# An Exactly Force-Balanced Boundary-Conforming Arbitrary-Lagrangian-Eulerian Method for Interfacial Dynamics

Zekang Cheng[a,b], Jie Li[b,a,*], Ching Y. Loh[c], Li-Shi Luo[c,d,*]

[a]*Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK*
[b]*BP Institute, University of Cambridge, Madingley Road, Cambridge, CB3 0EZ, UK*
[c]*Beijing Computational Sciences Research Center, Beijing, 100193, China*
[d]*Department of Mathematics & Statistics, Old Dominion University, Norfolk, VA 23529, USA*

**Abstract**

We present an interface conforming method for simulating two-dimensional and axisymmetric multiphase flows. In the proposed method, the interface is composed of straight segments which are part of mesh and move with the flow. This interface representation is an integral part of an Arbitrary Lagrangian-Eulerian (ALE) method on an moving adaptive unstructured mesh. Our principal aim is to develop an accurate and robust computational method for interfacial flows driven by strong surface tension and with weak viscous dissipation. We first construct discrete solutions satisfying the Laplace law on a circular/spherical interfaces exactly, *i.e.*, the balance between the surface tension and the pressure jump across an interface is achieved exactly. The accuracy and stability of these solutions are then investigated for a wide range of *Ohnesorge* numbers, Oh. The dimensionless amplitude of the *spurious current* is reduced to machine zero, *i.e.*, on the order of $10^{-15}$ for Oh $\geq 10^{-3}$. Finally, the accuracy and capability of the proposed method are demonstrated through a series of benchmark tests with larger interface deformations. In particular, the method is validated with Prosperitti's analytic results of the bubble/drop oscillations and Peregrine's

*Corresponding author
*Email addresses:* jl305@cam.ac.uk (Jie Li), lluo@odu.edu (Li-Shi Luo)

dripping faucet experiment, in which the values of Oh are small.

*Keywords:* Boundary conforming method, ALE, adaptive moving mesh, fluid-fluid interface, multiphase flows, spurious current.

---

## 1. Introduction

Multiphase flows with fluid-fluid interfaces are ubiquitous in natural phenomena and industrial processes, examples include breaking waves [1], drop impact [2], and spray and atomization [3, 4]. A difficulty in numerical simulations of interfacial flows is due to the great disparities in various competing forces, such as very strong surface tensions *vs.* relatively weak viscous dissipation, as in fluids like water and mercury in air. This presents a great challenge to the accuracy and stability of the numerical simulations. The so-called *spurious current* is a particular numerical artifact due to imbalanced interfacial forces. An inaccurate method may continuously generate energy in a system that cannot be sufficiently damped by the viscous and/or numerical dissipation so the simulation may explode prematurely. This issue is the main focus of this paper and best illustrated through numerical artifacts occurring near a circular/spherical interface.

The key feature of multiphase flows is the presence of interfaces. Depending on the relation between the interface and the computational mesh, the numerical methods are classified into two categories [5]: (i) interface-conforming methods in which the interface is always composed of lines of the mesh system; and (ii) non-conforming methods otherwise. The method presented in this paper falls into the former category, while some popular methods, such as front-tracking method, Volume of Fluid (VOF) method, Level-Set (LS) method and phase-field method [6–12], belong to the latter. Fig. 1 is a simple illustration of these two types of methods: (a) shows an interface-conforming method with an unstructured mesh; (b) and (c) two different non-conforming methods: the front-tracking and volume-tracking methods, respectively. In the two non-conforming methods, the blue lines are mesh lines that do not align with the interface (the red lines). Note that non-conforming methods do not necessarily use fixed meshes. The VOF solver, Gerris [13], and the phase-field method [14] employ adaptive meshing to enhance their computational efficiency. Since the interfaces are not mesh lines, they belong to the class of non-conforming methods.

In an interface-conforming method, the computational mesh is updated

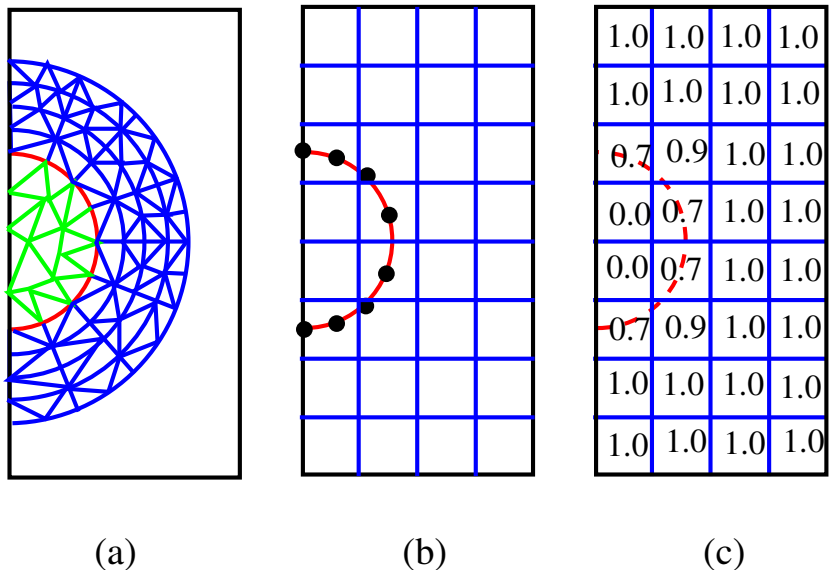


Figure 1: Schematics of different interface-tracking methods: (a) an interface-conforming method using an unstructured mesh; (b) a front-tracking method; (c) a volume-tracking method. The red solid lines in (a) and (b) indicate that the interface is explicitly tracked, while the red dashed line (c) indicates that the interface is only implicitly represented through a marker function.

constantly to follow the evolution of the interface, thus a robust meshing algorithm must be developed to ensure the proper working of this interface-conforming mesh system [15, 16]. Due to its great flexibility, unstructured meshes allows for the generation of high quality meshes. Despite their stringent demands on numerical and programming skills, the interface-conforming methods have a tremendous advantage in dealing with the physics taking place at/across the interface. Firstly, as shown in Fig. 1(a), each cell of the mesh contains only one type of fluid, hence fluid properties such as density and viscosity are well defined, and the discontinuity of fluid properties across interfaces is resolved exactly. Secondly, geometrical quantities such as the unit normal vector and the curvature on the interface can be computed directly with high precision. Finally, the dynamical boundary conditions at the interface appear naturally in finite volume or finite element formulations [17, 18]. Their numerical discretization is simply an evaluation of boundary integrals which can be computed very accurately. In a non-conforming method, because the interface does not align with the mesh lines, accurate

approximation of the boundary conditions can be a challenge, although new algorithms [19–21] have been developed to address the issue.

The foundation of our method is an accurate, efficient, and robust mesh generator which adapts the mesh to follow the motion of the interface. Our FEM method (the flow solver) differs only slightly from an FEM on a fixed mesh. The popular Taylor-Hood element $P_2$-$P_1$ is employed in this work. The conventional approach requires that the pressure is continuous all over the domain (*i.e.*, the $P_1$-$C^0$ element). This is inadequate for multiphase flow computation, because the pressure across the interface is discontinuous in general. Therefore we introduce a new element space, denoted as $P_1d$, for the pressure. In this space, the pressure has a single-value on interior vertices, but double-values on interface vertices. The temporal discretization of the Navier-Stokes equations is semi-implicit: the interface is moved (hence the computational domain) according to the fluid velocity at the old time $t^n$, while the Navier-Stokes equations is solved at the new time level $t^{n+1}$. This choice simplifies the method but restricts the time step to a CFL-type condition. This is not a constraint for us as we are mainly interested in unsteady flows encompassing disparate length-scales of several orders of magnitude.

For simplicity and efficiency, we opt for meshes composed of linear affine triangles, and the interface is represented by a collection of straight segments. The efficacy of this approach can be shown by exactly satisfying the Laplace law for a circular bubble at the discrete level. This analytic solution expresses the equilibrium between the surface tension and the pressure jump across a circular interface. It is a basic solution of multiphase flows but may be difficult to achieve exactly at discrete level. When the Laplace law is not satisfied exactly, instead of being quiescent as the fluid should, a spurious velocity field in interfacial region may be generated by the residual force due to the imbalanced between surface tension and the pressure jump. In a common scenario, when the surface tension is much stronger than the viscous dissipation, simulations may suffer severely from instability and "explode" prematurely. There have been abundant studies on the *spurious* velocity using non-conforming methods [19, 21–25], and recent progress was summarized by Popinet [5]. Exact balance with machine accuracy has been demonstrated for VOF/CSF and level set methods. However, the interface-conforming method has not yet to demonstrate the same capability to eliminate the spurious velocity, and this work is intended to fill this void. To this end, we construct the discrete solutions which the equilibrium between the surface tension and the associated pressure jump is achieved exactly, *i.e.*, the discrete counterpart of

*Laplace* law. The accuracy and stability of these solutions are then verified for a wide range of *Ohnesorge* number Oh (cf. Sec. 3.4 for the definition of Oh). Our numerical experiments show that the dimensionless amplitude of the *spurious* current in our method is on the orders of $10^{-15}$ for Oh $\geq 10^{-3}$. Finally, the accuracy and capability of our method are demonstrated through a series of benchmark tests. In particular, simulations of the bubble/drop oscillations and the dripping faucet are compared with both the theoretical results of Prosperetti [26] and the experiment of Peregrine et al. [27], respectively.

The remainder of the paper is organized as follows. Section 2 discusses the governing equations and associated boundary conditions. Section 3 presents the details of our numerical method. The adaptive mesh algorithm is introduced in Section 3.1, the characteristic method treating the material derivative is explained in Section 3.2 and the finite element method is detailed in Section 3.3. Section 3.4 proves that our numerical method establishes the equilibrium between pressure jump and surface tension across a circular interface exactly at the discrete level, the numerical counterpart of *Laplace law*. This is a crucial prerequisite to eliminate the *spurious* current. In section 4, the accuracy and robustness of our method are demonstrated through a number of test cases. Section 5 concludes the paper. The detailed proof of the equilibrium between pressure jump and surface tension in case of axisymmetric flows is given in the Appendix.

## 2. Governing equations and boundary conditions

We are studying a system of 2 immiscible fluids $L_i$ with densities $\rho_i$, viscosities $\mu_i$, and surface tension coefficient $\sigma$ between the two fluids, $i = 1$, 2, where $\rho_i$ and $\mu_i$ are constants in each fluid. The fluid flow satisfies the *incompressible* Navier-Stokes equations:

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0, \tag{1a}$$

$$\rho \frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} = -\boldsymbol{\nabla} p + \boldsymbol{\nabla} \cdot (\mu \mathbf{D}) + \rho \boldsymbol{g}, \tag{1b}$$

where $\boldsymbol{u} := (u, v)$ is the velocity, $p$ is the pressure, $\boldsymbol{g} := (g_x, g_y)$ is the gravitational acceleration,

$$\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} := \frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{u}$$

is the material derivative of the velocity, and $\mathbf{D}$ is the strain rate tensor,

$$\mathbf{D} := (\boldsymbol{\nabla} \boldsymbol{u}) + (\boldsymbol{\nabla} \boldsymbol{u})^{\mathrm{T}} := \begin{pmatrix} 2\partial_x u & \partial_y u + \partial_x v \\ \partial_y u + \partial_x v & 2\partial_y v \end{pmatrix}. \tag{2}$$

The Navier-Stokes equations must be supplemented with proper boundary conditions, which are particularly important for interfacial dynamics. The most common and easiest condition is a Dirichlet condition on the velocity, defined on some boundaries $\Gamma_u^{\mathrm{D}}$ and $\Gamma_v^{\mathrm{D}}$ for $u$ and $v$ respectively. We note that $\Gamma_u^{\mathrm{D}}$ and $\Gamma_v^{\mathrm{D}}$ may be different. For instance, if the $x$-symmetry condition is imposed about the $y$-axis, then the $y$-axis belongs to $\Gamma_u^{\mathrm{D}}$ where $u = 0$, but it does not belong to $\Gamma_v^{\mathrm{D}}$. The most relevant boundary condition for multiphase flows is a natural condition defined on the interfaces $\Gamma^{\mathrm{N}}$. This condition expresses the force balance between the surface tension and the stress at each point $\boldsymbol{x} := (x, y)$ on the interface:

$$[(-p\mathbf{I} + \mu\mathbf{D}) \cdot \boldsymbol{n}]_-^+ = \sigma\kappa\boldsymbol{n}, \quad \boldsymbol{x} \in \Gamma^{\mathrm{N}}, \tag{3}$$

where $[\cdot]_-^+$ is the difference between the physical quantity across the interface at a point $\boldsymbol{x}$, $\kappa$ is the curvature of the interface, and $\boldsymbol{n}$ is the unit normal vector. We note that $\Gamma^{\mathrm{N}}$ may be a union of disjoint sets $\Gamma_1^{\mathrm{N}}$, $\Gamma_2^{\mathrm{N}}$, ..., $\Gamma_{\mathrm{in}}^{\mathrm{N}}$. Free-surface flow is included naturally in this framework; when a phase is gas, we consider it to be at constant pressure $p_0$, and its density and viscosity zero.

Consider an interface between the two immiscible fluids passively driven by the fluid flow. The following kinematic boundary condition must be satisfied:

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{u}(\boldsymbol{x}) \ , \tag{4}$$

where $\boldsymbol{u}$ is the flow velocity.

