(t), \mathbf{v}) - b(q, \mathbf{u}(t)) + a_n(\mathbf{u}(t), \mathbf{v}), \\ L(V) = &(\rho \boldsymbol{g}, \mathbf{v})_\Omega - (\gamma \kappa \boldsymbol{n}_\Gamma, \langle \mathbf{v} \rangle)_\Gamma + l_n(\mathbf{v}),\end{aligned} \tag{2.9}$$

consist of a non-linear convection term

$$\mathcal{N}(\mathbf{u}(t), \mathbf{u}(t), \mathbf{v}) = (\rho(\mathbf{u}(t) \cdot \nabla)\mathbf{u}(t), \mathbf{v})_\Omega, \tag{2.10}$$

terms that arise from the integration by parts of $(\nabla \cdot \boldsymbol{\sigma}, \mathbf{v})_\Omega$

$$\begin{aligned}a(\mathbf{u}(t), \mathbf{v}) &= (2\eta\epsilon(\mathbf{u}(t)), \epsilon(\mathbf{v}))_\Omega - (2\eta\epsilon(\mathbf{u}(t)) \cdot \boldsymbol{n}, \mathbf{v})_{\partial\Omega} - (\{2\eta\epsilon(\mathbf{u}(t)) \cdot \boldsymbol{n}_\Gamma\}, \llbracket \mathbf{v} \rrbracket)_\Gamma, \\ b(p(t), \mathbf{v}) &= -(p(t), \nabla \cdot \mathbf{v})_\Omega + (p(t) \cdot \boldsymbol{n}, \mathbf{v})_{\partial\Omega} + (\{p(t) \cdot \boldsymbol{n}_\Gamma\}, \llbracket \mathbf{v} \rrbracket)_\Gamma\end{aligned} \tag{2.11}$$

and terms that enforce the interface condition $\llbracket \mathbf{u} \rrbracket = \mathbf{0}$ and the boundary condition $\mathbf{u} = \mathbf{u}_D$ weakly using Nitsche's method

$$\begin{aligned}a_n(\mathbf{u}, \mathbf{v}) = &- (2\eta\epsilon(\mathbf{v}) \cdot \boldsymbol{n}, \mathbf{u})_{\partial\Omega} + \lambda_{\partial\Omega}(h)\, \eta\, (\mathbf{u}, \mathbf{v})_{\partial\Omega} \\ &- (\{2\eta\epsilon(\mathbf{v}) \cdot \boldsymbol{n}_\Gamma\}, \llbracket \mathbf{u} \rrbracket)_\Gamma + \lambda_\Gamma(h)\, \{\eta\}\, (\llbracket \mathbf{u} \rrbracket, \llbracket \mathbf{v} \rrbracket)_\Gamma, \\ l_n(\mathbf{v}) = &- (2\eta\epsilon(\mathbf{v}) \cdot \boldsymbol{n}, \mathbf{u}_D)_{\partial\Omega} - (q \cdot \boldsymbol{n}, \mathbf{u}_D)_{\partial\Omega} + \lambda_{\partial\Omega}(h)\, \eta\, (\mathbf{u}_D, \mathbf{v})_{\partial\Omega}.\end{aligned} \tag{2.12}$$

Here, $\lambda_\Gamma(h) \in \mathbb{R}$ and $\lambda_{\partial\Omega}(h) \in \mathbb{R}$ are positive penalty parameters that have to be chosen large enough to guarantee the coercivity of the weak formulation and $\boldsymbol{n}$ is the outward normal on $\partial\Omega$.

Note that the interface terms in the equations above have been obtained through integration by parts of $(\nabla \cdot \boldsymbol{\sigma}, \mathbf{v})_\Omega$, summing over Ω_i and then using the relationship

$$[\![ab]\!] = [\![b]\!] \{a\} + \langle b \rangle [\![a]\!] \tag{2.13}$$

to reformulate them in terms of jumps and averages. The averages are weighted and defined as

$$\{\boldsymbol{v}\} := w_1 \boldsymbol{v}_1 + w_2 \boldsymbol{v}_2, \tag{2.14}$$
$$\langle \boldsymbol{v} \rangle := w_2 \boldsymbol{v}_1 + w_1 \boldsymbol{v}_2, \tag{2.15}$$

where w_1 and w_2 are weights to be chosen such that $w_1 + w_2 = 1, 0 \leq w_i \leq 1, i = 1, 2$. These weights need to be chosen carefully for problems involving a high contrast in material parameters (see e.g. [7] for advection diffusion problems). To ensure robustness with respect to a high contrast in the fluid viscosities as well as fluid densities, we choose the weights as the harmonic average over the kinematic viscosity $\nu_i := \frac{\eta_i}{\rho_i}$, $i = 1, 2$, i.e.

$$\begin{aligned} w_1 &:= \frac{\nu_2}{\nu_1 + \nu_2}, \\ w_2 &:= \frac{\nu_1}{\nu_1 + \nu_2}. \end{aligned} \tag{2.16}$$

Weak Formulation of the Level Set Advection The weak formulation for the level set advection problem reads: For $t \in (0, T]$, find $\phi(t) \in H^1(\Omega)$, such that

$$\left(\frac{\partial \phi(t)}{\partial t}, \psi \right)_\Omega + (\mathbf{u}(t) \cdot \nabla \phi(t), \psi)_\Omega = 0 \tag{2.17}$$

for all $\psi \in H^1(\Omega)$ with suitable initial and boundary conditions.

3 Stabilised Cut Finite Element Formulation

3.1 Mesh Quantities

In this section, we define mesh quantities that will be used in our finite element discretisation. We discretise the domain Ω, which will stay fixed in time, using a quasi-uniform triangulation, $\{\mathcal{T}_h\}_{0<h\leq 1}$, independent of the location of the interface $\Gamma(t)$ but coinciding with the outer boundary $\partial\Omega$. We consider the evolution of $\Gamma(t)$ through Ω in the time interval $t \in [0, T]$, which we decompose into n_t equal steps, i.e. $t_n = \Delta t\, n$ with $\Delta t = T/n_t$. For each t_n, we define a range of sets of active elements and facets (i.e. faces in 3D and edges in 2D) in $\mathcal{T}_h$ associated with $\Gamma^n = \Gamma(t_n)$ and $\Omega_i^n = \Omega_i(t_n)$.
Firstly, let us consider the δ neighbourhood of Γ^n defined as

$$\mathcal{O}_\delta^n = \{\boldsymbol{x} \in \mathbb{R}^d : dist(\boldsymbol{x}, \Gamma^n) \leq \delta\} \tag{3.1}$$

and the extended domains

$$\Omega_{i,\delta}^n = \Omega_i^n \cup \mathcal{O}_\delta^n, i = 1, 2. \tag{3.2}$$

Secondly, let us define submeshes of $\mathcal{T}_h$, associated with these extended domains, consisting of all elements in $\mathcal{T}_h$ that have at least a small part in domain $\Omega_{i,\delta}^n$ as

$$\mathcal{T}_{h,i}(t_n) = \{K \in \mathcal{T}_h : K \cap \Omega_{i,\delta}^n \neq \emptyset\}, i = 1, 2. \tag{3.3}$$

We call $\mathcal{T}_{h,i}(t)$ the *active mesh* of $\Omega_i(t)$. Note that these active meshes do not conform to $\Gamma(t)$ as shown in Figure 3.1. We denote the union of all elements in $\mathcal{T}_{h,i}(t_n)$ as

$$\Omega_i^*(t_n) = \bigcup_{K \in \mathcal{T}_{h,i}(t_n)} K, i = 1, 2. \tag{3.4}$$

$\Omega_i^*(t)$ is called the *fictitious domain* of $\Omega_i(t)$ (see Figure 3.1). Furthermore, consider the set of all elements with part in the δ neighbourhood, denoted by

$$G_h^n = \{K \in \mathcal{T}_h : K \cap \mathcal{O}_\delta^n \neq \emptyset\}. \tag{3.5}$$

We will apply stabilisation terms to a subset of facets (i.e. faces in 3D and edges in 2D) in these domains. In particular, for each active mesh at time t_n, we will apply stabilisation terms to so-called *interior facets* and *ghost penalty facets* of $\mathcal{T}_{h,i}(t_n)$. The set of interior facets, $\mathcal{F}_I^i(t_n)$, contains all facets that are shared by two elements in $\mathcal{T}_{h,i}(t_n), i = 1, 2$ (see Figure 3.2). The set of intersected facets, $\mathcal{F}_{int}(t_n)$, contains all facets intersected by the interface. And ghost penalty facets are all facets in the set

$$\mathcal{F}_G^i(t_n) = \{F \in \mathcal{F}_I^i(t_n) : K^+ \in G_h^n \text{ or } K^- \in G_h^n\}. \tag{3.6}$$

Here, K^+ and K^- are the two elements sharing the interior face $F \in \mathcal{F}_I^i(t_n)$ (see Figure 3.2). We place the following assumptions on the background mesh $\mathcal{T}_h$, to ensure that the interface $\Gamma(t)$ is sufficiently resolved: For $t \in [0, T]$,

- G1: $\Gamma(t)$ crosses element faces at most once.
- G2: For each element K intersected by $\Gamma(t)$, there exists a plane S_K and a piecewise smooth parametrization $\Phi : S_K \cap K \to \Gamma(t) \cap K$.
- G3: We assume that for each intersected element K, there are elements $K_i \in \Omega_i(t), i = 1, 2$, such that $\overline{K} \cap \overline{K_i} \neq \emptyset$. This means each intersected element shares at least one vertex or facet with an element K_1 in $\Omega_1(t)$ and an element K_2 in $\Omega_2(t)$.

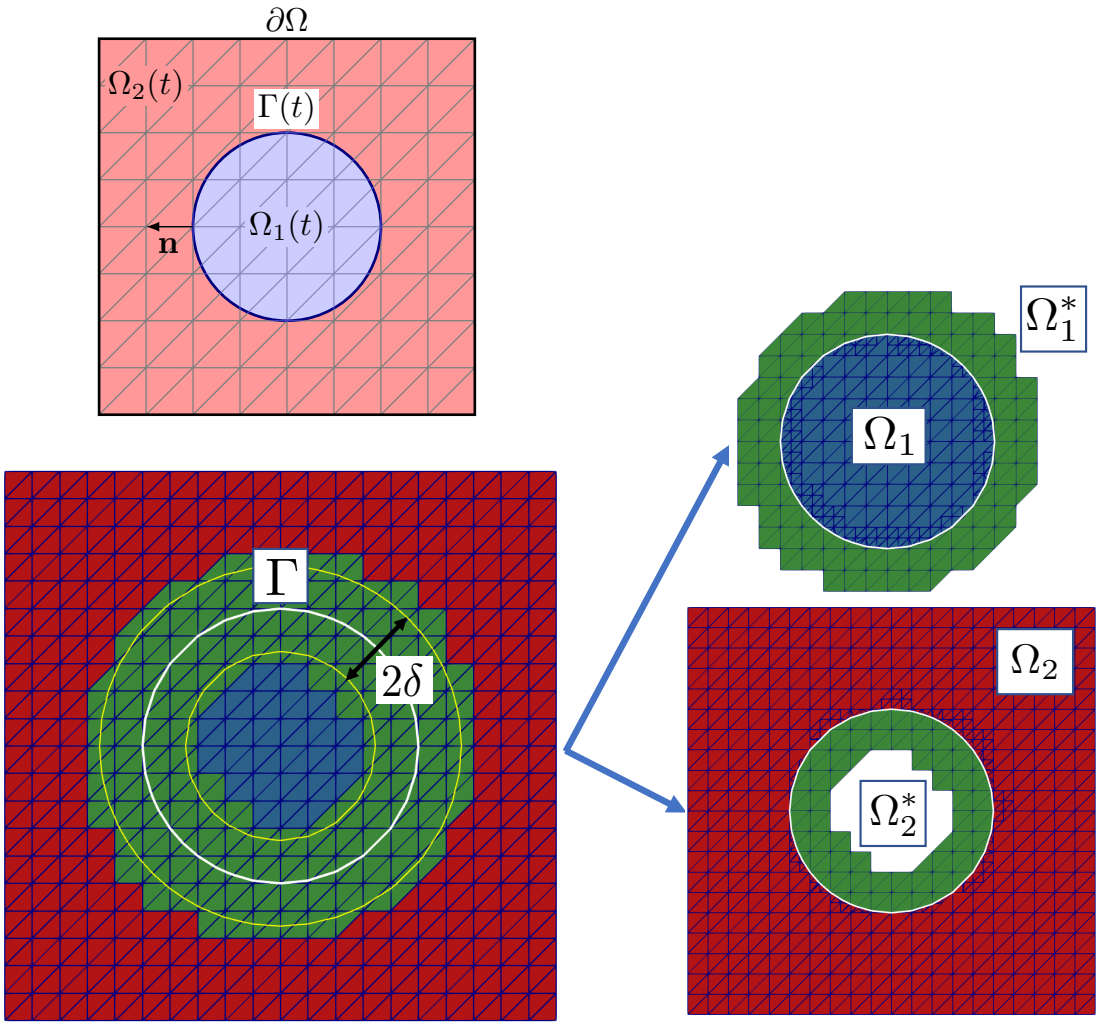
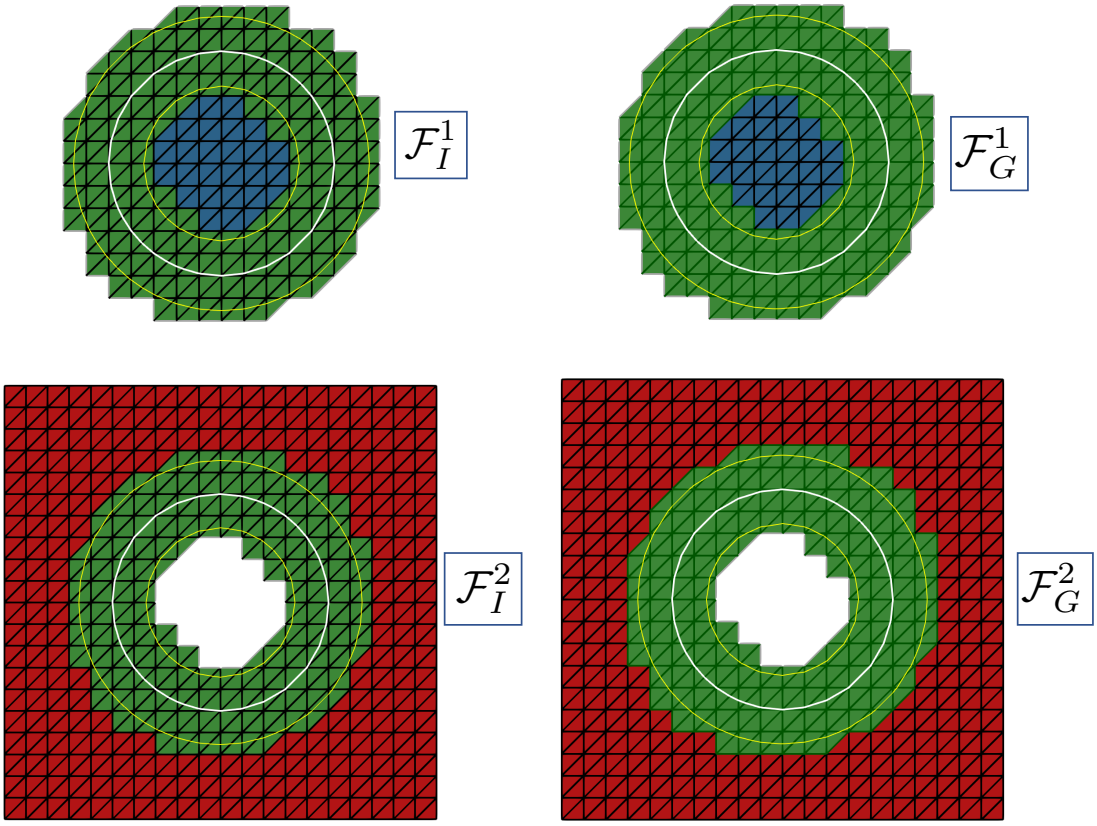
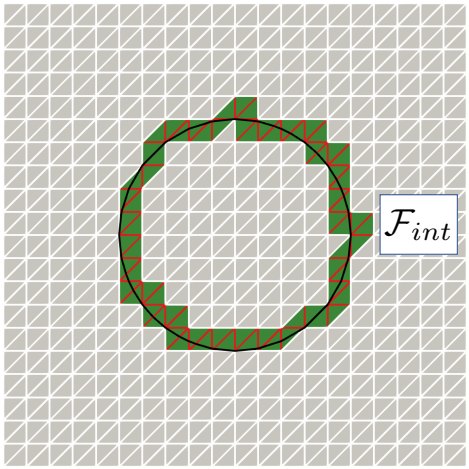


Figure 3.1: Navier-Stokes multi-domain discretisation. Cells with a distance smaller than δ to the interface are doubled (elements shaded in green). The interior domain Ω_1 gets extended to the fictitious domain Ω_1^* and the exterior domain, Ω_2, gets extended to the fictitious domain Ω_2^* resulting in overlapping fictitious domains (green shaded elements).



(a)



(b)

Figure 3.2: Illustration of different facet sets in two-phase Navier-Stokes problem. The top figure illustrates the definition of interior $\mathcal{F}_I^i$ (black edges) and ghost penalty facets $\mathcal{F}_G^i$ (dark green edges) and the bottom figure displays the facets which are intersected by the interface, $\mathcal{F}_{int}$ (red edges).

3.2 Discretisation in Time

As mentioned in the previous section, we decompose the time interval $[0, T]$ into n_t equally spaced time steps. For each t_n, we decouple the two-phase Navier-Stokes problem from the level set advection by first solving the two-phase Navier-Stokes problem in $\Omega = \Omega_1^n \cup \Omega_2^n$ and then use the obtained velocity to advect the level set function to obtain the updated domains Ω_i^{n+1}.
We employ an implicit Euler scheme to discretise the two-phase Navier-Stokes problem in time which yields

$$\rho\left(\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\Delta t}, \mathbf{v}\right)_\Omega + \mathcal{N}(\mathbf{u}^{n-1}, \mathbf{u}^n, \mathbf{v}) + a(\mathbf{u}^n, \mathbf{v}) + a_n(\mathbf{u}^n, \mathbf{v}) + b(p^n, \mathbf{v}) - b(q, \mathbf{u}^n) = L(V) \tag{3.7}$$

together with suitable initial conditions. Here, we have linearised the non-linear term using the approximation

$$\mathcal{N}(\mathbf{u}^n, \mathbf{u}^n, \mathbf{v}) \approx \mathcal{N}(\mathbf{u}^{n-1}, \mathbf{u}^n, \mathbf{v}) = \rho\left((\mathbf{u}^{n-1} \cdot \nabla)\mathbf{u}^n, \mathbf{v}\right)_\Omega. \tag{3.8}$$

Note that all integrals above are evaluated in the domains $\Omega_i(t_n)$, i.e.

$$(\mathbf{u}, \mathbf{v})_\Omega = \sum_{i=1}^{2} (\mathbf{u}_i, \mathbf{v}_i)_{\Omega_i^n}. \tag{3.9}$$

This poses the challenge to evaluate the velocity from the previous time step, $\mathbf{u}^{n-1}$, which was obtained in $\Omega_i(t_{n-1})$ in the updated domains $\Omega_i(t_n)$. To achieve this, we define an extension of $\mathbf{u}^{n-1}$ from $\Omega_i(t_{n-1})$ to the extended domains $\Omega_{i,\delta}^{n-1}$ using an extension operator

$$\mathcal{E}_i^{n-1} : H^1(\Omega_i^{n-1}) \to H^1(\Omega_{i,\delta}^{n-1}). \tag{3.10}$$

And we require the δ-neighbourhood to be large enough such that

$$\Omega_i^n \subset \Omega_{i,\delta}^{n-1}. \tag{3.11}$$

This guarantees that the extensions of $\mathbf{u}_i^{n-1}$, denoted by $\mathcal{E}_i^{n-1}\mathbf{u}_i^{n-1}$, $i = 1, 2$, are well defined in $\Omega_i(t_n)$. We introduce the velocity of the previous time-step evaluated as its extension in $\Omega = \Omega_1^n \cup \Omega_2^n$ as

$$\boldsymbol{\beta} := \begin{cases} \mathcal{E}_1^{n-1}\mathbf{u}_1^{n-1} & \text{for } \boldsymbol{x} \in \Omega_1^n, \\ \mathcal{E}_2^{n-1}\mathbf{u}_2^{n-1} & \text{for } \boldsymbol{x} \in \Omega_2^n \backslash \Gamma. \end{cases} \tag{3.12}$$

Using this extension definition, we reformulate the two-phase Navier-Stokes problem as

$$\rho\left(\frac{\mathbf{u}^n - \boldsymbol{\beta}}{\Delta t}, \mathbf{v}\right)_\Omega + \mathcal{N}(\boldsymbol{\beta}, \mathbf{u}^n, \mathbf{v}) + a(\mathbf{u}^n, \mathbf{v}) + a_n(\mathbf{u}^n, \mathbf{v}) + b(p^n, \mathbf{v}) - b(q, \mathbf{u}^n) = L(V). \tag{3.13}$$

The level set advection problem is discretised in time using a θ-scheme [18] which yields

$$\left(\frac{\phi^n - \phi^{n-1}}{\Delta t}, \psi\right)_\Omega + \theta\left(\mathbf{u}^n \cdot \nabla\phi^n, \psi\right)_\Omega + (1-\theta)\left(\boldsymbol{\beta} \cdot \nabla\phi^{n-1}, \psi\right)_\Omega = 0 \tag{3.14}$$

with suitable initial and boundary conditions. In this contribution, we choose $\theta = 0.5$ which yields a Crank-Nicolson scheme.