In this work, we will restrict ourselves to the system in two dimensions (2D). For the sake of convenience which will become obvious later in Sec. 3.3, the scalar form of the Navier-Stokes equations (1) is also given below:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{5a}$$

$$\rho\frac{\mathrm{d}u}{\mathrm{d}t} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x}\left(2\mu\frac{\partial u}{\partial x}\right) + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] + \rho g_x, \tag{5b}$$

$$\rho\frac{\mathrm{d}v}{\mathrm{d}t} = -\frac{\partial p}{\partial y} + \frac{\partial}{\partial y}\left(2\mu\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] + \rho g_y. \tag{5c}$$

A simple but nontrivial solution for two immiscible fluids was found by Laplace: for a circular (in 2D) or a spherical (in 3D) droplet (bubble) of radius $R$, as illustrated in Fig. 2, the velocity field is quiescent in the flow domain, *i.e.*, $\boldsymbol{u} = \mathbf{0}$, the pressure is constant inside ($p_1$) and outside ($p_2$) with $p_1 - p_2 = \sigma\kappa = \sigma/R$ in 2D, or $p_1 - p_2 = \sigma\kappa = 2\sigma/R$ in 3D. *The Laplace law* is particularly useful for the study of the *spurious* current induced by numerical artifacts [5, 19–22].

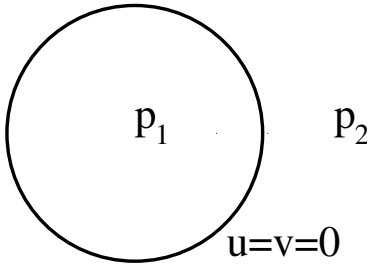


Figure 2: Laplace law for a static droplet (bubble) of radius $R$. The pressures inside and outside of the droplet (bubble) are $p_1$ and $p_2$, respectively.

## 3. Numerical method

In this section, we will present the numerical method to solve the Navier-Stokes equations (1) following our previous work [18, 28], including the implementation details which have not been published previously. Our proposed numerical methodology consists of four parts. The first part, in Sec. 3.1, is the adaptive mesh technique, which is an essential component of our solution methodology. The second, in Sec. 3.2, is the arbitrary Lagrangian-Euler method, which is the basis of our proposed moving mesh method. The third, in Sec. 3.3, is the finite-element discretization used in the present work. Finally, the fourth, in Sec. 3.4, is the treatment of interfacial tension force, which is another critical component of our methodology.

### *3.1. Adaptive Mesh Generation*

We treat the interfaces between two immiscible fluids as *discontinuities* and *track* them explicitly. In particular, the interfaces are coincident with mesh lines. Thus, our method belongs to the category of interface-conforming

methods, in which the fluid-fluid interface has no thickness. Our method requires the mesh to follow the evolution of the interface dynamically. The motion of the mesh vertices on the interfaces is determined by the flow velocity $\boldsymbol{u}$ at these vertices using a Lagrangian method (Eqn. (8) in [18]). The positions of all other vertices in the interior of the computational domain are relocated by Laplacian smoothing, which adjusts the location of a vertex $\boldsymbol{q}$ to the arithmetic mean of its neighboring incident vertices (vertices connected to $\boldsymbol{q}$ through a mesh edge) :

$$\boldsymbol{q} = \frac{1}{N_b} \sum_{j=1}^{n} \boldsymbol{p}_j, \tag{6}$$

where $\boldsymbol{p}_j$ is a neighboring incident vertex of $\boldsymbol{q}$ and $N_b$ is the number of incident neighbors. Fig. 3 illustrates an example of this operation.

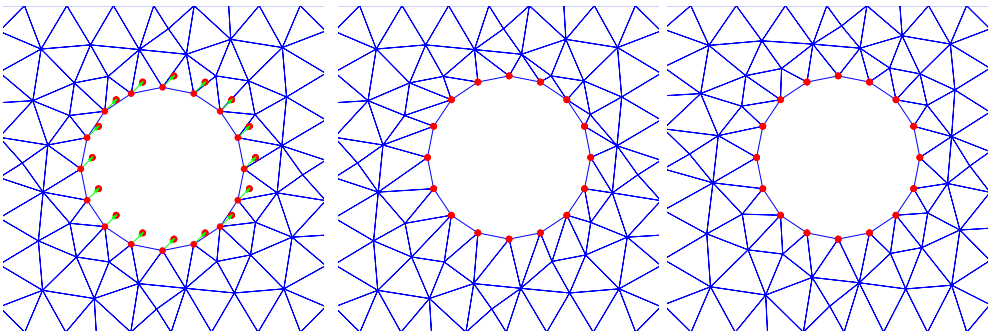


Figure 3: (a) Interface nodes (red circles) moved according to the Lagrangian method. (b) New mesh with poor quality triangles near the interface. (c) Result of Laplacian smoothing operations.

The Laplacian smoothing keeps the mesh topology intact, but it is not sufficient to maintain good mesh quality. Mesh quality is maintained with three additional operations: edge swapping, edge splitting, and edge contraction. The last two operations require the definition of a *local* length $l_s$, which specifies the intended mesh size at each location within the flow domain. $l_s$ depends on a constant $\alpha$ which relates $l_s$ of an interface vertex to the local curvature $\kappa$ and a constant $G$ which limits the difference of $l_s$ between neighboring mesh vertices. We refer the reader to [18] for the details.

*3.2. Arbitrary Lagrangian-Eulerian Method*

According to the kinematic boundary condition, Eq. (4), the interfaces are defined by the sets of material points moving with the flow velocity field $\boldsymbol{u}$, and are most conveniently described from a Lagrangian viewpoint, rather than the Eulerian one of a fixed observer. On the other hand, it is well known that purely Lagrangian methods are not well suited for solid boundaries, because the mesh is sheared by the flow and thus needs frequent re-meshing to maintain the mesh quality. Also, inflow and outflow boundaries are not naturally dealt with by purely Lagrangian methods. We thus employ an Arbitrary Lagrangian-Eulerian (ALE) method which couples the advantages and avoids the drawbacks of both methods.

In the ALE framework, the mesh is advected with an advection velocity $\boldsymbol{a}$, which may locally differ from the material velocity $\boldsymbol{u}$. When $\boldsymbol{a} = 0$, the method is reduced to the Eulerian method, ideal for dealing with fixed boundaries; when $\boldsymbol{a} = \boldsymbol{u}$, the method is reduced to the Lagrangian method, specially tailored for moving boundaries. The bulk region of the mesh provides a smooth transition between the two methods. At each time step, the node $\boldsymbol{x}^n$ of the old triangulation $\{T_h^n\}$ is moved to $\boldsymbol{x}^{n+1} = \boldsymbol{x}^n + \boldsymbol{a}\Delta t$ to obtain the new triangulation $\{T_h^{n+1}\}$, as illustrated in Fig. 4, where only the triangles around node $\boldsymbol{x}^n$ of $T_h^n$ are illustrated.

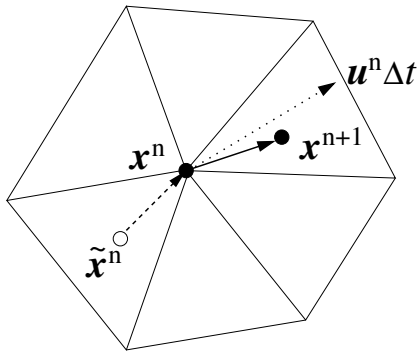


Figure 4: Moving mesh and method of characteristics: the new mesh node $\mathbf{x}^{n+1}$ is connected to the old node $\boldsymbol{x}^n$ via the vector $\boldsymbol{a}\Delta t$ (represented by the arrow with a solid-line tail), while the characteristic foot $\tilde{\mathbf{x}}$ connected to $\boldsymbol{x}^n$ via the vector $(\boldsymbol{u}-\boldsymbol{a})\Delta t$ (represented by the arrow with a broken-line tail). The position of the material point at time $t^{n+1}$ is depicted by the arrow with a dotted-line tail $\boldsymbol{u}\Delta t$.

The nonlinear convection terms in the Navier-Stokes equations are difficult to deal with numerically, and they are most conveniently discretized *via*

a *method of characteristics* in the context of the ALE framework [30]. The material derivative of a physical quantity $f$ is

$$\frac{\mathrm{d}f}{\mathrm{d}t} := \frac{\partial f}{\partial t} + (\boldsymbol{u} - \boldsymbol{a}) \cdot \boldsymbol{\nabla} f. \tag{7}$$

The temporal discretization of the above equation is realized via a semi-implicit method:

$$\frac{\mathrm{d}f}{\mathrm{d}t}(\boldsymbol{x}^{n+1}) \approx \frac{f^{n+1}(\boldsymbol{x}^{n+1}) - f^n(\tilde{\boldsymbol{x}}^n)}{\Delta t} \tag{8}$$

where $\tilde{\boldsymbol{x}}^n$ at time level $n$ is the characteristic foot of $\boldsymbol{x}^{n+1}$ (cf. Fig. 4):

$$\tilde{\boldsymbol{x}}^n := \boldsymbol{x}^{n+1} - \boldsymbol{u}^n(\boldsymbol{x}^n)\Delta t = \boldsymbol{x}^n - (\boldsymbol{u}(\boldsymbol{x}^n) - \boldsymbol{a}(\boldsymbol{x}^n))\,\Delta t$$

In the moving mesh method, $f^n$ and $f^{n+1}$ are defined on triangulations $\{T_h^n\}$ and $\{T_h^{n+1}\}$, respectively. For the sake of accuracy and simplicity, we restrict the time step $\Delta t$ to be so small such that both $\tilde{\boldsymbol{x}}^n$ and $\boldsymbol{x}^{n+1}$ fall within one of the neighboring triangles of the node $\boldsymbol{x}^n$. Once the triangle in which $\tilde{\boldsymbol{x}}^n$ lies is identified, the quantity $f^n(\tilde{\boldsymbol{x}})$ is interpolated from the nodal values of $f^n$ in this triangular element. It is easily done for an affine Lagrangian element whose edges are straight lines. First the barycentric coordinates $\lambda_1$, $\lambda_2$ and $\lambda_3$ of $\tilde{\boldsymbol{x}} := (\tilde{x}, \tilde{y})$ are calculated:

$$\begin{aligned} \lambda_1 + \lambda_2 + \lambda_3 &= 1 \\ \lambda_1 x_1 + \lambda_2 x_2 + \lambda_3 x_3 &= \tilde{x} \\ \lambda_1 y_1 + \lambda_2 y_2 + \lambda_3 y_3 &= \tilde{y} \end{aligned}$$

where $x_i$ and $y_i$, $i = 1, 2, 3$, are the coordinates of the vertices of the triangle. Then $f^n(\tilde{\boldsymbol{x}})$ is interpolated according to its approximate finite element space, which is either $P_1$ or $P_2$ in this work.

### *3.3. Finite element discretization of the Navier-Stokes equations*

To apply the finite-element methodology, the governing equation is first converted to a weak formulation (Eqs. (12) – (14) in [18]). These equations are discretized on a finite-element triangulation $\{T_h\}$, where interfaces between different phases are lines of the mesh system, dividing the computational domain into different regions, each of which contains only one

phase. We use a semi-implicit method for the temporal discretization: a forward Euler integration for the Lagrangian movement of nodes, Navier-Stokes equation. The method is hence of first order in time. We made this choice for the simplicity of the method; a second order method in time would be desirable but much more complicated as it will involve calculations on two different meshes at each time step. We are mainly interested in unsteady flows encompassing disparate length-scales across several orders of magnitude, hence requiring small time-steps. The first order accuracy in time does not lead to inaccuracy of our results.

In this work, Taylor-Hood elements are used to discretize the velocity and pressure. The velocity is approximated by a quadratic function in each triangular element, and is assumed to be continuous over the entire domain. Our method differs from the conventional $P_2$-$P_1$ FEM only on that we allow the pressure to take double values on the interface nodes. This is necessary because the pressure is discontinuous due to the surface tension forces. We call this pressure space $P_1d$ in order to differentiate it from the conventional continuous space $P_1$ (cf., *e.g.*, [18]).

In our method, the mesh is composed of linear affine elements (triangles with straight edges). It would be more accurate to employ isoparametric elements (triangles with curved edges) near the boundaries and interfaces. But the elements with curved edges would lead to a much more complicated implementation, in particular when managing the moving mesh. Consistent with our use of affine elements, the interface is explicitly represented by a series of marker points (mesh vertices) $\boldsymbol{x}_i$, $i = 1, 2, \ldots, N$, connected by straight segments, as illustrated in Fig. 5. The curvature $\kappa$ at interface vertex $\boldsymbol{x}_i$ is calculated by fitting a circle through three adjacent nodes $\boldsymbol{x}_{i-1}$, $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i+1}$ [29]. Although our representation of the interface is piecewise linear, numerical tests show second-order convergence. In particular, when the interface is circular, our method does reproduce the exact results up to machine accuracy.

We use a *Uzawa* iterative algorithm for the resolution of the Navier-Stokes equations. We are mainly interested in axisymmetric flows. The 2D method is converted to a method for axisymmetric flows following [31], and the discrete equations are solved in a similar manner as in 2D (cf., *e.g.*, [18] for the details).