3.3 Discretisation in Space

3.3.1 Domain and interface approximation

Domain approximation As previously mentioned, we describe our domains $\Omega_1(t)$ and $\Omega_2(t)$ using a level set function. This level set function is discretised using a continuous piecewise quadratic finite element space on the fixed background mesh $\mathcal{T}_h$ denoted by $\mathcal{W}_h$. Then, to define our discrete domains $\Omega_{i,h}$ and Γ_h, we use a two-grid solution proposed by [19, 18]. This two-grid solution can be outlined as follows. Firstly, we project the piecewise quadratic level set function into a piecewise linear finite element space defined by a hierarchically refined mesh $\mathcal{T}_{h/2}$ (*i.e.* each edge is bisected). Let us call $\mathcal{I}(\phi_h)$ the projection of the piecewise quadratic level set defined by

$$\mathcal{I}(\phi_h(\mathbf{v})) = \phi_h(\mathbf{v}) \text{ for all nodes } \mathbf{v} \text{ in } \mathcal{T}_{h/2}. \tag{3.15}$$

We then use this piecewise linear interpolation to determine the intersection between $\mathcal{I}(\phi_h)$ and the refined grid to obtain the piecewise linear approximation of $\Omega_{i,h}$ and Γ_h as illustrated in Figure 3.3.

Interface normals The normal pointing from Ω_1 to Ω_2 can be obtained using the level set function as

$$n_\Gamma(x,t) = \frac{\nabla\phi(x,t)}{\|\nabla\phi(x,t)\|}. \tag{3.16}$$

In the finite element approximation, the normal n_Γ is obtained from the piecewise quadratic level set function through a L^2-projection onto the continuous piecewise linear space

$$\mathcal{X}_h^d := \left\{ v_h \in [C^0(\Omega)]^d : v_h|_K \in \mathcal{P}_1(K) \, \forall K \in \mathcal{T}_h \right\}. \tag{3.17}$$

Here, $d = 2, 3$ is the geometrical dimension. We determine the normal by finding $n_\Gamma \in \mathcal{X}_h^d$ such that for all $\delta n_\Gamma \in \mathcal{X}_h^d$

$$(n_\Gamma, \delta n_\Gamma)_\Omega = \left(\frac{\nabla\phi}{|\nabla\phi|}, \delta n_\Gamma \right)_\Omega. \tag{3.18}$$

Reinitialisation To ensure that the level set function is close to the signed distance function, we reinitialise the level set function in regular time intervals using a fast marching scheme as described in [18, 19].

3.3.2 Stabilised Cut Two-Phase Navier-Stokes Discretisation

Let $\mathcal{V}_h$ denote the space of piecewise linear polynomials on $\mathcal{T}_h$, i.e.

$$\mathcal{V}_h = \left\{ v_h \in C^0(\Omega) : v_h|_K \in \mathcal{P}_1(K) \, , \, \forall K \in \mathcal{T}_h \right\} \tag{3.19}$$

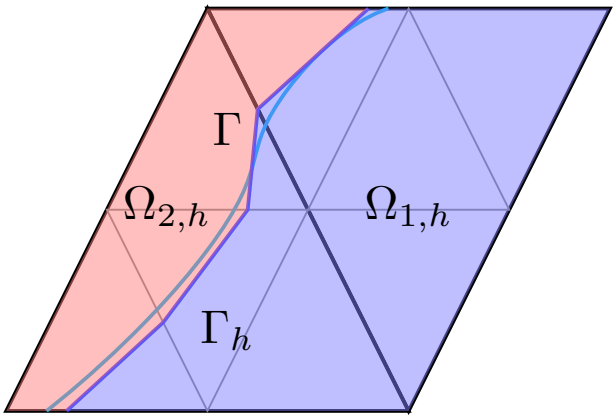


Figure 3.3: Illustration of linear approximation of interface Γ and domains $\Omega_i, i = 1, 2$, on the refined mesh $\mathcal{T}_{h/2}$ with respect to the mesh $\mathcal{T}_h$.

and let

$$\boldsymbol{V}_h^n = \mathcal{V}_{h,1}^n \oplus \mathcal{V}_{h,2}^n \quad \text{with } \mathcal{V}_{h,i}^n = \mathcal{V}_h|_{\mathcal{T}_{h,i}(t_n)}, i = 1, 2. \tag{3.20}$$

Choosing $p_h(t_n) \in \boldsymbol{V}_h^n$ now means that $p_h(t_n)$ consists of a pair $(p_{h,1}^n, p_{h,2}^n)$, with $p_{h,i}^n \in \mathcal{V}_{h,i}^n$, and $p_h(t_n) \in \boldsymbol{V}_h$ is double valued for elements that are both in $\mathcal{T}_{h,1}(t_n)$ and $\mathcal{T}_{h,2}(t_n)$, i.e. all elements in the δ neighbourhood G_h^n of the moving interface. This allows p_h to be discontinuous at the interface (see Figure 3.1) similar to XFEM methods. Analogously, we define the vector-valued linear finite element space $[\boldsymbol{V}_h^n]^d$ for the velocity. Note that the velocity is also double valued for elements in G_h^n and together with the zero jump condition, this yields a velocity which can exhibit kinks at the interface. Additionally, the definition of $\mathbf{u}_{h,i}$, $i = 1, 2$ on the extended fictitious domains Ω_i^* will be used to evaluate the velocity of the previous time-step, $\mathbf{u}_{h,i}^{n-1}$, in the updated domains $\Omega_{h,i}^n$. In this contribution, the extension of the velocity and the pressure from the physical domain, $\Omega_{h,i}^n$, to the fictitious domain, $\Omega_i^*(t_n)$ will be achieved through facet stabilisation terms defined hereafter.
The stabilised finite element formulation of the two-phase Navier-Stokes problem reads: For $t_n \in \{t_1, \dots t_{n_t}\}$, find $U_h := (\mathbf{u}_h^n, p_h^n) \in [\boldsymbol{V}_h^n]^d \times \boldsymbol{V}_h^n$ such that for all $V_h := (\mathbf{v}_h, q_h) \in [\boldsymbol{V}_h^n]^d \times \boldsymbol{V}_h^n$

$$A(U_h, V_h) + J(U_h, V_h) = L(V_h), \tag{3.21}$$

where

$$\begin{aligned} A(U_h, V_h) =& \rho \left(\frac{\mathbf{u}_h^n}{\Delta t}, \mathbf{v}_h \right)_\Omega + \mathcal{N}(\boldsymbol{\beta}_h, \mathbf{u}_h^n, \mathbf{v}_h) + a(\mathbf{u}_h^n, \mathbf{v}_h) + a_n(\mathbf{u}_h^n, \mathbf{v}_h) \\ &+ b(p_h^n, \mathbf{v}_h) - b(q_h, \mathbf{u}_h^n), \qquad (3.22) \\ L(V_h) =& \rho \left(\frac{\boldsymbol{\beta}_h}{\Delta t}, \mathbf{v}_h \right)_\Omega + f_\Gamma(\mathbf{v}_h), \qquad (3.23) \\ J(U_h, V_h) =& j_\beta(\mathbf{u}_h, \mathbf{v}_h) + j_{div}(\mathbf{u}_h, \mathbf{v}_h) + j_p(p_h, q_h) + g_u(\mathbf{u}_h, \mathbf{v}_h), \qquad (3.24) \\ f_\Gamma(\mathbf{v}_h) =& -(\gamma \kappa \mathbf{n}_\Gamma, \langle \mathbf{v}_h \rangle)_{\Gamma_h^n} \qquad (3.25) \end{aligned}$$

with given initial condition $\mathbf{u}(\boldsymbol{x},0)=\mathbf{u}_0$. Here, the integrals are evaluated as

$$(\mathbf{u}_h,\mathbf{v}_h)_\Omega=\sum_{i=1}^{2}(\mathbf{u}_{h,i},\mathbf{v}_{h,i})_{\Omega^n_{h,i}}\,. \tag{3.26}$$

And the penalty terms to enforce the interface and boundary conditions are scaling with the mesh size h as follows: $\lambda_{\partial\Omega}(h)=\frac{\hat{\lambda}_{\partial\Omega}}{h}$, $\lambda_\Gamma(h)=\frac{\hat{\lambda}_\Gamma}{h}$. The stabilisation terms in J consist of jump penalty terms over facets of the active meshes. Inspired by [47, 8, 5], we use the following stabilisation terms

$$j_\beta(\mathbf{u}_h,\mathbf{v}_h)=\gamma_u\sum_{i=1}^{2}\sum_{F\in\mathcal{F}_I^i}h^2\rho_i\left\|\boldsymbol{\beta}_h\cdot\boldsymbol{n}_F\right\|_{\infty,F}\left(\left[\!\left[\nabla\mathbf{u}_h\right]\!\right]_{\boldsymbol{n}_F},\left[\!\left[\nabla\mathbf{v}_h\right]\!\right]_{\boldsymbol{n}_F}\right)_F, \tag{3.27}$$

$$j_{div}(\mathbf{u}_h,\mathbf{v}_h)=\gamma_{div}\sum_{i=1}^{2}\sum_{F\in\mathcal{F}_I^i}h^2\rho_i\left\|\boldsymbol{\beta}_h\right\|_{\infty,K}\left(\left[\!\left[\nabla\cdot\mathbf{u}_h\right]\!\right],\left[\!\left[\nabla\cdot\mathbf{v}_h\right]\!\right]\right)_F, \tag{3.28}$$

$$j_p(p_h,q_h)=\gamma_p\sum_{i=1}^{2}\sum_{F\in\mathcal{F}_I^i}\frac{h^3}{\eta_i\max(Re^i_K,1)}\left(\left[\!\left[\nabla p_h\right]\!\right]_{\boldsymbol{n}_F},\left[\!\left[\nabla q_h\right]\!\right]_{\boldsymbol{n}_F}\right)_F, \tag{3.29}$$

$$Re^i_K=\frac{h\rho_i\left\|\boldsymbol{\beta}_h\right\|_{\infty,K}}{\eta_i}. \tag{3.30}$$

Additionally, we use a ghost penalty regularisation for the velocity, which is required at low Reynolds numbers

$$g_u(\mathbf{u}_h,\mathbf{v}_h)=\gamma_{gu}\sum_{i=1}^{2}\sum_{F\in\mathcal{F}_G^i}\eta_i h\left(\left[\!\left[\nabla\mathbf{u}_h\right]\!\right]_{n_F},\left[\!\left[\nabla\mathbf{v}_h\right]\!\right]_{n_F}\right)_F. \tag{3.31}$$