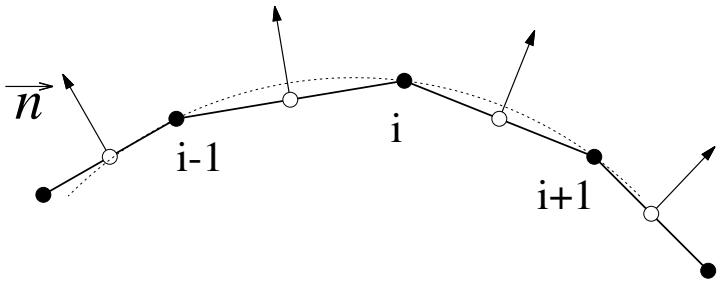


Figure 5: The interface is explicitly represented by a series of marker points (mesh vertices) connected by straight segments (mesh edges). The dashed line is a local circle fitted onto nodes $i$, $i-1$ and $i+1$.

*3.4. Existence of static equilibrium solutions*

Now we present a *key* theoretical result of this work. In general, most numerical methods for interfacial flows have acceptable accuracy for the motion of the interface. However, accurate representation of boundary conditions remains a problem. An inaccurate representation can result in an unbalanced surface tension force which generates small numerical artifacts near the interface — the so-called spurious currents [6, 9]. This problem becomes acute when dealing with common fluids like water, for which the surface tension force is often much stronger than the viscous dissipation. This is best characterized by the dimensionless Ohnesorge number $\mathrm{Oh} := \mu/\sqrt{\rho\sigma D}$, defined as the ratio of the viscous force and surface tension. Note that Laplace number $\mathrm{La} := 1/\mathrm{Oh}^2$ is sometime used instead of Ohnesorge number Oh (cf., *e.g.*, [33]). For a water droplet of diameter $D = 1\,\mathrm{cm}$ in air, $\mathrm{Oh} \approx 10^{-3}$. Theoretically, the stationary *Laplace* solution for a circular (spherical) bubble or droplet does exist (cf. Fig. 2). However, there is no guarantee of a similar solution for numerical methods in general. An imbalance between the surface tension and pressure jump across the interface not only degrades the accuracy of the methods, but more importantly, threatens the continuation of the computations as they frequently explode prematurely. Two influential works on fixed mesh methods have been proposed in the past [19, 21]. In our method, the numerical counterpart of the (static) *Laplace* solutions can be constructed effortlessly for a circular (spherical) droplet or bubble.

Consider a circular droplet of fluid 1 with radius $R$ initially at rest in the ambient fluid 2 (cf. Fig. 2). The equilibrium (static) solution is the *Laplace* solution: the velocity $u = v = 0$, and the pressure $p$ is constant both inside ($p_1$) and outside ($p_2$) with $p_1 - p_2 = \sigma/R$. A portion of the discrete configuration is illustrated in Fig. 6, in which the circular interface is marked with a thick solid line, $\Gamma^{\mathrm{N}}$. From Eqs. (13) and (14) in [18], we have to prove that

$$-\int_{\Omega} \frac{\partial \phi}{\partial x} p \mathrm{d}\boldsymbol{x} = \int_{\Gamma^{\mathrm{N}}} \sigma \kappa n_x \phi \mathrm{d}s \tag{9}$$

for every basis function $\phi$ associated with a node of a Taylor-Hood element.

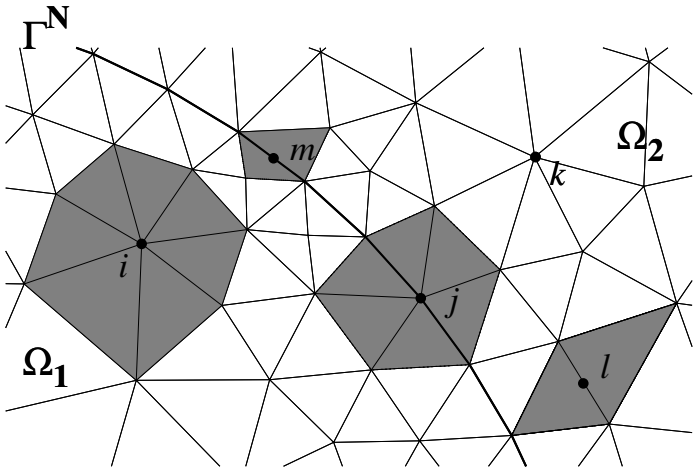


Figure 6: Triangulation near a circular/spherical interface $\Gamma^{\mathrm{N}}$ which separates two phases in $\Omega_1$ and $\Omega_2$. The thick line depicts the interface while the thin lines depict interior mesh edges. Nodes $i$ and $j$ are associated with 2 vertices of the mesh, and nodes $l$ and $m$ are associated with 2 edges of the mesh. The shaded regions $\Omega_i$, $\Omega_j$, $\Omega_l$, and $\Omega_m$ are composed of triangles around nodes $i$, $j$, $l$ and $m$, respectively.

Consider first the case of node $i$ associated with an interior vertex inside fluid 1 shown in Fig. 6. Let $\Omega_i$ denote the shaded region composed of the triangles around this vertex. The associated basis function $\phi_i$ vanishes outside $\Omega_i$ including on boundary $\partial\Omega_i$. Hence the right hand side of Eq. (9) is zero, as $\Omega_i$ does not intersect interface $\Gamma^{\mathrm{N}}$. The left hand side:

$$-\int_{\Omega} \frac{\partial \phi_i}{\partial x} p \, \mathrm{d}\boldsymbol{x} = -p_1 \int_{\Omega_i} \frac{\partial \phi_i}{\partial x} \, \mathrm{d}\boldsymbol{x} = -p_1 \int_{\partial\Omega_i} \phi_i n_x \, \mathrm{d}s = 0, \tag{10}$$

where we have used the fact that within $\Omega_i$ the pressure is a constant $p = p_1$. Consider node $l$ associated with an interior edge in fluid 2 (shown in Fig. 6), and let $\Omega_l$ denote the shaded region composed of the 2 triangles around this edge. Eq. (9) for the basis function $\phi_l$ is readily derived if $i$ is replaced by $l$ and $p_1$ by $p_2$ in the above calculation.

Consider now the interface node $j$ as illustrated in Fig. 6, and let $\Omega_j$ denote the shaded area composed of the triangles around the vertex $j$, and $\Gamma_j$ the portion of the interface in $\Omega_j$. The area $\Omega_j$ is composed of $\Omega_{j,1}$ in fluid 1 and $\Omega_{j,2}$ in fluid 2. The curvature of the interface is calculated by fitting a circle through three vertices; the numerical curvature is a constant, $\kappa = 1/R$ (exact up to the machine accuracy). The pressure $p$ is not a constant in the region $\Omega_j$, but is constant in each of $\Omega_{j,1}$ and $\Omega_{j,2}$. This is made possible by the use of $P_1d$ space in our method, which allows pressure to have double values on the interface. The left hand side of Eq. (9) becomes:

$$-\int_\Omega \frac{\partial \phi_j}{\partial x} p \mathrm{d}\boldsymbol{x} = -p_1 \int_{\Omega_{j,1}} \frac{\partial \phi_j}{\partial x} \mathrm{d}\boldsymbol{x} - p_2 \int_{\Omega_{j,2}} \frac{\partial \phi_j}{\partial x} \mathrm{d}\boldsymbol{x} = -p_1 \int_{\partial\Omega_{j,1}} \phi_j n_x \mathrm{d}s$$
$$- p_2 \int_{\partial\Omega_{j,2}} \phi_j n_x \mathrm{d}s = \int_{\Gamma_j} (p_1 - p_2)\phi_j n_x \mathrm{d}s = \int_{\Gamma_j} \left(\frac{\sigma}{R}\right) \phi_j n_x \mathrm{d}s = \int_{\Gamma^{\mathrm{N}}} \sigma\kappa\phi_j n_x \mathrm{d}s.$$

Hence we have established Eq. (9) in discrete form. We note that $n_x$ in $\Omega_{j,1}$ and $\Omega_{j,2}$ is of opposite sign at the intersection $\Gamma_j$, and $\phi_j = 0$ on $\partial\Omega_j$. Now consider an interface edge node $m$ (cf. Fig. 6), and let $\Omega_m$ denote the shaded region composed of the 2 triangles around this edge. Eq. (9) for the basis function $\phi_m$ is readily derived if $j$ is replaced by $m$ in the above calculation.

Similarily, the $y$-momentum equation can be obtained by replacing the $x$-derivative by the $y$-derivative in the above derivations. Therefore we have completed the proof for the two-dimensional case. The proof for the axisymmetric case is given in the Appendix.

## 4. Numerical results

This section presents the numerical results obtained by using the proposed numerical method. A series of numerical tests with varying degrees of complexity and difficulty are conducted to validate the capability of the numerical scheme. The first test is a static circular (spherical) droplet in both two dimensional and axisymmetrical cases with Oh $\geq 10^{-3}$. This test can definitively demonstrate that our numerical scheme completely eliminates the

*spurious* current up to machine accuracy, as shown in Sec. 3.4. The second test is an oscillating bubble/droplet. The numerical accuracy of the proposed method is validated by the theory of Prosperetti [26]. The third test is the formation of a pendant droplet from a dripping faucet involving no-slip walls which were not considered in the previous two tests. The fourth and final test is the water dripping phenomenon, with real physical parameters, which is a stringent test to the numerical methods.

*4.1. Stationary droplet and spurious currents*

It has been shown in Sec. 3.4 that our method admits discrete static Laplace solutions exactly. We investigate the accuracy and stability of these solutions and the convergence of simulations with slightly perturbed solutions. In what follows, a circular/spherical droplet of diameter $D = 1.0$ is considered. The density of the droplet is chosen as $\rho_1 = 1.0$ and the interfacial tension coefficient $\sigma = 1.0$. For simplicity both the viscosity and density ratios of the inner and outer fluid are set to be 1.0. We vary the viscosity $\mu$ to obtain the desired Oh number. Our numerical results are presented in terms of dimensionless time and maximum spurious velocity. There are two natural independent timescales, the capillary time scale $T_\sigma$ and the viscous time scale $T_\mu$:

$$T_\sigma := \sqrt{\frac{\rho D^3}{\sigma}}, \quad T_\mu := \frac{\rho D^2}{\mu} \ . \tag{11}$$

Time is rescaled by $T_\mu$ for the results in this section. To enhance the significance of this study, our numerical tests are carried out for a time larger than $T_\mu$.

The amplitude of the spurious current is characterized by the maximum velocity $U_{\max}$ in the computational domain (which should be zero when the flow is quiescent.) The accuracy of the method is measured using the dimensionless velocity, *i.e.*, the *Capillary* number:

$$\mathrm{Ca} = \frac{\mu U_{\max}}{\sigma}. \tag{12}$$

The initial and boundary conditions are set up as follows. The flow domain is a rectangle of size $[0, 30] \times [-30, 30]$ in the unit of a circle with diameter $D = 1.0$ centered at $(0, 0)$, such that only the right half of the droplet is considered in the computations. The symmetry condition is imposed at the left boundary of the computational domain. An initial mesh is

generated with a circular interface. The mesh parameter $G$ is fixed at 0.1. We vary $\alpha$ to control the number of the interface nodes and hence the mesh resolution. The velocity is set to be zero everywhere initially, *i.e.*, $\boldsymbol{u} = \mathbf{0}$. The pressure does not require initialization, as it is determined from the velocity for incompressible flows. In our simulations, we first seek large $\alpha$ (coarse meshes) such that the computation remains stable and convergent. We then seek large time-steps to do so. We use 10 iterations of the *Uzawa* method to ensure that the incompressibility condition is satisfied with machine accuracy.

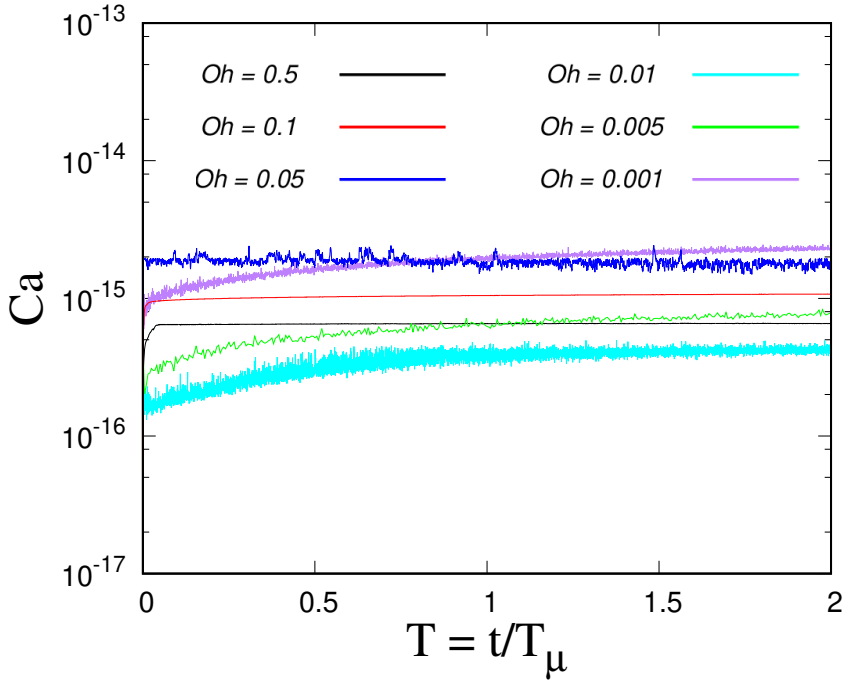


Figure 7: Simulations of a half circular droplet. The evolution of Capillary number Ca as a function of the dimensionless time, $t/T_\mu$, with various values of Oh. The numbers in the legend indicate the Oh number. The computation is performed with minimum mesh size and maximum time-step that permit a stable solution.