Here, $\left[\!\left[\nabla x\right]\!\right]_{\boldsymbol{n}_F}$ denotes the normal jump of the quantity x over the facet, F, defined as $\left[\!\left[\nabla x\right]\!\right]_{\boldsymbol{n}_F}=\nabla x|_{K_F^+}\boldsymbol{n}_F-\nabla x|_{K_F^-}\boldsymbol{n}_F$, where $\boldsymbol{n}_F$ denotes a unit normal to the facet F with fixed but arbitrary orientation. The stabilisation parameters γ_p, γ_u, γ_{div}, γ_{gu} have to be chosen large enough. The term j_β stabilises the convective terms which is necessary for high Reynolds numbers, j_{div} ensures additional control on the incompressibility condition necessary at high Reynolds numbers, j_p ensures the inf-sup stability for an equal order approximation of velocity and pressure and prevents ill-conditioning. The term j_p is required for low and high Reynolds number flows. In contrast, the terms j_β and j_{div} can be omitted for low Reynolds number flows, i.e. $\gamma_u=\gamma_{div}=0$ can be chosen. The term g_u ensures stability independent of the cut location and prevents ill-conditioning for the velocity.
In addition to stabilising the numerical scheme, facet terms (3.27) – (3.31) extend the velocity and the pressure from the physical domains $\Omega_{h,i}$ to the fictitious domains $\Omega^*_i(t_n)$. This enables us to define the velocity of the previous time step in the current domains as

$$\boldsymbol{\beta}_h(\boldsymbol{x})=\begin{cases}\mathbf{u}^{n-1}_{h,1}(\boldsymbol{x}) & \text{for } \phi(\boldsymbol{x},t_n)\leq 0,\\ \mathbf{u}^{n-1}_{h,2}(\boldsymbol{x}) & \text{for } \phi(\boldsymbol{x},t_n)>0.\end{cases} \tag{3.32}$$

To ensure that the domains $\Omega_{h,i}^{n+1}$ are contained in the fictitious domains of $\Omega_i^*(t_n)$, we choose the δ neighbourhood as the domain formed by all intersected elements and their neighbouring elements. That means a band of one cell layer on each side of the intersected elements resulting in a δ-neighbourhood of three cell bands (see Figure 5.4). The interface strip thickness can also be chosen dependent on the velocity at the interface, e.g. $\delta \geq \|\mathbf{u} \cdot \boldsymbol{n}_\Gamma\|_{\infty,\Gamma_h} \Delta t$ [32] to ensure that $\Omega_{h,i}^{n+1} \subset \Omega_i^*(t_n)$ and the level set function can be used to define the δ-neighbourhood (if the level set function is sufficiently close to a signed distance function). However, as we are facing time step restrictions due to the decoupling of the Navier-Stokes equation and the level set advection as well as from the linearisation of the convective term, the 3-cell band δ-neighbourhood is numerically shown to be thick enough.

3.3.3 Discretised Level Set Advection

As mentioned above we discretise the level set function using a continuous quadratic finite element space, $\mathcal{W}_h$. We stabilise the advection problem (3.14) in space with streamline diffusion (SUPG). The discretised advection problem reads: Find $\phi_h^n \in \mathcal{W}_h$, such that for all $\delta\phi \in \mathcal{W}_h$

$$a_\phi(\phi_h^n, \delta\phi) = l_\phi(\delta\phi) \tag{3.33}$$

with

$$\begin{aligned} a_\phi(\phi_h^n, \delta\phi) &= \left(\frac{\phi_h^n}{\Delta t} + \theta \mathbf{u}_h^n \cdot \nabla \phi_h^n, \delta\phi + \tau_{SD}(\mathbf{u}_h^n \cdot \nabla \delta\phi) \right)_\Omega, \\ l_\phi(\delta\phi) &= \left(\frac{\phi_h^{n-1}}{\Delta t} + (\theta - 1)\boldsymbol{\beta}_h \cdot \nabla \phi_h^{n-1}, \delta\phi + \tau_{SD}(\mathbf{u}_h^n \cdot \nabla \delta\phi) \right)_\Omega \end{aligned} \tag{3.34}$$

with the streamline diffusion parameter [24]

$$\tau_{SD} = 2 \left(\frac{1}{\Delta t^2} + \frac{|\mathbf{u}_h^n \cdot \mathbf{u}_h^n|}{h^2} \right)^{-\frac{1}{2}} \tag{3.35}$$

and initial condition $\phi_h^0 = \phi_0$ and Dirichlet conditions on inflow boundaries.

3.4 Stabilised Scheme for the Surface Tension Force

In this section, we introduce a scheme to smoothen the curvature approximation of the fluid interface. This smoothing is used to reduce the magnitude of spurious velocity modes which arise from errors in the curvature computation. This is particularly important for flows in which surface tension forces dominate viscous forces as these spurious velocities can be the source of numerical instabilities.
The surface tension force term

$$f_\Gamma(\mathbf{v}_h) = -(\gamma \kappa \mathbf{n}_\Gamma, \langle \mathbf{v}_h \rangle)_{\Gamma_h^n} \tag{3.36}$$

contains the mean curvature $H := \kappa \boldsymbol{n}_\Gamma$ which can be reformulated in terms of the Laplace Beltrami operator as (see e.g. [23],[28], [19])

$$H = \kappa \boldsymbol{n}_\Gamma = -\Delta_\Gamma id_\Gamma. \tag{3.37}$$

Here, id_Γ denotes the identity mapping on Γ and $\Delta_\Gamma = \nabla_\Gamma \cdot \nabla_\Gamma$ is the Laplace-Beltrami operator with ∇_Γ denoting the tangential gradient given by

$$\nabla_\Gamma = P_\Gamma \cdot \nabla, \quad P_N = I - \boldsymbol{n}_\Gamma \otimes \boldsymbol{n}_\Gamma, \tag{3.38}$$

where I is the identity matrix and $\otimes$ is the outer product. Note that, $\nabla_\Gamma id_\Gamma = P_\Gamma \cdot \nabla id_\Gamma = P_\Gamma I = P_\Gamma$.

We will now replace the mean curvature vector $H = \kappa \boldsymbol{n}_\Gamma$ with a smoothened discrete curvature [24, 23, 10], H_h, as detailed in the following. Let $[\mathcal{H}_h^n]^d$ denote the vector-valued continuous piecewise linear finite element space over the domain formed by the band of all intersected elements at time $t = t_n$. Then, we determine a smoothened mean curvature vector $H_h \in [\mathcal{H}_h^n]^d$ by solving

$$\begin{aligned} (H_h, \delta H) + J_H(H_h, \delta H) &= (P_\Gamma, \nabla_\Gamma \delta H)_{\Gamma_h^n}, \quad \forall \delta H_h \in [\mathcal{H}_h^n]^d, \\ J_H(H_h, \delta H) &= \sum_{F \in \mathcal{F}_{int}(t_n)} \gamma_H \left([\![\nabla H_h]\!]_{\boldsymbol{n}_F}, [\![\nabla \delta H]\!]_{\boldsymbol{n}_F} \right)_F . \end{aligned} \tag{3.39}$$

Here, γ_H is a positive penalty parameter. This stabilised mean curvature vector H_h enters the right hand side of the two-phase Navier-Stokes problem (3.25) in the surface tension force term as

$$f_\Gamma(\mathbf{v}_h) = -(\gamma H_h, \langle \mathbf{v}_h \rangle)_{\Gamma_h^n}. \tag{3.40}$$

4 Coupled Formulation

Algorithm 1 summarises the two-phase flow cut finite element scheme presented in the previous sections.

5 Numerical Results

The proposed two-phase flow solver is implemented in the CutFEM library [9] based on FEniCS [1, 21].

5.1 Rising Bubble Benchmark

In this section, we validate our numerical scheme on the well known two-dimensional rising bubble benchmark presented in [29]. In this benchmark, a circular gas bubble with radius $r = 0.25$ and centre point $\boldsymbol{x}_m = (0.5, 0.5)^T$ is surrounded by a higher density fluid in a container occupying domain $\Omega = [0, 1] \times [0, 2]$. Due to buoyancy forces caused by gravity, $\boldsymbol{g} = (0, -0.98)^T$, the lighter gas bubble rises in the container. At the top and bottom boundary, we set homogenous Dirichlet conditions for the velocity and at the sides, we

Algorithm 1 CutFEM Two-Phase Flow Algorithm

Set $t = 0$, $\mathbf{u}_h^0 = \mathbf{u}_0$, $\phi_h^0 = \phi_0$.
while $t \leq T$ **do** ▷ T is the final time
    Determine Ω_h^n and Γ_h^n through intersection computations of zero level set with background mesh.
    Compute normal $\boldsymbol{n}_\Gamma$ using (3.18).
    Compute the stabilised mean curvature vector H_h using (3.39).
    procedure TWO-PHASE NAVIER-STOKES($\mathbf{u}_h^{n-1}$, $\boldsymbol{n}_\Gamma$, H_h, Ω_h^n, Γ_h^n)
        Solve the Navier-Stokes problem (3.21).
        return $\mathbf{u}_h^n$, p_h^n.
    end procedure
    procedure LEVEL SET ADVECTION($\mathbf{u}_h^n$)
        Solve advection problem (3.34) for level set.
        Reinitialise level set function using fast marching scheme [19, 18].
        Apply mass correction using volume correction [19, 18].
        return ϕ_h^n.
    end procedure
    $t = t + \Delta t$, $\phi_h^{n-1} = \phi_h^n$, $\mathbf{u}_h^{n-1} = \mathbf{u}_h^n$.
end while

set slip conditions, i.e. $\mathbf{u} \cdot \boldsymbol{n} = 0$ (see schematics in Figure 5.1). We choose a timestep of $\Delta t = h_{min}/2$, where h_{min} is the smallest mesh size h in the background mesh $\mathcal{T}_h$ and compute the solution over time interval $[0, 3]$. As in the original benchmark, we study two cases as summarised in Table 5.1. In the first test case the rising bubble deforms into an elliptical shape and in the second test case the bubble develops long liquid strands. Like in the original benchmark, we evaluate the results in terms of the mean rise velocity

$$V_c = \frac{\int_{\Omega_1} u_y \, dx}{\int_{\Omega_1} 1 \, dx}, \tag{5.1}$$

where u_y is the y-component of the velocity $\mathbf{u}$; the y-component of the center of mass

$$Y_c = \frac{\int_{\Omega_1} y \, dx}{\int_{\Omega_1} 1 \, dx} \tag{5.2}$$

and the degree of circularity [46]

$$\bar{c} = \frac{2\sqrt{\pi \int_{\Omega_1} 1 \, dx}}{\int_\Gamma 1 \, ds}, \tag{5.3}$$

which compares the circumference of the bubble to the circumference of a circle with the same area as the bubble.

For both test cases, we set the penalty terms to $\gamma_u = 0.01$, $\gamma_p = 0.1$, $\gamma_{div} = 0.01$, $\gamma_{gu} = 0.01$, $\gamma_H = 0.001$, $\hat{\lambda}_{\partial\Omega} = 10.0$, $\hat{\lambda}_\Gamma = 10.0$ unless specified otherwise. We investigate the results in comparison to the benchmark values for mesh sizes $h = \{1/40, 1/80, 1/160\}$. For test case 1, we compare our results to the finest mesh results ($h = 1/320$) of the code TP2D [44] and for test case 2, we compare our results to the finest mesh results ($h = 1/160$) of FreeLIFE [40] from the rising bubble benchmark [29].