We first present the two-dimensional results. Fig. 7 shows the evolution of Ca($t$) for 6 different values of Oh $\geq 10^{-3}$. The Ca for all Oh starts with the order of magnitudes of $10^{-15}$ or less. This is no surprise since the initial conditions are in fact the static solution in our numerical method. The value of Ca is shown for a dimensionless time $T := t/T_\mu = 2.0$ in Fig. 7. Ca shows a slight increase, but remains on the order of $10^{-15}$ in all cases. This shows that our static solution is stable and the *spurious* current is due to the machine error for Oh $\geq 10^{-3}$. The maximum Ca at $T = 2.0$, the values of Oh and other two relevant parameters are tabulated in Table 1. From the physical point of view, the smaller Oh is, the weaker the viscous dissipation, resulting in more capillary waves of higher frequency (smaller wave length). This is reflected by the mesh resolutions required to maintain the stability and convergence of the simulations. Simulations with smaller Oh require a finer mesh (hence smaller time steps) to capture all of the relevant physics of smaller scales, otherwise the computation will be unstable.

Table 1: Maximum Ca at $T = 2.0$ for simulations of a half circular droplet.

| Oh | $\alpha$ | No. of interface nodes | $\Delta t$ | Ca |
|---|---|---|---|---|
| 0.5 | 0.2 | 8 | $2.0 \times 10^{-2}$ | $6.7 \times 10^{-16}$ |
| 0.1 | 0.1 | 16 | $1.0 \times 10^{-2}$ | $1.1 \times 10^{-15}$ |
| $5 \times 10^{-2}$ | 0.05 | 32 | $5.0 \times 10^{-3}$ | $1.9 \times 10^{-15}$ |
| $10^{-2}$ | 0.05 | 32 | $2.0 \times 10^{-3}$ | $5.2 \times 10^{-16}$ |
| $5 \times 10^{-3}$ | 0.025 | 64 | $1.0 \times 10^{-3}$ | $7.5 \times 10^{-16}$ |
| $10^{-3}$ | 0.00625 | 256 | $2.5 \times 10^{-4}$ | $2.3 \times 10^{-15}$ |

The importance of obtaining the long-time equilibrium solution has been emphasized previously [5]. In our simulations, the initial condition is an exact solution with machine accuracy and is stable indefinitely in time. To further test the stability and robustness of the equilibrium solution, we introduce a sinusoidal disturbance to the initial circular interface:

$$r = \frac{1}{2} + \frac{\epsilon}{N} \sum_{n=1}^{N} \sin(n\theta) \ , \tag{13}$$

where $N$ is the number of sinusoidal modes, $\epsilon$ the amplitude of the disturbance and $\theta$ the polar angle of the interface point in the polar coordinate system. In this numerical test, we used $N = 100$ and $\epsilon = 10^{-6}$. Fig. 8 shows

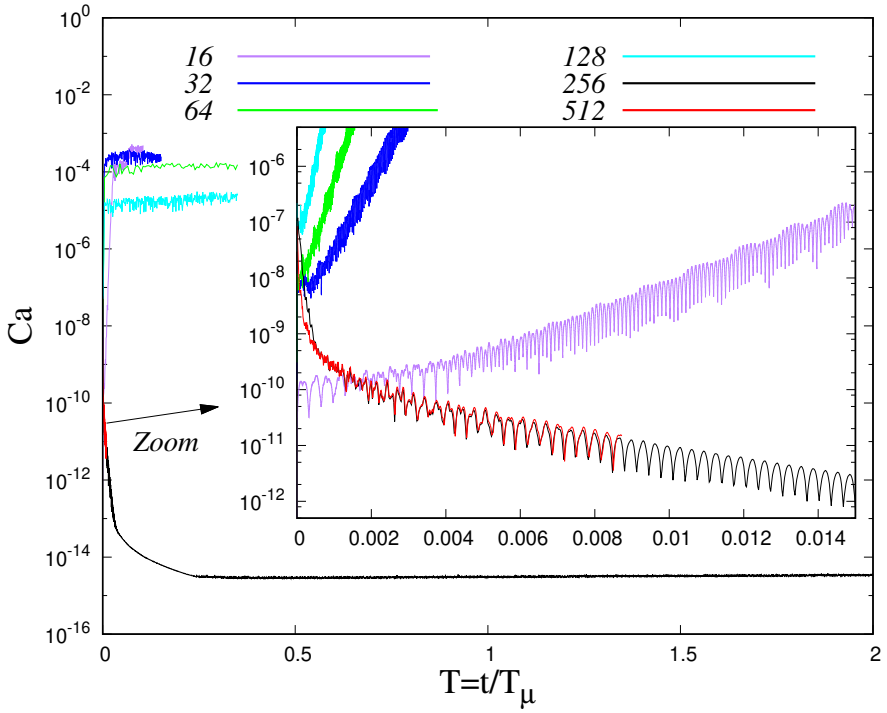


Figure 8: The evolution of Capillary number Ca for a half circular droplet with Oh = $10^{-3}$ on different meshes. The initial interface is prescribed by Eq. (13). The number of nodes on the interface is indicated by the numbers in the legend.

the evolution of Ca($t$) for the disturbed initial shape and Oh = $10^{-3}$. In the simulations performed on coarse meshes with 16, 32, 64, and 128 interfacial nodes, the solution deviates from the static solution: Ca($t$) grows exponentially (note that Fig. 8 is in semi-log scale.) in the initial stage and below $10^{-3}$. It can be seen that the coarser the mesh, the slower the Ca grows in the initial stage; the reason may be due to the lack of unstable modes of higher frequency on coarser meshes. Note that the simulations on the two finer (larger) meshes do converge to the theoretical static solution. As shown in the insert of Fig. 8, the evolution of Ca($t$) in simulations on the two finer meshes (*i.e.*, the number of interfacial nodes of 256 and 512) are virtually identical to each other.

We have also investigated the stability of numerical simulations of a quarter droplet. The computational domain is [0, 30] × [0, 30], where the symmetry condition is applied on the left and bottom boundaries. Fig. 9 shows

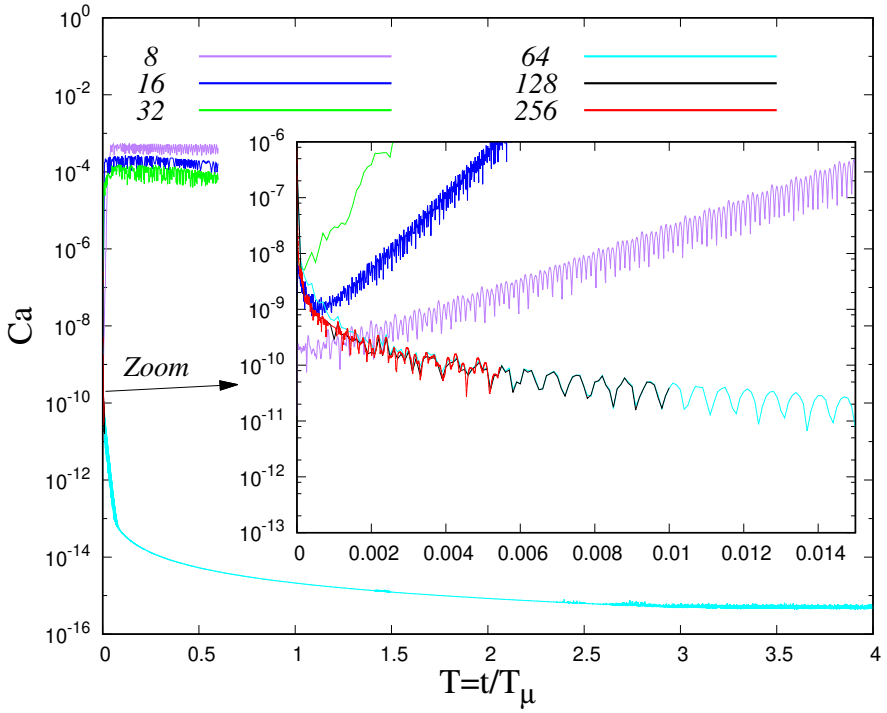


Figure 9: The evolution of Capillary number Ca for a quarter circular droplet with Oh = $10^{-3}$ on different meshes. The initial interface is prescribed by Eq. (13). The number of nodes on the interface is indicated by the numbers in the legend.

the evolution of the maximum Ca($t$) with Oh = $10^{-3}$. Similar to the case of the half droplet, in the simulations performed on the coarse meshes (with 8, 16, and 32 interfacial nodes), Ca grows exponentially in the initial stage and saturates below $10^{-3}$, while Ca obtained with the three finer meshes (with 64, 128 and 256 interfacial nodes) agree very well with each other as shown in the insert of Fig. 9. This figure also shows the evolution of Ca for $T = 4.0$ with 64 interfacial nodes and Ca converges to machine zero. However, the half-droplet simulations are not equivalent to the quarter-droplet simulations. We note two distinctions between them: (1) The half-droplet simulation with 128 interfacial nodes has the same interface resolution as the quarter-drop simulation with 64 interfacial nodes, but the half-droplet case does not decay to the static solution, while the quarter-drop case does; (2) Convergence in the half droplet case (Ca $\sim 10^{-15}$ at $T = 0.3$) is faster than in the quarter droplet case (Ca $\sim 10^{-15}$ at $T = 3.0$). The results indi-

cate that the additional symmetry constraint in the quarter-droplet make the system more rigid and less dissipative. Our method does not use numerical viscosity to damp oscillations. Relevant physical length scales must be duly resolved for a simulation to be stable. It is no surprise that below a certain resolution (256 for half droplet and 64 for a quarter drop) our simulations do not converge to zero.

Table 2: Maximum Ca at $T = 2.0$ for simulations of a spherical droplet.

| Oh | $\alpha$ | No. of interface nodes | $\Delta t$ | Ca |
|---|---|---|---|---|
| 0.5 | 0.2 | 8 | $2.0 \times 10^{-2}$ | $1.1 \times 10^{-15}$ |
| 0.1 | 0.1 | 16 | $1.0 \times 10^{-2}$ | $1.5 \times 10^{-15}$ |
| $5 \times 10^{-2}$ | 0.05 | 32 | $5.0 \times 10^{-3}$ | $1.9 \times 10^{-15}$ |
| $10^{-2}$ | 0.05 | 32 | $2.0 \times 10^{-3}$ | $1.4 \times 10^{-15}$ |
| $5 \times 10^{-3}$ | 0.025 | 64 | $1.0 \times 10^{-3}$ | $1.5 \times 10^{-15}$ |
| $10^{-3}$ | 0.00625 | 256 | $2.5 \times 10^{-4}$ | $4.2 \times 10^{-15}$ |

Axisymmetrical flows are more realistic than two-dimensional ones, thus more physically relevant. This subsection ends with the simulation of a spherical droplet. The initial condition is exactly the same as in the case of a two-dimensional circular droplet. The initial pressure is not specified and the axisymmetrical flow solver yields a pressure jump twice of that of the circular droplet: $p_{\text{outer}} - p_{\text{inner}} = 2\sigma/R$.

The result is similar to that of the 2D circular droplet. Similar to Table 1, the maximum Ca at the dimensionless time $T = 2.0$, the Oh number and two relevant parameters are tabulated in Table 2. The values of Oh are the same as those in Table 1 for a 2D circular droplet. The optimal numerical parameters such as the minimal mesh sizes and the maximal time-steps are also the same. Fig. 10 shows the evolution of Ca as a function of $T$ for several values of Oh. The capillary number Ca is on the order of $10^{-15}$ in the duration $T = 2.0$.

There are three time scales based on mesh sizes: the convective time $t_c$, the viscous time $t_\mu$ and the capillary time

$$t_\sigma = \sqrt{\frac{(\rho_1 + \rho_2)h^3}{4\pi\sigma}} \tag{14}$$

where $h$ is the mesh size at the interface [32]. Our method is a semi-implicit one with the viscous terms solved implicitly so $t_\mu$ does not have much influ-

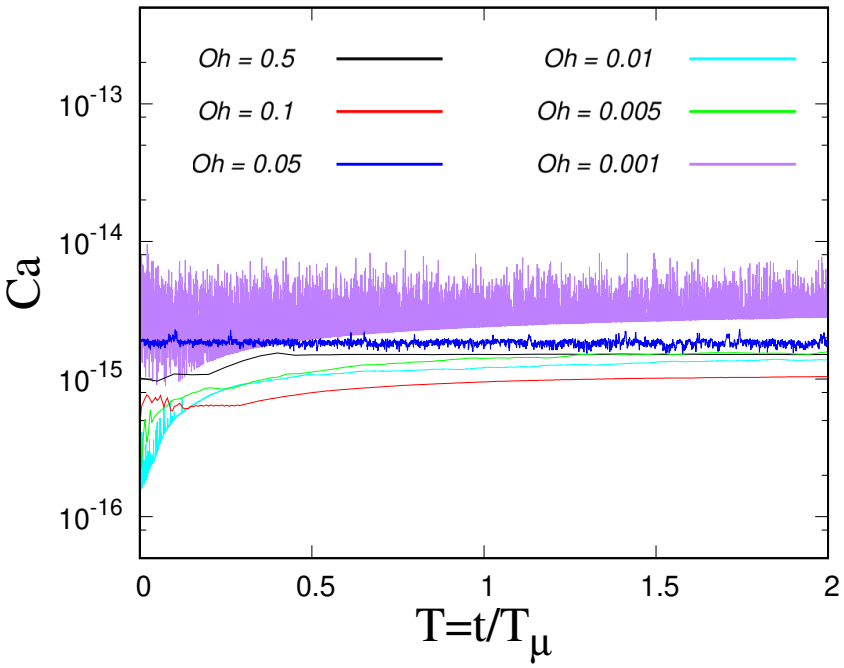


Figure 10: Temporal evolution of Capillary number $Ca$ for a spherical droplet. The numbers in the legend indicate the $Oh$ number. The computation is performed on minimal mesh sizes and maximal time-steps that permit a stable solution.

ence on the time-step. In general, the time-step must be smaller than $t_c$ (a CFL-type condition) and $t_\sigma$ which decreases with $h$ as $h^{3/2}$. Consider the case of the Laplace solution in section 4.1: $\mathbf{u} = 0$, the time-step is dictated by $t_\sigma$. Fig. 11 shows the data (represented in circles) calculated from Table 1. It is clear that the data follows approximately Eq. (14) represented by the solid line.