Test case 1 Figure 5.2 shows that we achieve excellent agreement between the reference solution of TP2D and our cut finite element Navier-Stokes scheme for the final bubble shape at $t = 3$, for the evolution of the centre of mass, for the rise velocity and for the circularity.
Figure 5.3 shows the positive and negative effects of the smoothened curvature computation (3.39). Considering the same material parameters as in test case 1 and setting the gravity to zero reveals well known spurious velocities caused by the surface tension force term (3.36). Compared to the direct application of the Laplace Beltrami operator (see Figure 5.3(b))

$$f_\Gamma(\mathbf{v}_h) = -(\gamma\kappa\mathbf{n}_\Gamma, \langle\mathbf{v}_h\rangle)_{\Gamma_h^n} = -(\gamma\nabla_\Gamma id_\Gamma, \langle\nabla_\Gamma\mathbf{v}_h\rangle)_{\Gamma_h^n}, \tag{5.4}$$

the smoothened mean curvature force term (3.40) helps control the magnitude of the spurious velocities (see Figure 5.3(c)). However, if the curvature stabilisation parameter γ_H is chosen too large the smoothened curvature approximation can change the shape of the bubble significantly. As shown in Figure 5.3(a), the bubble is significantly more deformed for high values of γ_H. It is therefore crucial to choose a small value for γ_H. The chosen value $\gamma_H = 10^{-3}$ yields an appropriate reduction in spurious velocities with no negative effects on the benchmark values.
Figure 5.4 shows the ghost penalty extensions of the velocity and the pressure from the physical domains to the fictitious domains for $t = 1.225$. We observe that the ghost penalties provide a rich and stable extension and yield a stable solution independent of the cut location of the fluid interface. The extensions and doubling of cells in the interface layer enable us to capture sharp pressure jumps and velocity kinks at the interface (see Figure 5.5).

Test case 2 Figure 5.6 shows the center of mass, rise velocity and circularity with mesh refinement in comparison to FreeLIFE for mesh size $h = 1/160$. We achieve excellent agreement of our results with FreeLIFE until $t \approx 1.5$, which coincides with the onset of the liquid strand formation in the wake of the bubble (see Figure 5.7). Similar discrepancies between the results were observed in the original benchmark [29] and no agreement for the formation and size of the liquid filaments could be reached in [29]. This is due to the fact that the liquid strands breaking or not breaking is highly dependent on the mesh resolution. In our case once thin liquid filaments are contained within one element this part of the strand is no longer detected and will cause the strand to break. The breaking of the liquid strands causes oscillations in the circularity values (see Figure 5.6 (c) for $t > 2.5$). For $h = 1/40$ and $h = 1/80$, the liquid strands break while for $h = 1/160$

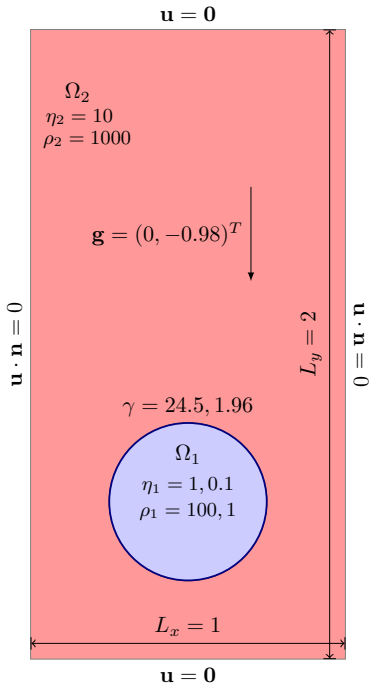
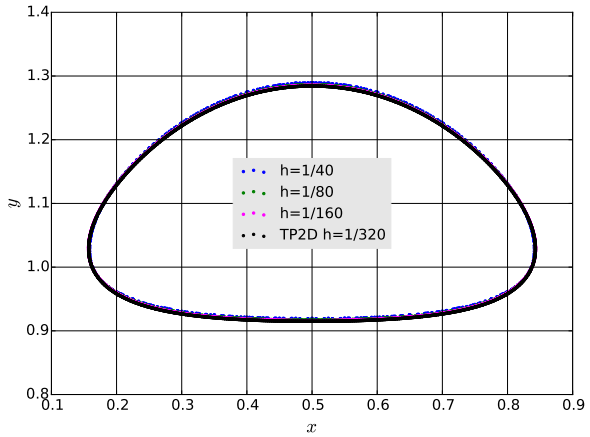


Figure 5.1: Schematics of rising bubble benchmark.

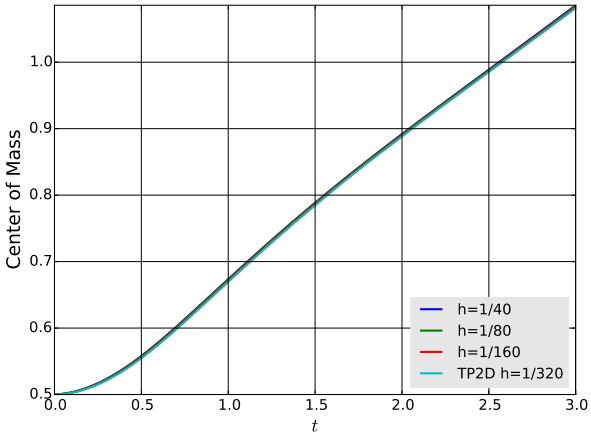
Case	γ	ρ_1	ρ_2	η_1	η_2
1	24.5	100	1000	1	10
2	1.96	1	1000	0.1	10

Table 5.1: Summary of material parameters for rising bubble benchmark.

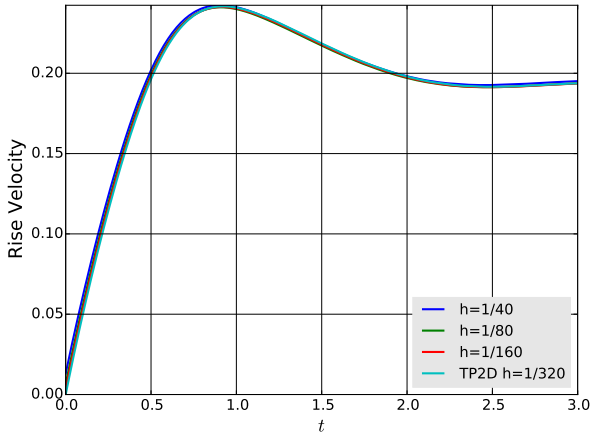
they do not break (see Figure 5.7(a)). Figure 5.7(b) shows the influence of the velocity stabilisation parameters $\gamma_u = \gamma_{div} = \gamma_{gu} = \{0.001, 0.01, 1\}$ onto the liquid filaments for the final time $t = 3.0$ and $h = 1/80$. We observe that increasing the velocity stabilisation parameters leads to a thickening of the liquid strands and that no break up occurs for $\gamma_u = 1$ for $h = 1/80$. This can be explained by interpreting the gradient jump penalties, j_u, j_{div} and g_u as a small amount of additionally viscosity which has this stabilising effect onto the liquid filaments. Note that while an increase in the velocity parameters changes the thickness of the strands, it does not significantly impact the bulk part of the bubble



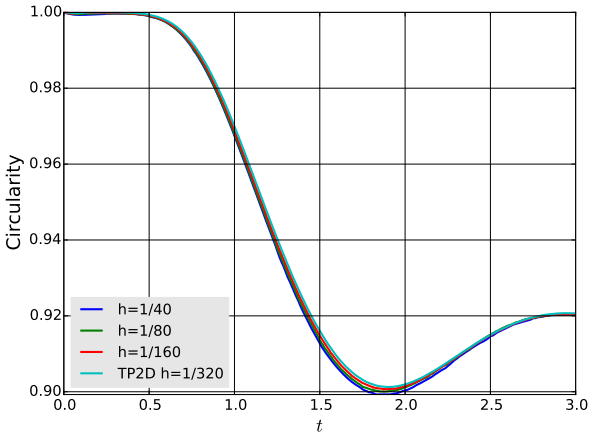
(a) Bubble shape at time $t = 3$.



(b) Center of Mass.

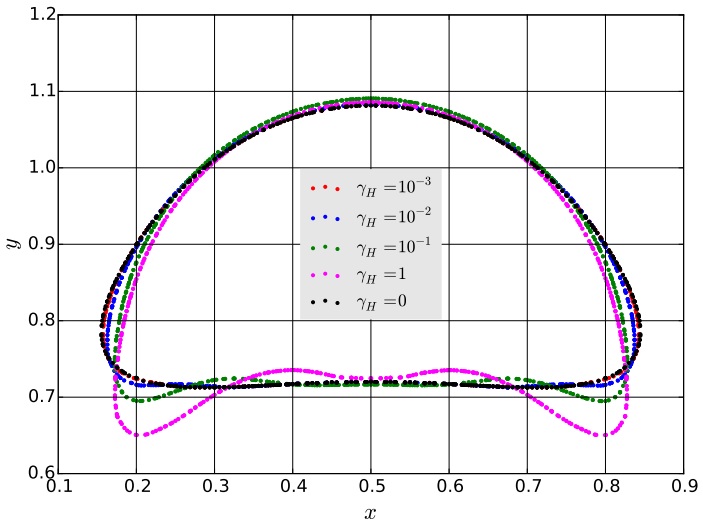


(c) Rise Velocity.

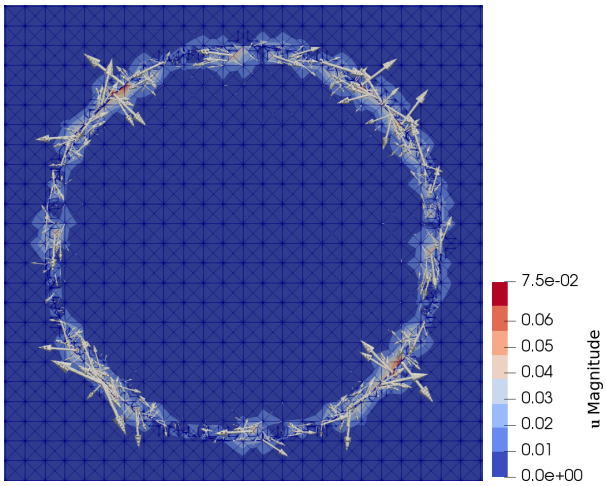


(d) Circularity.

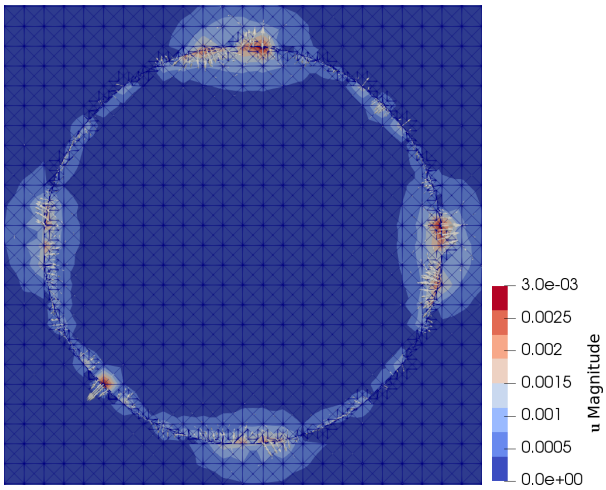
Figure 5.2: Numerical results of rising bubble benchmark for test case 1.



(a) Bubble Shape with curvature stabilisation at $t = 1.9$.