### *4.2. An oscillating bubble/droplet*

In the previous subsection, only "infinitesimal" perturbation was introduced to the initial circular interface. The simulations were essentially around

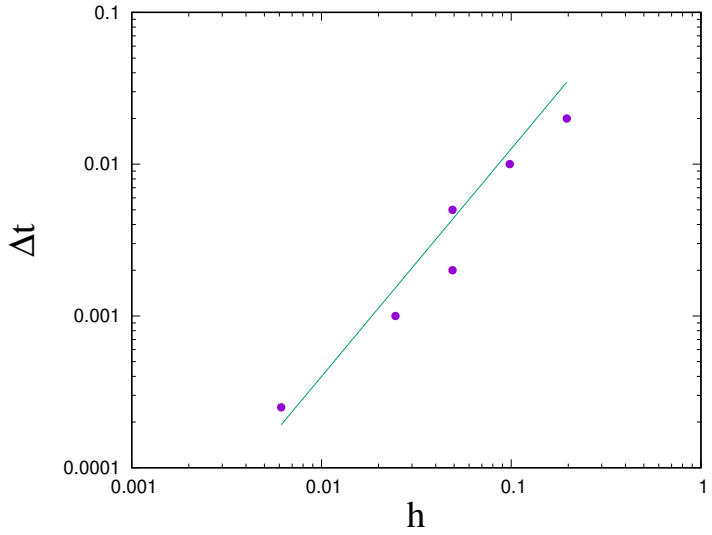


Figure 11: The relation between the time step $\Delta t$ and the interface mesh size $h$ in the numerical experiment of the Laplace solution. The circles are calculated from data in Table 1 and the solid line represents Eq. (14).

the static solution. In this section, the amplitude of perturbation is increased to a larger size. We consider a drop/bubble initially at rest. The interface is perturbed as:

$$R(\theta) = R_0 + \epsilon_n P_n(\cos\theta),\ 0 \le \theta \le \pi, \tag{15}$$

in which $R_0$ is the unperturbed radius and $P_n$ is the Legendre polynomial of order $n$.

This initial-value problem has an analytic solution [26], which is expressed in *Laplace* transform of the oscillation amplitude $a_n$ of mode $n$ as:

$$\tilde{a}_n(p) = \frac{1}{p}\left(a_n(0) + \frac{p\dot{a}_n(0) - \omega_n^2 a_n(0)}{p^2 + 2b_n p + \omega_n^2 + 2\beta_n b_n p\tilde{Q}_n(p)}\right) \tag{16}$$

in which $\dot{a}_n$ is the time derivative of $a_n(t)$, $\omega_n$ an angular frequency, and $b_0$ the decay rate. The expressions of $\omega_n$, $b_n$, $\beta_n$ and $\tilde{Q}_n(p)$ are different for an oscillating droplet or bubble, and they are given in Table 3, where $\nu$ is the kinematic viscosity, and $I_n(q)$ and $K_n(q)$ are modified Bessel functions of the first and second kind, respectively. It is noted that the comparison with Prosperetti's theory for the oscillation of a bubble was first presented in [20].

We first simulate the oscillation of a free-surface bubble. In this test, the physical parameters are $R_0 = 1$, $\rho = 1.0$, $\sigma = 1.0$ and $\nu = 0.01414$ (Oh =

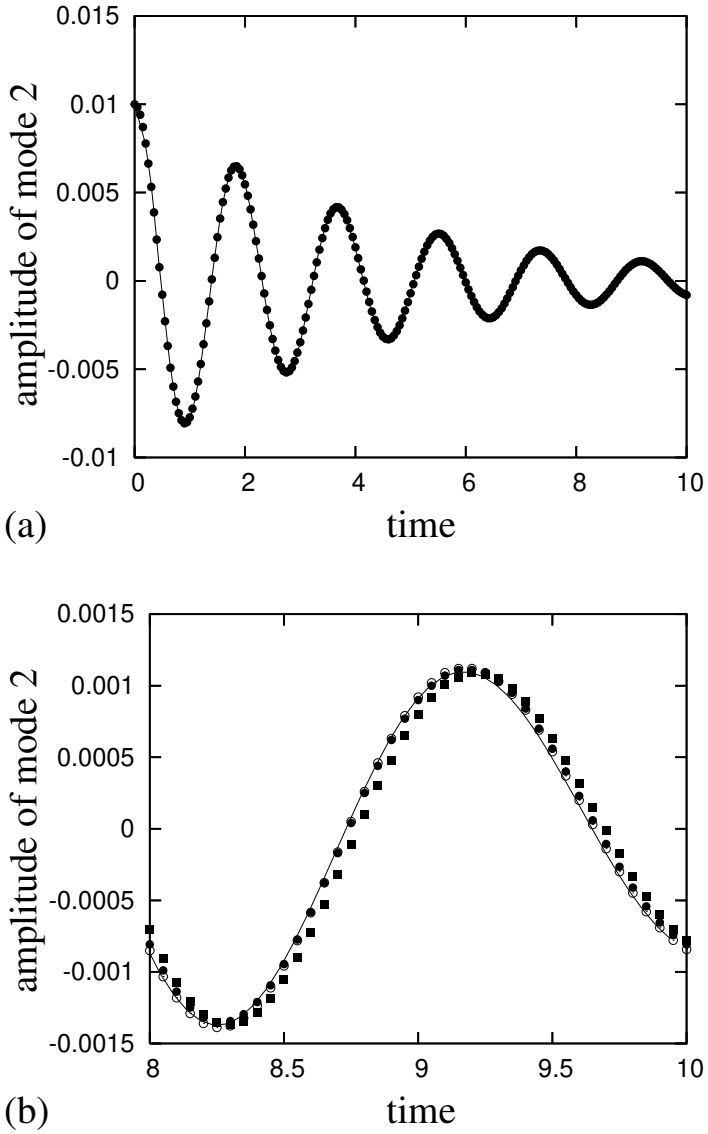


Figure 12: Evolution of the second mode of deformation of a bubble set in a liquid initially at rest. (a) The simulation using a mesh with 854 vertices (64 interface nodes, black circle) and the theoretical result of Eq. (16) represented by the solid line. (b) A zoom on the second mode alongside with the results of a unstructured mesh of 592 vertices (32 interface nodes, the black squares) and structured mesh of size $128 \times 128$.

0.01). Only perturbation of the second mode is considered and the initial

Table 3: The initial value solution of an oscillating drop/bubble

| | Droplet | Bubble |
|---|---|---|
| $\omega_n^2$ | $n(n-1)(n+2)\sigma/\rho R_0^3$ | $(n+1)(n-1)(n+2)\sigma/\rho R_0^3$ |
| $b_n$ | $(n-1)(2n+1)\nu/R_0^2$ | $(n+2)(2n+1)\nu/R_0^2$ |
| $\beta_n$ | $(n-1)(n+1)/(2n+1)$ | $n(n+2)/(2n+1)$ |
| $\tilde{Q}_n(p)$ | $[1-\frac{1}{2}\Gamma_n(R_0\sqrt{p/\nu})]^{-1}$ | $-[1+\frac{1}{2}\Gamma_n(R_0\sqrt{p/\nu})]^{-1}$ |
| $\Gamma_n(q)$ | $qI_{n+1/2}(q)/I_{n+3/2}(q)$ | $qK_{n+1/2}(q)/K_{n-1/2}(q)$ |

perturbation is set at $\epsilon_2 = 0.01$. Fig. 12(a) compares the numerical result of the amplitude of mode 2 with the analytic solution (16). The black circles represent the numerical result on a mesh of 854 vertices (control parameters $\alpha = 0.025$ and $G = 0.1$), while the solid line represents the theoretical prediction. The simulation agrees with the theory very well quantitatively. Fig. 12(b) shows a close up at the sixth crest with two additional numerical results. The black squares represent the numerical result using a mesh of 592 vertices ($\alpha = 0.05$ and $G = 0.1$). The number of interface nodes (32 nodes) in this coarser mesh is half of that (64 nodes) in the previous mesh. We observe a clear convergence towards the analytic solution as the mesh is refined. Finally, the empty circles in this figure are the results obtained on a structured mesh of size $128 \times 128$ (16384 vertices) [17]. The quality is similar to that of the unstructured mesh with 854 vertices. We note that the relative difference at this crest is about 3% between our simulation and Prosperetti's theory. The agreement on the evolution of modes 3 and 4 between our simulations and Prosperetti's theory is equally well (but not shown here).

In Prosperitti's solution to this problem, the oscillating frequency is not a constant, which thus differs from the normal-mode solution on the periodicity. To verify the convergence rate, we compare the length of the first 5 oscillating period. We have carried out simulations of an oscillating bubble with Oh = 0.01 ($\nu = 0.01414$, $R = 1.0$, $\sigma = 1.0$ and $\rho = 1.0$) and an initial perturbation of the second mode: $\epsilon_2 = 0.01$. With the mesh gradient parameter $G$ fixed at 0.15, the mesh size scale $\alpha$ is varied from 0.1 to 0.0125 for meshes of different resolutions. The mesh parameter and the corresponding number of interface nodes are documented in Table 4. The length of the first five periods of the analytic solution is 9.170. Table 4 shows the numerical results and the relative error. Figure 13 shows the relative error depending on the number of

interface nodes $N$, which can be approximated by $4.5/N^2$. The second-order convergence is demonstrated.

Table 4: Mesh parameters and the corresponding numerical values of the length of the 5-th period

| Interface nodes No. | mesh parameter $\alpha$ | the first 5 period | relative error |
|---|---|---|---|
| 16 | 0.1 | 9.350 | 1.963% |
| 32 | 0.05 | 9.212 | 0.458% |
| 64 | 0.025 | 9.180 | 0.109% |
| 128 | 0.0125 | 9.172 | 0.022% |

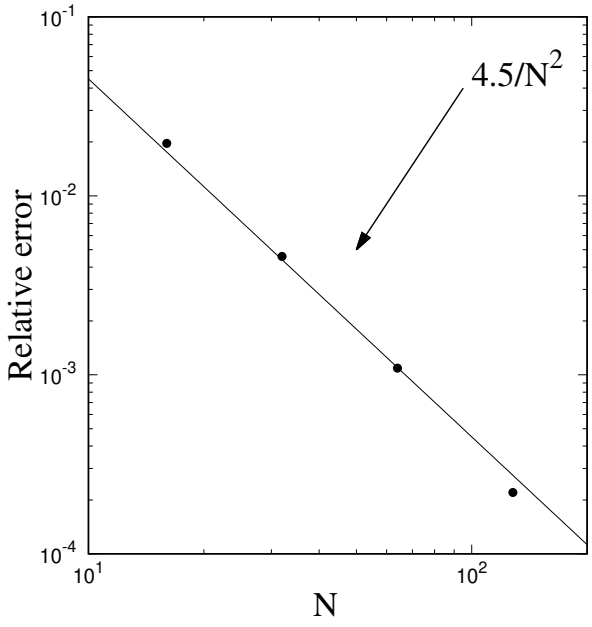


Figure 13: The relative error of the first 5 period is plotted against the number of interface nodes $N$ in dots.

The volumes at the start and the end of the simulation in the above simulation are documented in Table 5. Note that the initial volume differs due to the different mesh resolution. The relative volume changes are all of the order of $10^{-6}$ in these simulations. We have also performed simulations

for $G = 0.1$, 0.2, 0.3, 0.4 and 0.5 with $\alpha = 0.0125$, The relative volume variations remain the order of $10^{-6}$, though the larger the value of $G$ (the coarser the mesh), the larger the numerical dissipation of the kinetic energy.

Table 5: The volume of the oscillating bubble at time = 0 and 10

| Interface nodes No. | initial volume | volume at time = 10 | relative change |
|---|---|---|---|
| 16 | 4.14879 | 4.14875 | $-9.6 \times 10^{-6}$ |
| 32 | 4.17896 | 4.17897 | $2.4 \times 10^{-6}$ |
| 64 | 4.18652 | 4.18653 | $2.4 \times 10^{-6}$ |
| 128 | 4.18841 | 4.18842 | $2.4 \times 10^{-6}$ |

In the next numerical test, we compute the oscillation of a bubble as well as a droplet in a vacuum at Oh $= 10^{-3}$. For this value of Oh, we set $\nu = 0.001414$. The mesh used in this numerical experiment has 256 nodes on the interface. Fig. 14 shows the evolution of mode 2 alongside the analytic solution. The top figure is the result of an oscillating droplet and the bottom figure is that of an oscillating bubble. We observe that the droplet oscillation decays much slower than does the bubble. Both numerical results are in excellent quantitative agreement with the respective analytic solutions. Simulations of an oscillating droplet/bubble at Oh $\sim 10^{-3}$ or smaller are rarely seen in the literature. To our knowledge, this is the first successful comparison with Prosperetti's theory at such a small value of Oh, validating the accuracy of our method and its capability in eliminating the *spurious* currents. 2D oscillations of a quarter liquid drop floating in a lighter fluid has been simulated for Oh $= 0$ in [33], which uses an upwinding method with grid-size dependent numerical viscosity. On a $512 \times 512$ mesh, the numerical viscosity results in a Laplace number La $\approx 10^7$, equivalent to Oh $\approx 3 \times 10^{-4}$.

Exponential decay of the kinetic energy of a 2D oscillating drop was reported in Fuster *et al.* [33]. Fig. 15(a) shows the evolution of the kinetic energy $E = \int_\Omega \rho u^2 dV$ of an axisymmetric bubble for Oh $= 0.01$, 0.005 and 0.001 for mode $n = 2$, exponential decays $E \sim \exp(-bt)$ are observed with the decay rate $b$ roughly proportional to the kinematic viscosity: $b = 0.485$, 0.258 and 0.056 respectively (Fig. 15(b)). Prosperitti's solution in [26] does not provide the energy evolution, so an explicit comparison cannot be done.

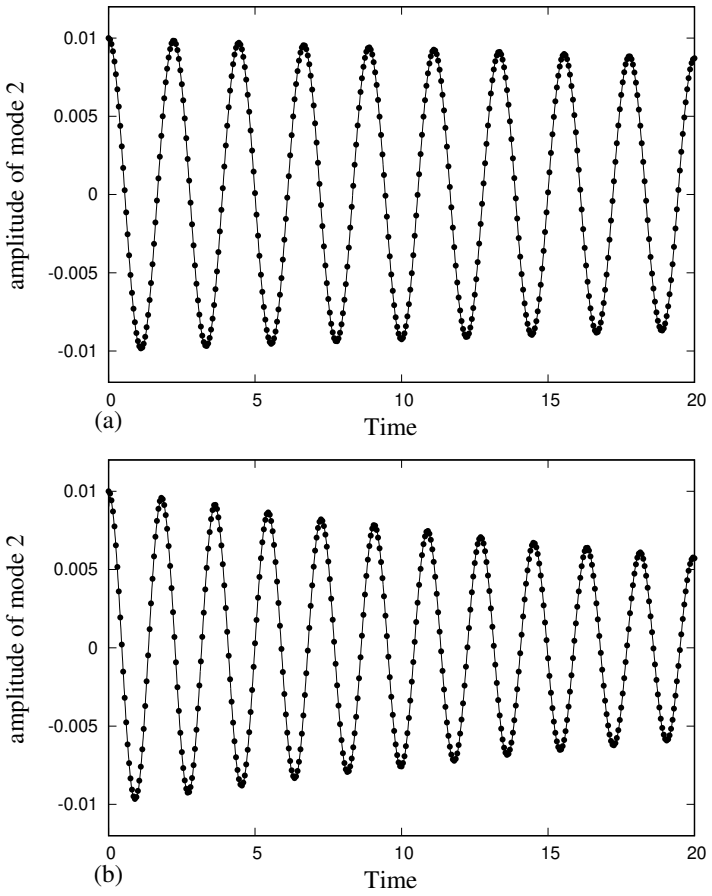


Figure 14: Oh = $10^{-3}$. Comparisons between the analytic solution (in solid lines) and the simulation results (in discs) of an (a) oscillating droplet, and (b) oscillating bubble.

However, his solution is only different from the classical normal-mode analysis [44] due to viscous diffusion of vorticity. For low values of viscosity (*i.e.*, small Oh), the differences between the normal-mode solution and Prosperetti's solution is negligible; in the limit of $\nu = 0$, they are the same. In the normal-mode solution, the amplitude decays with a rate $b_n = (n+2)(2n+1)\nu/R^2$; the velocity potential is proportional to the amplitude, hence decays with rate $b_n$. The kinetic energy $E$ should decays with rate $2b_n$. This gives a value 0.05656 for Oh = 0.001. It is very close to our numerical value 0.056, the relative difference is 1%. For Oh = 0.01, $2b_n = 0.5656$, the numerical decay rate is 0.485, the relative difference is larger as expected because the

viscosity effect becomes more prominent and the error of the normal-mode analysis is large.

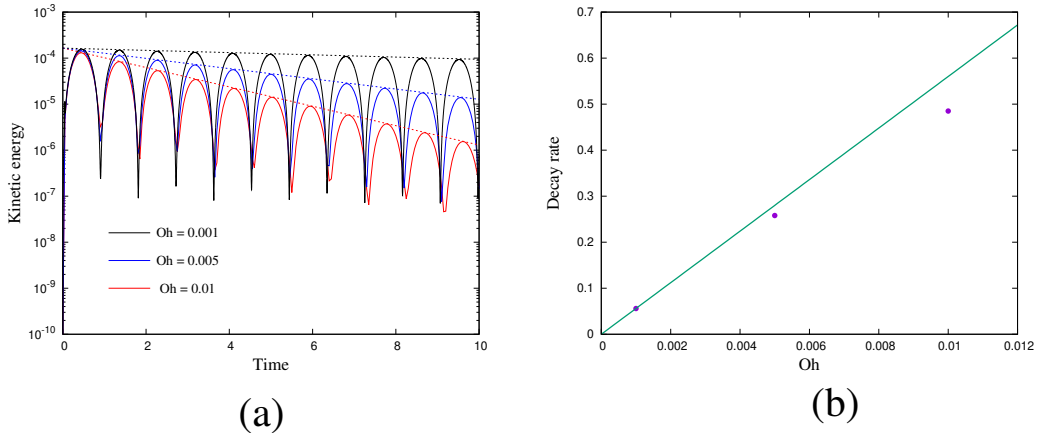


Figure 15: (a) Evolution of the kinetic energy $E$ as a function of time for Oh = 0.01, 0.005 and 0.001. (b) Kinetic energy decay rate as function of Oh.

### *4.3. Pendant drop*

A dripping faucet is a fascinating phenomenon which has been studied for at least two interesting reasons. One is its long-term behavior as a chaotic dynamical system. It was shown that the dripping time intervals exhibit various types of periodic and chaotic behaviors. A simple "mass-on-a-spring" model and an Ising-like stochastic model have been used to interpret the observed phenomena [34]. The other interesting aspect, pertaining to the present work, is the hydraulic behaviors of the drop formation near the breakup point. Breakup of a drop is a critical phenomenon related to a singularity of the nonlinear Navier-Stokes equation [35]. The handling of the topological change of the interface is challenging theoretically as well as numerically. A first detailed photographic account of this bifurcation process was provided by Peregrine *et al.* [27], which is used here to validate our method.

The experiment in [27] is of water dripping out a glass tube of diameter $D = 5.2\,\mathrm{mm}$. The flow is so slow that it can be approximated as a series of quasi-equilibrium states at the early stage. Hence, we start our study of a pendant drop at equilibrium. For a faucet of a sufficiently small diameter $D$, drops with a volume less than a threshold volume are stable and

axisymmetric. We restrict ourselves to this case. There are two controlling dimensionless numbers. The first is chosen as the ratio between the gravitational force and the surface tension force, *i.e.*, Bond number:

$$\mathrm{Bo} := \frac{\rho g D^2}{\sigma} \ . \tag{17}$$

For a water drop hanging from a tube of diameter $D = 5.2\,\mathrm{mm}$, $\mathrm{Bo} = 3.64$. The other parameter is chosen as the dimensionless volume

$$V = \frac{\text{drop volume}}{D^3} \ . \tag{18}$$

Without loss of generality, we consider a free-surface problem in which the ambient air pressure is assumed to be a constant and set to 0. Let $s$ denote the arc-length of the drop interface starting from the axial axis (cf. Fig. 16(a)). The interface is represented as $(r(s), z(s))$ in the cylindrical coordinate system. Consider also the angle $\theta$ between the tangent of the interface and the axial axis. These four geometrical quantities satisfy two simple equations:

$$\frac{\mathrm{d}r}{\mathrm{d}s} = \sin\theta \quad \text{and} \quad \frac{\mathrm{d}z}{\mathrm{d}s} = -\cos\theta \ . \tag{19}$$

Finally, the shape of the drop is determined by the boundary condition at the interface, *i.e*, the balance between the gravitational force and surface tension force:

$$\sigma\left(\frac{\mathrm{d}\theta}{\mathrm{d}s} - \frac{\cos\theta}{r}\right) = -\rho g z \quad \Rightarrow \quad \frac{\mathrm{d}\theta}{\mathrm{d}s} = \frac{cos\theta}{r} - \frac{\rho g z}{\sigma} \tag{20}$$

where $g$ is the gravitational acceleration pointing downward in the $z$ direction. The initial condition at the bottom pole of the drop ($s = 0$) is:

$$\begin{cases} r(0) = 0, \\ z(0) = P_b/(\rho g), \\ \theta(0) = \pi/2. \end{cases} \tag{21}$$

where $P_b$ is the unknown pressure at the bottom pole and serves as a controlling parameter that determines the drop shape. For a given value of $P_b$, we integrate Eqs. (19) and (20) as an initial value problem until $r$ reaches the desired value $R$. Then the volume of the drop is calculated:

$$\text{volume} = \int_{z(0)}^{z} \pi r^2 \mathrm{d}z. \tag{22}$$

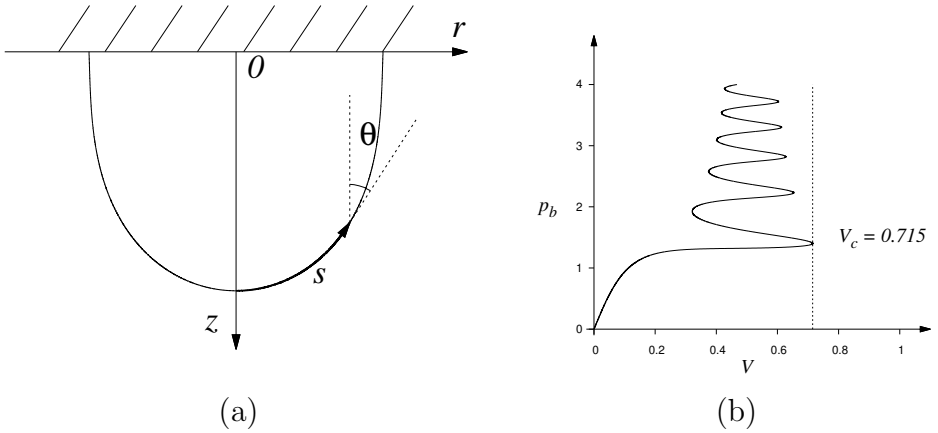


Figure 16: (a) Schematic of a pendant drop; (b) The dimensionless pressure at the bottom of the drop $p_b = P_b/(\rho g D)$ is plotted as a function of the dimensionless volume $V$. Bo = 3.64

For Bo = 3.64, the dimensionless pressure $p_b = P_b/(\rho g D)$ is plotted as a function of the dimensionless volume $V$ in Fig. 16(b). The dimensionless threshold volume is found to be $V_c \approx 0.715$. For drop volume $V < V_c$, the pendant drop may have more than one equilibrium shape, however, only the solution of the branch from the origin to the first turning point in Fig. 16(b) is a stable static solution [36].

We use our Navier-Stokes solver to compute the equilibrium shape of a pendant drop. We start with a free-surface simulation with fluid slowly dripping from the faucet (cf. the next section for the details of the simulation set-up). The interface is initially flat and once the volume of the pendant drop reaches a desired value, the flow rate is then turned off. As we are only interested in the final steady solution, a large viscosity is used (100 times of the viscosity of water) so that the drop shape converges to the equilibrium solution quickly. Since the initial pendant drop volume is zero and the volume influx is small in our simulation, we expect the converged static solution to correspond to the stable equilibrium shape. Fig. 17 displays the comparison between our simulation results and the corresponding stable static solutions. The simulated pendant drop shapes (represented in dots) are right on top of the theoretical shape (drawn in solid lines). Because of numerical errors on a finite mesh, the numerical solutions deviate from the theoretical solutions. A convergence test was performed for $V = 0.562$. The relative error of the

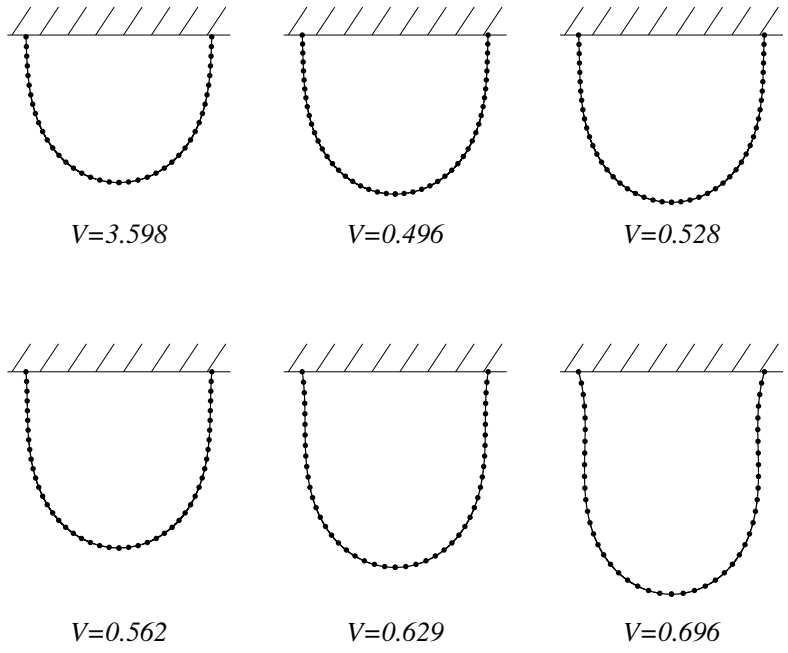


Figure 17: The equilibrium shape of a pendant drop with different volume $V$ obtained from the numerical solution of the ODEs (in solid line) and from the simulation results of the Navier-Stokes solver (in dots). Bo = 3.64.

coordinate of the drop tip as a function of mesh size $h$ on the interface is shown in Fig. 18 in a logarithm scale and a second-order convergence is clearly demonstrated.