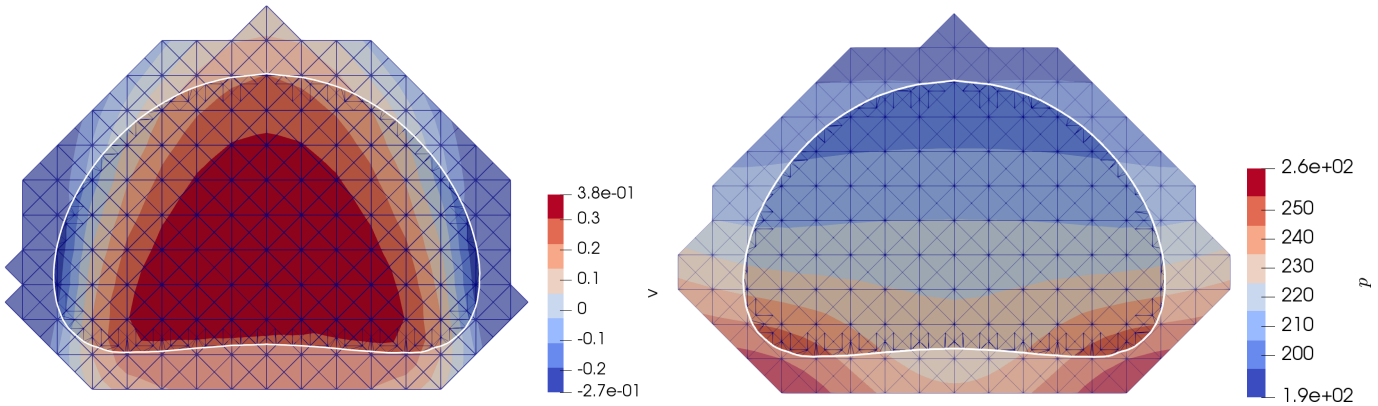


(b) Velocity magnitude for Laplace Beltrami operator.



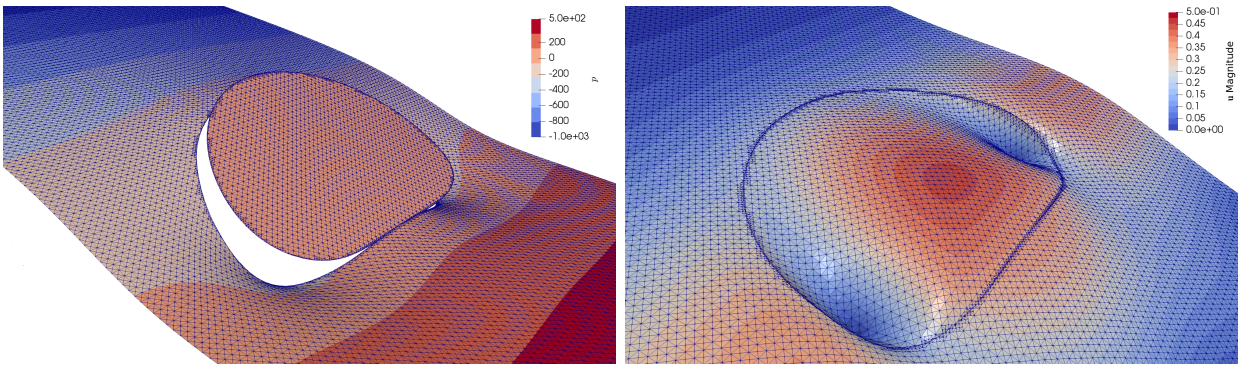
(c) Velocity magnitude for stabilised mean curvature.

Figure 5.3: Effect of curvature stabilisation on the bubble shape (a) and the reduction in spurious velocities for rising bubble test case 1 with zero gravity at time $t = 0.125$ from Laplace Beltrami operator ($\gamma_H = 0$) to stabilised curvature computation ($\gamma_H = 0.001$) (b),(c).



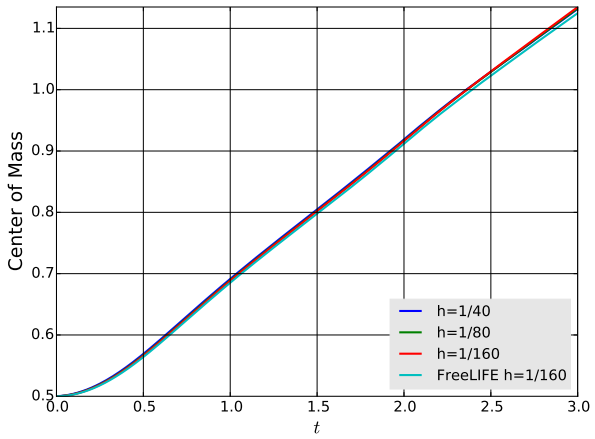
(a) (b)

Figure 5.4: Extended y-component of the velocity (a) and the pressure (b) for test case 1 for a coarse mesh with $h = 1/20$ at time $t = 1.225$.

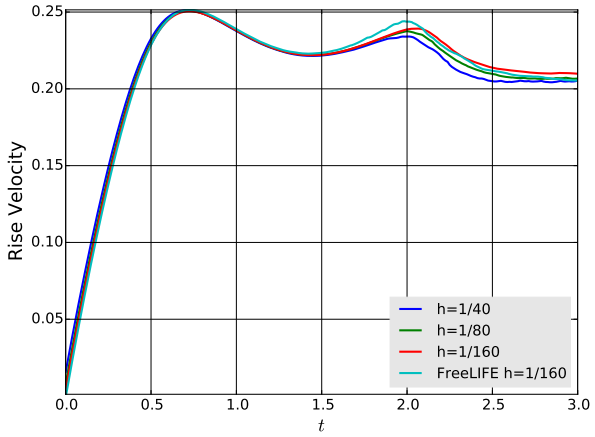


(a) (b)

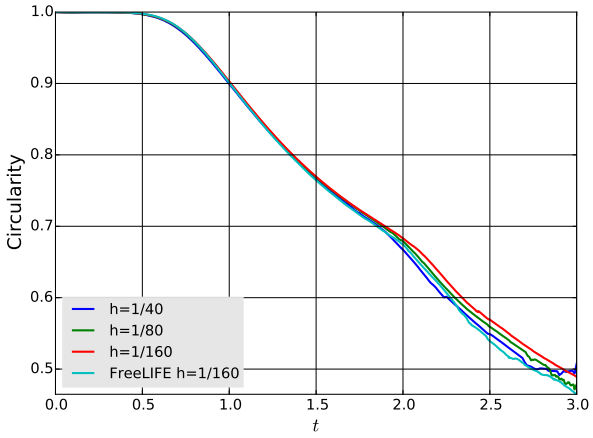
Figure 5.5: The cut finite element scheme is capable of representing kinks in the velocity and jumps in the pressure at the interface. Here, the pressure jump (a) and the velocity kink in the magnitude of the velocity (b) is shown for test case 1 at time $t = 1$ for $h = 1/80$.



(a) Center of Mass.

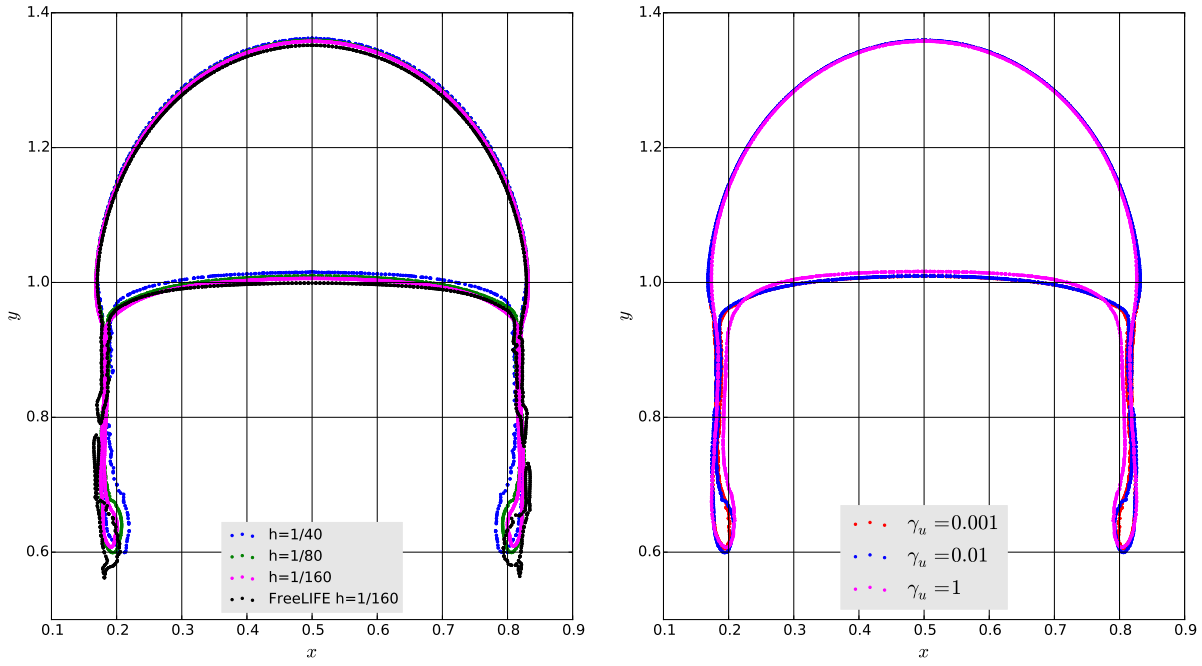


(b) Rise Velocity.



(c) Circularity.

Figure 5.6: Center of mass, rise velocity and circularity for test case 2 for mesh refinement in comparison with FreeLIFE $h = 1/160$.



(a) Bubble shape with mesh refinement.

(b) Bubble shape with varying γ_u.

Figure 5.7: Bubble shapes for test case 2 with mesh refinement and with varying velocity stabilisation parameter $\gamma_u = \gamma_{div} = \gamma_{gu}$ for mesh size $h = 1/80$ at $t = 3.0$.

(see Figure 5.7(b)).