In the above examples, we are only interested in static solutions. We used a large viscosity because the computation converges quickly. In order to demonstrate that our method is free of spurious currents, we have also carried out 2 additional computations with the water viscosity and 10 times the viscosity of water. As shown in Fig. 19, the maximum capillary number are all reduced to the machine precision (order of $10^{-14}$). The reduction of the Ca follows an exponential decay with pre-factors roughly proportional to the values of the viscosity. These computations also demonstrate the ability of our method to eliminate the spurious current in a more complicated situation. We believe this is the first successful example for a non-spherical static interface.

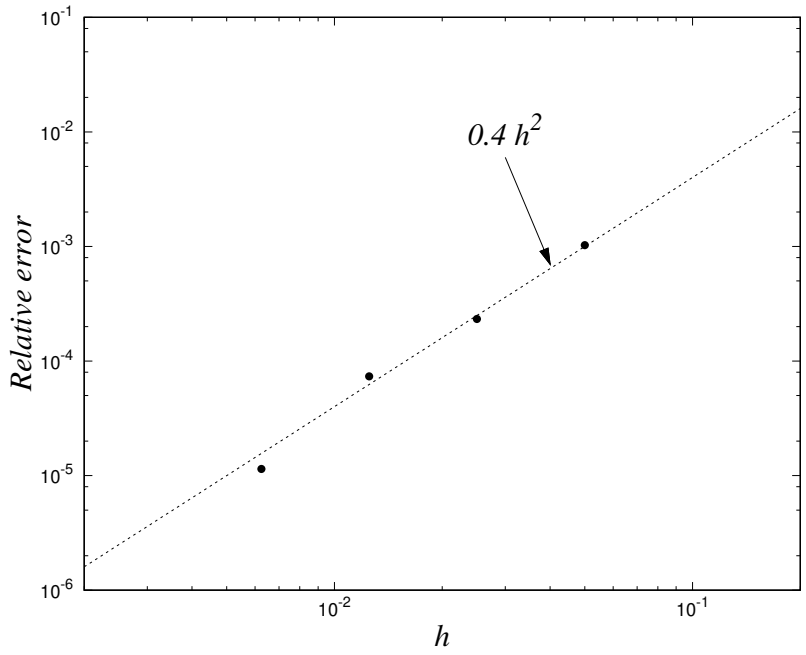


Figure 18: Second-order convergence of the tip coordinate $z$. The discs represents the numerical error as function of interface mesh size $h$ in a logarithm scale, while the dashed straight line has a slope equal to 2.0 in the log scale. $V = 0.562$.

### *4.4. Dripping faucet*

For water dripping slowly out of a faucet, there is a maximum volume for the stable state. Beyond this critical volume the surface tension force is no longer able to hold the drop against gravity; the drop will elongate, form a neck in the middle and eventually break away from the faucet. The breakup process is so rapid that the detail cannot be detected by the naked eye. In the experiment of Peregrine et al. [27], the inlet flow-rate is not explicitly given. Instead, they state that "*the drops were dripping as slowly as possible in order to minimize the initial motion in the pendant mass of water*." This experiment involves a great disparity of spatio-temporal scales, including (1) the initial filling of the drop, probably on the order of *minutes*, (2) the formation of a neck and the transition to a columnar bridge, on the order of *milliseconds*, (3) before and after the two breakups where similarities solutions are relevant, on the order of $100\,\mu$s, and (4) the recoil of a water thread involving up and down capillary traveling waves and its relaxation to a spherical droplet. The entire process has been well reproduced previously

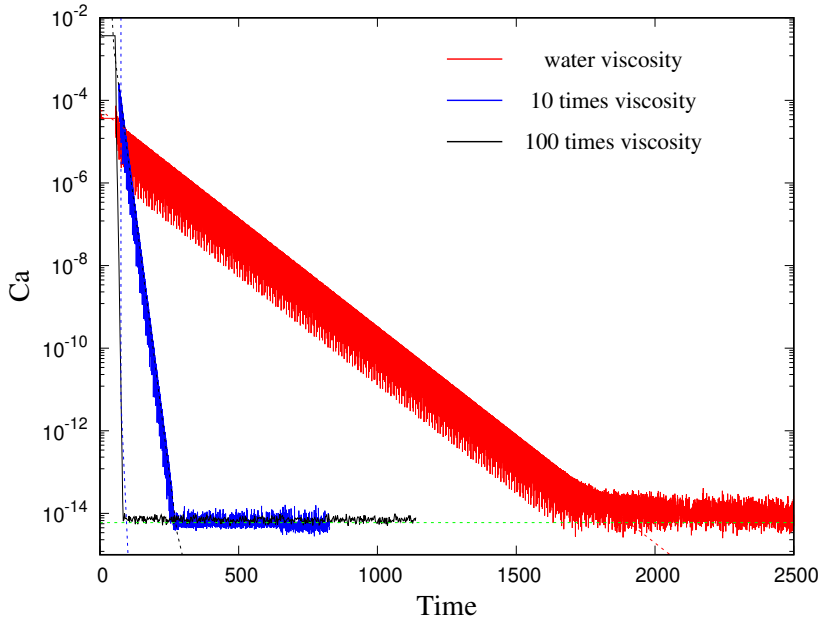


Figure 19: Convergence to the equilibrium shape of a pendant drop. The temporal evolution of $Ca$ as a function of time $t$.

by a number of numerical studies based on one-dimensional theory [35, 37] and the VOF method [38].

The relevant dimensionless physical parameters in the simulation are listed in Table 6. For water, $\rho_1 = 10^3\,\mathrm{kg/m^3}$, $\mu_1 = 10^{-3}\,\mathrm{Pa{\cdot}s}$; and for ambient air, $\rho_2 = 1.25\ \mathrm{kg/m^3}$ and $\mu_2 = 1.85 \times 10^{-5}\,\mathrm{Pa{\cdot}s}$. $\sigma = 0.072\ \mathrm{N/m}$. Due to the weak viscous dissipation ($\mathrm{Oh} = 1.6 \times 10^{-3}$), the very slow filling stage (1) is dominated by the surface tension force and is prone to numerical instability caused by *spurious currents*. Thus this is a good benchmark test for validating both the accuracy and the robustness of a numerical scheme.

Table 6: Dimensionless numbers in the simulation of a dripping faucet. Subscript 1 denotes the dripping water and 2 denotes ambient air in two-phase simulations.

| | |
|---|---|
| $\mathrm{Oh} = \mu_1/\sqrt{\rho_1 \sigma D}$ | $1.6 \times 10^{-3}$ |
| $\mathrm{We} = \rho_1 U^2 D/\sigma$ | $6.40 \times 10^{-4}$ |
| $\mathrm{Bo} = \rho_1 g D^2/\sigma$ | 3.64 |
| $\rho_1/\rho_2$ | 816.33 |
| $\mu_1/\mu_2$ | 45.08 |

*4.4.1. Numerical set-up*

In our numerical simulation, the initial shape of the interface is either flat or spherical, as shown in Fig. 20. At the inlet boundary, a parabolic velocity profile for the velocity is set:

$$\begin{aligned} u(r) &= 0, \\ v(r) &= -U(r^2 - R^2)/R^2. \end{aligned} \tag{23}$$

where $R = 2.6\,\mathrm{mm}$ as in the experiment, and $U$, the maximum velocity at the inlet, is set to a very small value, $3.0 \times 10^{-3}\,\mathrm{m/s}$. Both free-surface and two-phase simulation are performed. For two-phase simulation, in addition to the inlet tube, the computational domain also includes a square of size $20R \times 20R$. The no-slip boundary condition is applied to all solid boundaries, at the top (except at the tube mouth) and vertical boundaries. At the bottom boundary a parabolic velocity profile is imposed in order to satisfy the mass conversation. In our simulations, we rescale the mesh by the tube radius $R$. To ensure a sufficient resolution within the inlet tube, a maximum mesh size of $l_{\mathrm{inlet}} = 0.025$ is set at the inlet boundary and the tube wall boundary. A global maximum mesh size $l_{\max}$ is set to 1.0 to guarantee an adequate mesh resolution in the region far away from the interface. Other mesh parameters are: $\alpha = 0.01$, $G = 0.2$ and $l_{\mathrm{interface}} = 0.02$. An example of the two-phase mesh is shown in Fig. 21: the mesh is almost uniform in regions far from the interface. The mesh size is the smallest on the interface and grows linearly with the distance from the interface.

*4.4.2. Simulation results*

Fig. 22 superposes the interface computed from our simulation (represented in red solid lines) to the photographs from the experiment at different stages [27]. It demonstrates remarkably good agreement between the experiment and the simulation. Frame (1) is the initial phase of necking which has an appreciable up-down symmetry. In frame (2) the neck has given way to an almost cylindrical thread between the pendent mass and the nascent drop. Frame (3) shows a sharp conic shape near the breakup point just after the bifurcation; the interface nearby is far from symmetric up and down. In frame (4), after the separation, the unbalanced surface tension rapidly accelerates liquid in both water bodies, with the formation of a knob at the tip of the thread, and the appearance of a dimple at the top of the new drop. In frame (5), the recoil effects propagate up the conical region onto the remnant of the liquid thread with small amplitude waves traveling ahead the

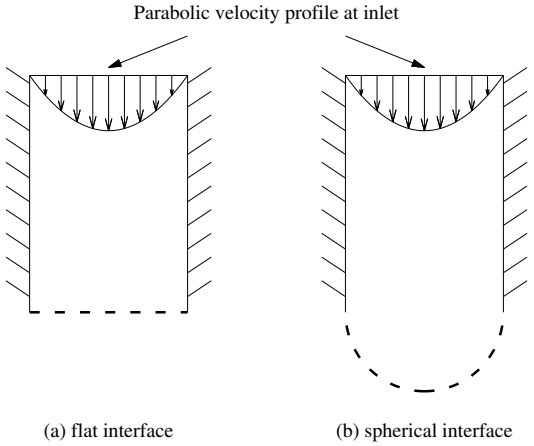


Figure 20: Two initial configurations (thick broken lines represent interfaces): (a) a flat interface, and (b) a spherical interface. At the inlet boundary a parabolic profile is imposed for velocity.

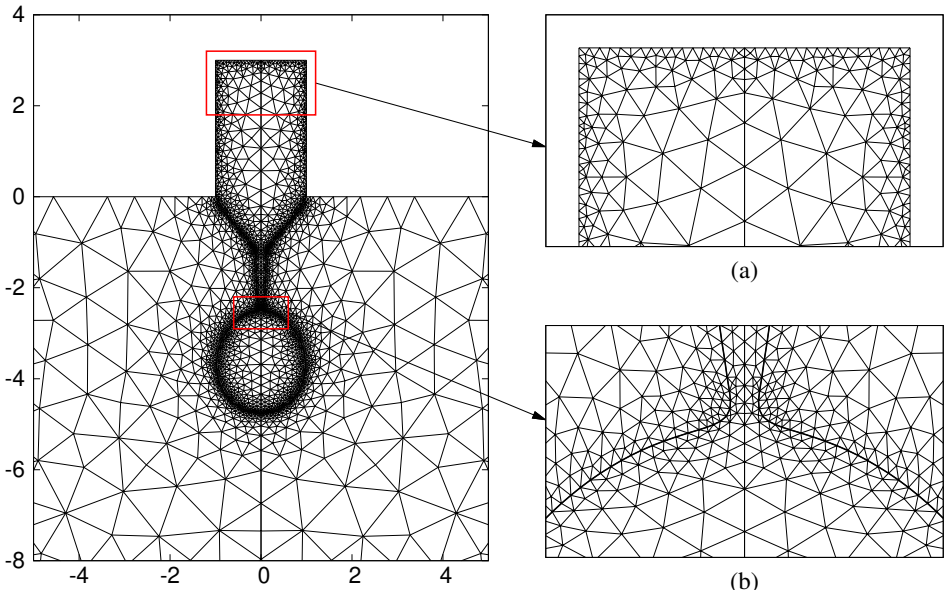


Figure 21: An example of the mesh used in two-phase simulation: (a) a close-up of the inlet boundary; (b) a close-up of the interface.

disturbance like any other capillary ripples. In the meantime, a second bifurcation takes place at the top of the liquid thread. After the second breakup, frames (6)–(9) show the intricate and complicated interaction between up and down traveling capillary waves, and our simulation faithfully captured the three and four major waves on the liquid thread, including a minuscule detail, *i.e.*, the tiny nodule at the top of the middle droplet in frame (9). It should be noted that the photograph is not exactly axisymmetric. For instance, in frame (1), the drop tilts clearly to the right. It is difficult to speculate what the reason is. It may be due to some imperfections in the experiment apparatus, or the flow is not "slow" (quiet) enough while the viscous damping is too weak to damp the oscillations.