5.2 Droplet in microfluidic 5:1:5 contraction and expansion

In this section, we consider the deformation of a single droplet in a 5:1:5 contraction and expansion (see e.g. [12],[25],[27]) flow. We consider a domain decomposed into an entry domain of height 5 and length 5 attached to a narrow channel of height 1 and length 8 and an expansion region of height 5 and length 7 (see Figure 5.8). We consider a droplet placed in the middle of the entry domain $\boldsymbol{x}_m = (2.5, 2.5)^T$ of diameter larger than the narrow channel ($d = 1.2$). We round the corners of the contraction and expansion with circles of radius $r = 0.5$ to prevent stress singularities at these corners. At the top and bottom boundary, we set homogenous Dirichlet boundary condition for the velocity, i.e. $\mathbf{u} = \mathbf{0}$ on $\partial\Omega_D$. At the left boundary, we set a velocity inflow profile of

$$\mathbf{u} = (0.048\,(5 - y)y, 0)^T \text{ on } \partial\Omega_{in}. \tag{5.5}$$

The scaling factor, 0.048, is chosen to obtain an average velocity of $\bar{U} = 1$. At the right boundary, $\partial\Omega_{out}$, we set homogenous Neumann conditions for the velocity, i.e. $\frac{\partial \mathbf{u}}{\partial \boldsymbol{n}} = 0$, and $p = 0$. The non-dimensional number, which best characterises this flow configuration is the Capillary number ([12],[25],[27]),

$$Ca = \frac{\eta_2 \bar{U}}{\gamma}, \tag{5.6}$$

of the surrounding fluid. The capillary number expresses the ratio of viscous to capillary forces. Here, L is the height of the narrow channel, i.e. $L = 1$. As our intend is to simulate conditions in a microfluidic device, where inertial forces are small due to the small dimensions of the geometry, we choose $\rho_1 = \rho_2 = 1.0$. Additionally, capillary forces tend to dominate viscous forces in microfluidic devices and hence $Ca < 1$. To account for this force balance, we set the surface tension coefficient to $\gamma = 20$ and choose $\eta_2 < 20$. With the aim of observing different bubble shapes, we fix $\eta_1 = 1$ and vary η_2 to create the following three different cases

$$\text{Case 1: } \eta_2 = \quad 10 \quad \Rightarrow Ca = 0.5, (\eta_1 < \eta_2), \tag{5.7}$$

$$\text{Case 2: } \eta_2 = \quad 1 \quad \Rightarrow Ca = 0.05, (\eta_1 = \eta_2), \tag{5.8}$$

$$\text{Case 3: } \eta_2 = \quad 0.1 \quad \Rightarrow Ca = 0.005, (\eta_1 > \eta_2). \tag{5.9}$$

The mesh consisting of a total of 34808 elements is refined once around the narrow channel and refined again towards the channels walls (see Figure 5.8). We choose a time step size of $\Delta t = h_{min}/2 = 0.0062$ and compute the solution in the time interval $t \in [0, 17]$. We set the penalty parameters to $\gamma_u = \gamma_{div} = \gamma_{gu} = 0.1$, $\gamma_p = 0.1$, $\gamma_H = 0.001$, $\hat{\lambda}_{\partial\Omega} = 10.0$, $\hat{\lambda}_\Gamma = 10.0$.
Figure 5.9 shows the evolution of the droplet through the contraction and expansion regions. The droplet for case 1 undergoes a large deformation and forms a "bullet shape"

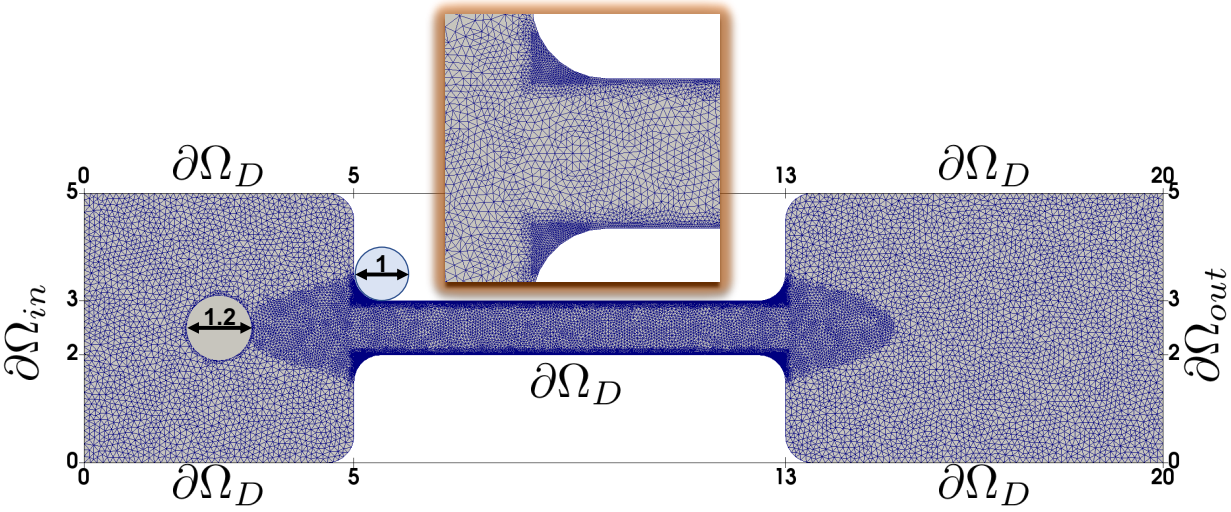
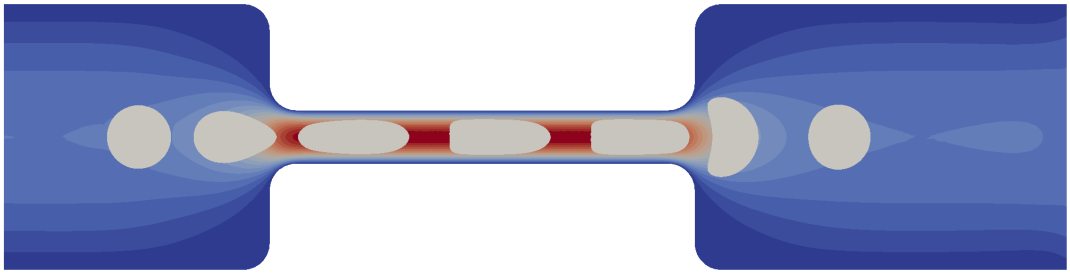


Figure 5.8: Geometry and mesh for the 5:1:5 contraction-expansion problem.

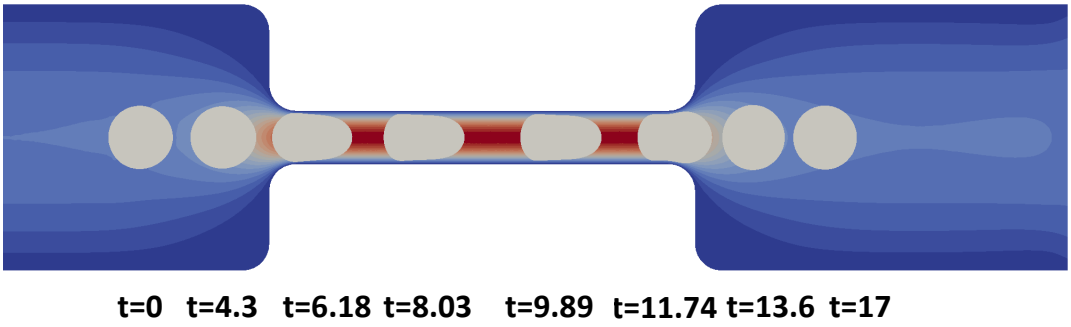
inside the narrow channel as well as a "half moon shape" in the expansion region. For cases 2 and 3 the droplet takes on a rounder shape inside the channel and relaxes faster to the circular shape in the expansion region. The time interval needed for the droplet to travel through the narrow channel increases from case 1 to case 3, i.e. the mean velocity decreases from case 1 to case 3. Figure 5.10 shows the pressure profiles and droplet shapes for the droplet in the middle of the narrow channel. We observe a decrease in the fluid layer thickness between the droplet and the wall from case 1 to case 3. The droplet shape becomes increasingly circular from case 1 to case 3. A high pressure forms in the thin liquid layer between the droplet and the channel walls which prevents the droplet from touching the walls. In the rear of the droplet, we have a pressure concentration which sharpens with increasing Capillary number. We have a higher curvature in the front of the droplet than in the rear, which for case 1 yields the bullet shape. This contrast in curvatures decreases with decreasing Capillary number. Figure 5.10 shows that the presented cut finite element method is capable of capturing the jumps in the pressure profiles for low Capillary number without spurious oscillations and is capable of resolving the thin liquid layer between the droplet and the wall.

6 Conclusions

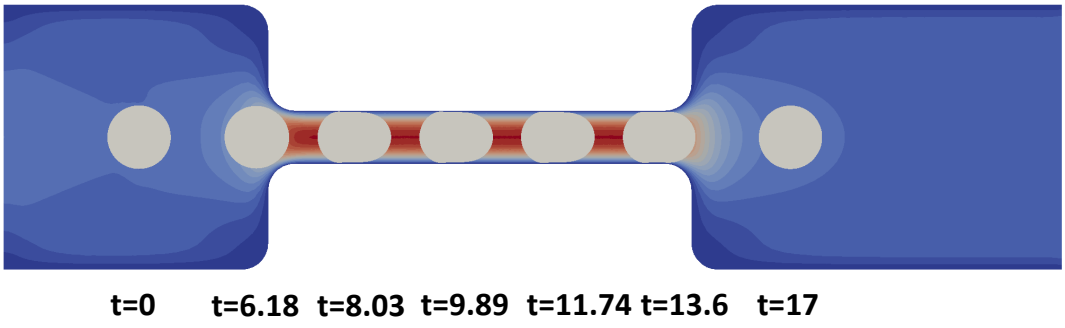
We have presented a cut finite element method for two-phase Navier-Stokes problems with a unified continuous interior penalty stabilisation strategy. Ghost penalties are used to extend the velocity and pressure from physical domains to extended fictitious domains for each phase to guarantee a well defined velocity field of the past time step in the current



(a) Case 1, $Ca = 0.5$.



(b) Case 2, $Ca = 0.05$.



(c) Case 3, $Ca = 0.005$.

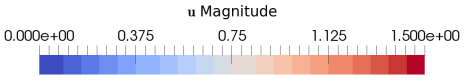
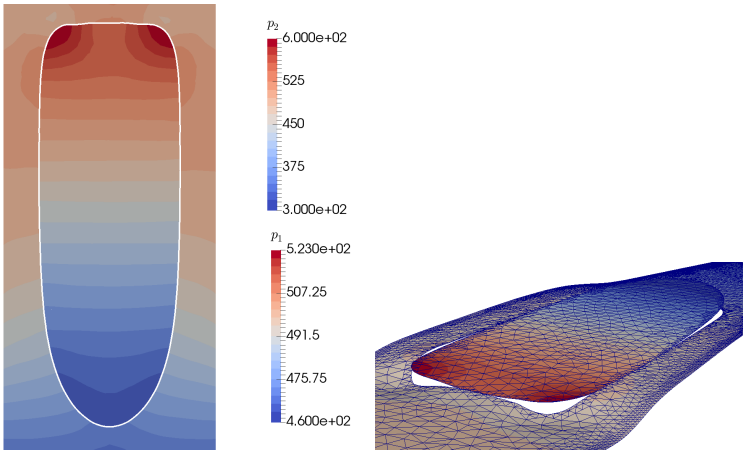
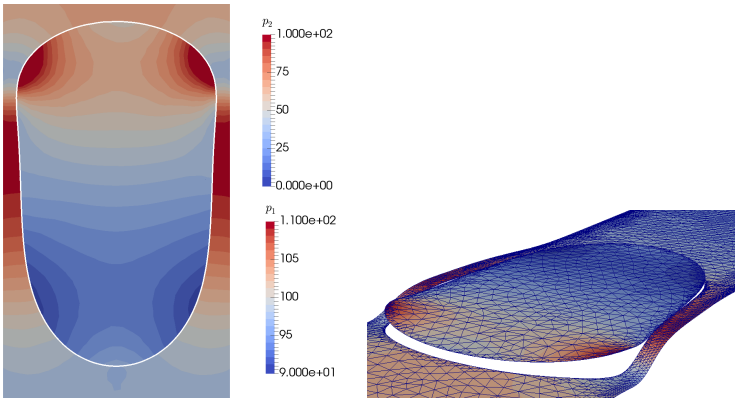


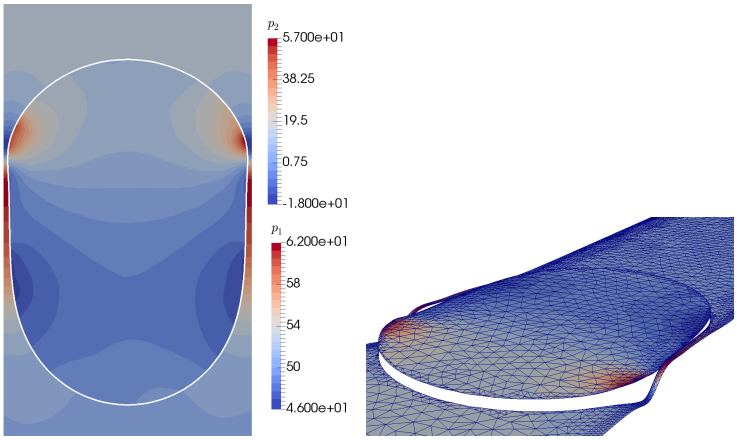
Figure 5.9: Development of droplet shapes in 5:1:5 contraction-expansion for $Ca = 0.5, 0.05, 0.005$.



(a) Case 1, $Ca = 0.5$.



(b) Case 2, $Ca = 0.05$.



(c) Case 3, $Ca = 0.005$.

Figure 5.10: Pressure profiles and droplet shape in middle of narrow channel for 5:1:5 contraction-expansion with decreasing Capillary number.

geometry. In addition, we have employed a smoothened curvature computation to reduce spurious velocity oscillations.
We have validated our scheme on two problems: the rising bubble benchmark problem [29] and a droplet in a 5:1:5 contraction and expansion microfluidic flow. We have found excellent agreement between our scheme and the rising bubble benchmark. However, we have shown that the curvature stabilisation parameter needs to be chosen with care in order to avoid unphysical shape deformations. We have demonstrated that the strengths of the presented approach are its capability to represent pressure jumps and velocity kinks, to resolve thin lubrication layers as well as its stability for low Capillary number flows.

Acknowledgements

The authors gratefully acknowledge the financial support provided by the Welsh Government and Higher Education Funding Council for Wales through the Sêr Cymru National Research Network in Advanced Engineering and Materials under grants NRNG06 and NRN102. Susanne Claus would also like to thank Erik Burman and Sarah Zahedi for numerous inspiring and helpful discussions.

References

[1] M. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M.E. Rognes, and G.N. Wells. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100):9–23, 2015.

[2] D.M. Anderson, G.B. McFadden, and A.A. Wheeler. Diffuse-interface methods in fluid mechanics. *Annual review of fluid mechanics*, 30(1):139–165, 1998.

[3] J.U. Brackbill, D.B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *Journal of computational physics*, 100(2):335–354, 1992.

[4] E. Burman. Ghost penalty. *Comptes Rendus Mathematique*, 348(21-22):1217–1220, 2010.

[5] E. Burman and M.A. Fernández. Continuous interior penalty finite element method for the time-dependent Navier–Stokes equations: space discretization and convergence. *Numerische Mathematik*, 107(1):39–77, 2007.

[6] E. Burman and M.A. Fernández. An unfitted Nitsche method for incompressible fluid–structure interaction using overlapping meshes. *Computer Methods in Applied Mechanics and Engineering*, 279:497–514, 2014.

[7] E. Burman and P. Zunino. Numerical approximation of large contrast problems with the unfitted Nitsche method. In *Frontiers in Numerical Analysis-Durham 2010*, pages 227–282. Springer, 2011.

[8] E. Burman, M.A. Fernández, and P. Hansbo. Continuous interior penalty finite element method for Oseen's equations. *SIAM J. Numer. Anal.*, 44(3):1248–1274, 2006.

[9] E. Burman, S. Claus, P. Hansbo, M.G. Larson, and A. Massing. CutFEM: discretizing geometry and partial differential equations. *Int. J. Numer. Meth. Eng.*, 2014.

[10] M. Cenanovic, P. Hansbo, and M.G. Larson. Finite element procedures for computing normals and mean curvature on triangulated surfaces and their use for mesh refinement. *arXiv preprint arXiv:1703.05745*, 2017.

[11] J. Chessa and T. Belytschko. An enriched finite element method and level sets for axisymmetric two-phase flow with surface tension. *International journal for numerical methods in engineering*, 58(13):2041–2064, 2003.

[12] C. Chung, J.M. Kim, M.A. Hulsen, K.H. Ahn, and S.J. Lee. Effect of viscoelasticity on drop dynamics in 5:1:5 contraction/expansion microchannel flow. *Chemical Engineering Science*, 64(22):4515–4524, 2009.

[13] S. Claus and P. Kerfriden. A stable and optimally convergent LaTIn-CutFEM algorithm for multiple unilateral contact problems. *International Journal for Numerical Methods in Engineering*, 113(6):938–966, 2018.

[14] S. Claus, S. Bigot, and P. Kerfriden. A CutFEM method for Stefan-Signorini problems with application in pulsed laser ablation. *arXiv preprint arXiv:1805.01925*, 2018.

[15] J. Donea, A. Huerta, J.-Ph. Ponthot, and A. Rodriguez-Ferran. *Arbitrary Lagrangian–Eulerian Methods*, chapter 14. John Wiley & Sons Ltd., 2004.

[16] T.-P. Fries. The intrinsic XFEM for two-fluid flows. *International Journal for Numerical Methods in Fluids*, 60(4):437–471, 2009.

[17] T.-P. Fries and T. Belytschko. The intrinsic XFEM: a method for arbitrary discontinuities without additional unknowns. *International journal for numerical methods in engineering*, 68(13):1358–1385, 2006.

[18] S. Groß and A. Reusken. *Numerical methods for two-phase incompressible flows*. Springer, 2011.

[19] S. Groß, V. Reichelt, and A. Reusken. A finite element based level set method for two-phase incompressible flows. *Computing and visualization in science*, 9(4):239–257, 2006.

[20] S. Groß, T. Ludescher, M. Olshanskii, and A. Reusken. Robust preconditioning for XFEM applied to time-dependent Stokes problems. *SIAM Journal on Scientific Computing*, 38(6):A3492–A3514, 2016.

[21] J.S. Hale, L. Li, C.N. Richardson, and G.N. Wells. Containers for portable, productive, and performant scientific computing. *Computing in Science & Engineering*, 19(6):40–50, 2017.

[22] P. Hansbo, M.G. Larson, and S. Zahedi. A cut finite element method for a Stokes interface problem. *Applied Numerical Mathematics*, 85:90–114, 2014.

[23] P. Hansbo, M.G. Larson, and S. Zahedi. Stabilized finite element approximation of the mean curvature vector on closed surfaces. *SIAM Journal on Numerical Analysis*, 53(4):1806–1832, 2015.

[24] P. Hansbo, M.G. Larson, and S. Zahedi. A cut finite element method for coupled bulk-surface problems on time-dependent domains. *Computer Methods in Applied Mechanics and Engineering*, 307:96–116, 2016.

[25] D.J.E. Harvie, J.J. Cooper-White, and M.R. Davidson. Deformation of a viscoelastic droplet passing through a microfluidic contraction. *Journal of Non-Newtonian fluid mechanics*, 155(1-2):67–79, 2008.

[26] F. Heimann, C. Engwer, O. Ippisch, and P. Bastian. An unfitted interior penalty discontinuous Galerkin method for incompressible Navier–Stokes two-phase flow. *International Journal for Numerical Methods in Fluids*, 71(3):269–293, 2013.

[27] V.T. Hoang, J. Lim, C. Byon, J.M. Park, et al. Three-dimensional simulation of droplet dynamics in planar contraction microchannel. *Chemical Engineering Science*, 176:59–65, 2018.

[28] S. Hysing. A new implicit surface tension implementation for interfacial flows. *International Journal for Numerical Methods in Fluids*, 51(6):659–672, 2006.

[29] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, and L. Tobiska. Quantitative benchmark computations of two-dimensional bubble dynamics. *International Journal for Numerical Methods in Fluids*, 60(11):1259–1288, 2009.

[30] D. Jacqmin. Calculation of two-phase Navier–Stokes flows using phase-field modeling. *Journal of Computational Physics*, 155(1):96–127, 1999.

[31] M. Kirchhart, S. Groß, and A. Reusken. Analysis of an XFEM discretization for Stokes interface problems. *SIAM Journal on Scientific Computing*, 38(2):A1019–A1043, 2016.

[32] C. Lehrenfeld and M.A. Olshanskii. An Eulerian Finite Element Method for PDEs in time-dependent domains. *arXiv preprint arXiv:1803.01779*, 2018.

[33] C. Lehrenfeld and A. Reusken. Optimal preconditioners for Nitsche-XFEM discretizations of interface problems. *Numerische Mathematik*, 135(2):313–332, 2017.

[34] A. Massing, M.G. Larson, A. Logg, and M.E. Rognes. A stabilized Nitsche overlapping mesh method for the Stokes problem. *Num. Math.*, pages 1–29, 2014.

[35] A. Massing, M.G. Larson, A. Logg, M. Rognes, et al. A Nitsche-based cut finite element method for a fluid-structure interaction problem. *Communications in Applied Mathematics and Computational Science*, 10(2):97–120, 2015.

[36] J. Melenk and I. Babuška. The partition of unity finite element method: basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 139(1-4): 289–314, 1996.

[37] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005.

[38] N. Moës, J. Dolbow, and T. Belytschko. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering*, 46 (1):131–150, 1999.

[39] E. Olsson and G. Kreiss. A conservative level set method for two phase flow. *Journal of Computational Physics*, 210(1):225–246, 2005.

[40] N. Parolini and E. Burman. A finite element level set method for viscous free-surface flows. In *Applied and Industrial Mathematics in Italy*, pages 416–427. World Scientific, 2005.

[41] T.-P. Sauerland, H.and Fries. The stable XFEM for two-phase flows. *Computers & Fluids*, 87:41–49, 2013.

[42] B. Schott, U. Rasthofer, V. Gravemeier, and W.A. Wall. A face-oriented stabilized Nitsche-type extended variational multiscale method for incompressible two-phase flow. *International Journal for Numerical Methods in Engineering*, 104(7):721–748, 2015.

[43] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational physics*, 114(1):146–159, 1994.

[44] S. Turek. *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach*, volume 6. Springer, 1999.

[45] S. Unverdi and G. Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics*, 100(1):25–37, 1992.

[46] H. Wadell. Sphericity and roundness of rock particles. *The Journal of Geology*, 41(3): 310–331, 1933.

[47] C. Winkelmann. *Interior penalty finite element approximation of Navier-Stokes equations and application to free surface flows*. PhD thesis, 2007.

[48] M. Winter, B. Schott, A. Massing, and W.A. Wall. A Nitsche cut finite element method for the Oseen problem with general Navier boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 330:220–252, 2018.