The driving mechanism of liquid breakup is surface tension force. Theoretical solutions [39] show that the minimum diameter near the breakup point decreases linearly in time, and similarity solutions have been found. This process is adequately described by the macroscopic continuum theory of fluid mechanics. In a numerical simulation with explicit interface representation, the breakup is performed manually and the global behavior of the system is expected to be insensitive to the exact moment when this surgery is performed as long as the thread is small enough. In the above simulation, this operation is performed when the minimum neck diameter reaches below one hundredth of the faucet diameter. In Fig. 23, the interface shape near the breakup is shown when the minimum neck diameter has reached $10^{-2}$, $10^{-3}$ and $10^{-4}$ of the faucet diameter. The different interfaces overlay on each other very well except near the breakup point. This is consistent with the existence of a similarity solution, and suggests that the breakup size of one hundredth of the faucet diameter is adequate for the purpose of this study. It should be mentioned that the computing cost increases rapidly with the decrease of the neck size without obvious gain in the accuracy of the results. It has been shown in [40] for the first time that when the interface of a viscous drop can overturn before the drop breaks. This phenomenon is also shown in Fig. 23. When the overturning occurs, one-dimensional long-wave theory is no longer applicable.

It is also worth mentioning that the initial condition affects the final breakup shape little. We have performed simulations with two different initial interface shapes: a flat one and a spherical one (cf. Fig. 20). The evolution of the vertical coordinate $z$ of the bottom pole of the drop is plotted as function of volume $V$ for both cases in Fig. 24. Since the inflow flux is constant, $V$ increases linearly in time, this figure can be reinterpreted as the evolution of

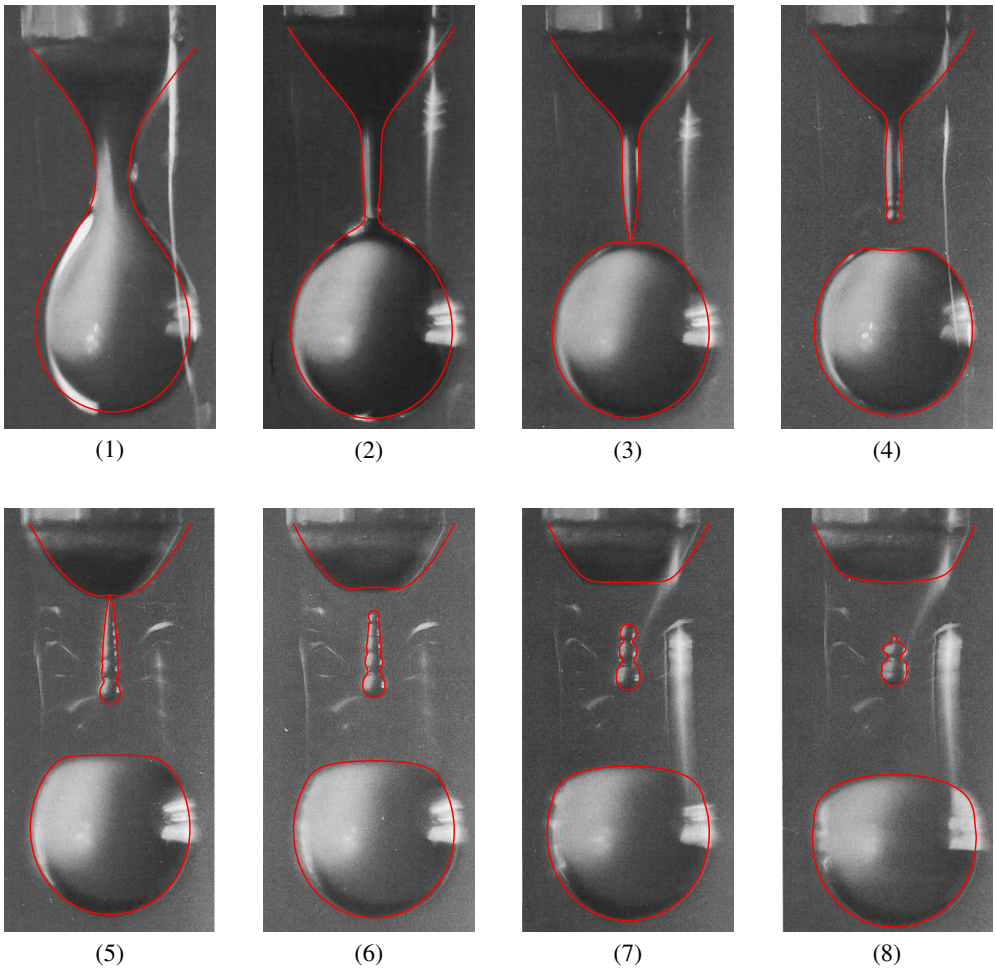


Figure 22: Simulated interface shape (drawn in red solid lines) from a free-surface simulation superposed on experimental results from [27]: (1-2) before the first breakup; (3) the instant after first breakup; (4) recoil of the dripping fluid; (5) the instant after the second breakup; (6-8) oscillation of the secondary drop.

$z$ as function of $t$. The semi-spherical interface is not at equilibrium. The $z$ curve (the blue curve) starting from this shape exhibits oscillation around the curve from the flat initial interface. Once they enter the fast necking stage (near time $t = 2.865$, corresponding to the critical volume $V_c = 0.715$), the two curves converge due to gravity-induced straining. There are no significant discrepancies between the two from this time on (interface shapes not shown

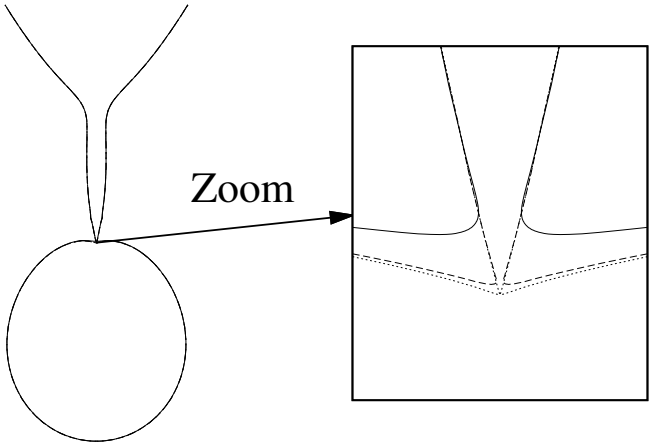


Figure 23: Interface shape and a close-up of the neck region in the free-surface simulation when the neck diameter reaches $10^{-2}$ (solid line), $10^{-3}$ (dashed line) and $10^{-4}$ (dotted line) of the faucet diameter.

here).

The above results are obtained from a two-phase simulation. Fig. 25 compares two-phase and free-surface simulations using the same mesh parameters and initial conditions. The difference between the two is negligible. This justifies the use of the free-surface flow here when one fluid is much lighter and less viscous than the other.

While interface topology changes are handled effortlessly in volume tracking methods such as LS and VOF methods, it is not so for methods based on an explicit representation of the interface. The most difficult task is the re-arrangement of the logical connection between interface nodes: some old connections are to be broken, and new connections are to be established. Assuming that all this is done correctly, the aftermaths interface is very sharp (with high curvature) and irregular at the break-up region, prone to severe numerical instability. Nevertheless, we have found a simple, automatic and robust solution for axi-symmetrical flows. The technical details are described as follows.

The key to success is based on our control of the mesh layout prior to the topology change; we require that the mesh possess only one single layer of triangles in the liquid thread, which is adequate to represent the physical problem accurately [39]. Let's consider the two-phase simulation of the drip-

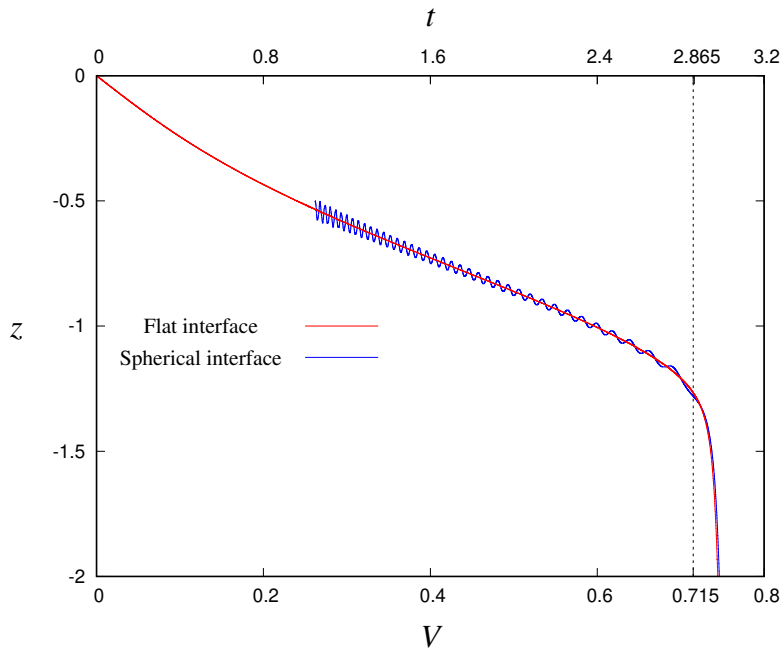


Figure 24: Evolution of the vertical position $z$ of the bottom pole as function of time $t$ and volume $V$. The blue line represents the result starting with a spherical interface and the red line that of flat interface. All quantities in this figure are rescaled with the diameter $D$ and velocity $U$.

ping faucet at an instant prior to breakup as shown in Fig. 25 (1). This is shown in Fig. 26 (a) with a close-up as Fig. 26 (b). Because the flow is axisymmetrical, the computation is performed on a half domain. The thick red line represents the jet interface, and the left boundary ($r = 0$) the symmetry axis. In this example, close to the breakup point, the distance $d$ between the interface and the symmetry axis is so small that it determines the ideal length scale $l_s$. We use $\beta = 1.0$, $r_{\min} = 1.0$ and $r_{\max} = 2.8$ in this calculation, from Eqn. (10) in [18], $l_s = d$, all edges shorter than $r_{\min}\beta l_s = d$ are contracted; that is, there will be no edge shorter than $d$ near the breakup point. Only edges of length larger than $r_{\max}\beta d = 2.8d$ are split. The result is a distinctive feature of this mesh: a single layer of triangles within the liquid jet between the interface and the symmetry axis.

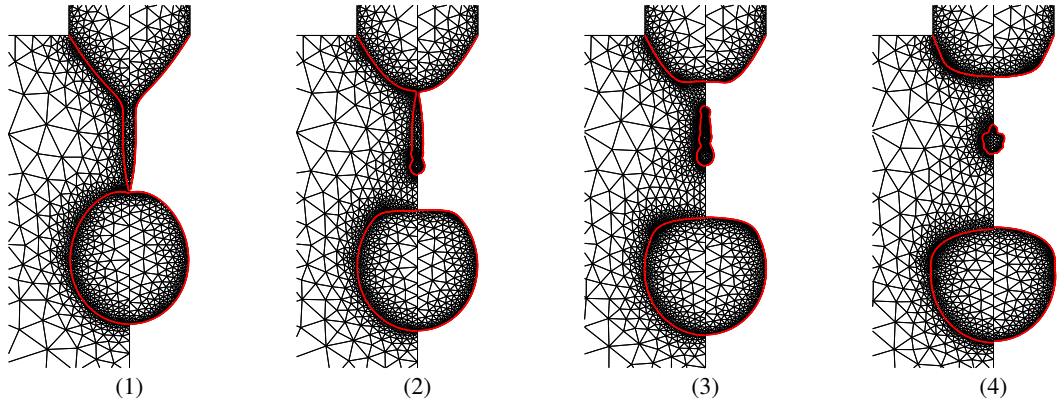


Figure 25: Comparisons of interface shapes from two-phase and free-surface simulations (mesh parameters: $\alpha = 0.02$, $G = 0.2$ and $l_{\text{interface}} = 0.02$). The left/right half of each sub-figures are from free-surface/two-phase simulations respectively. The interface is depicted in red solid lines.

The above simple mesh lay-out simplifies the scenario of the topology surgery to one logical pattern. We introduce a new length scale $l_{\text{break}}$ for the breakup (here $l_{\text{break}} = 0.01$); when the radius of the thread is detected to be smaller than this length at an interfacial node, we perform the breakup surgery. In Fig. 26 (b), this point is indicated by a black square and noted as $N$. We identify its two neighbors $U_1$ and $D_1$ on the interface, and its two neighbors $U_2$ and $D_2$ on the symmetry axis. In Fig. 26 (c), the interface is broken at node $N$: $U_1$ is connected to $U_2$ and $D_1$ is connected to $D_2$; $U_1N$ and $ND_1$ are no longer interfacial edges. The logical handling of this interface surgery is straightforward and the computing cost minimal. In doing this, we have switched the three triangles enclosed by line $U_2U_1ND_1D_2$ from the liquid phase to the gas phase. There is a small violation of mass conservation, but the change of mass for both phases is tiny, of order of $l_{\text{break}}^3$, hence of no concern. The topological change leaves two high-curvature tip regions. In order not to complicate the computation, we limit the minimal mesh size $l_{\min}$ to the same of order of $l_{\text{break}}$, so that the curvature $\kappa$ has no effect in determining the ideal mesh size $l_s$ near the surgery node. After 100 iterations, the two high-curvature tip regions were relaxed to be of moderate curvature, as shown in Fig. 26 (d).

This process was applied also automatically to the second breakup shown in Fig. 25 (2). Indeed, the algorithm was implemented in such way that it

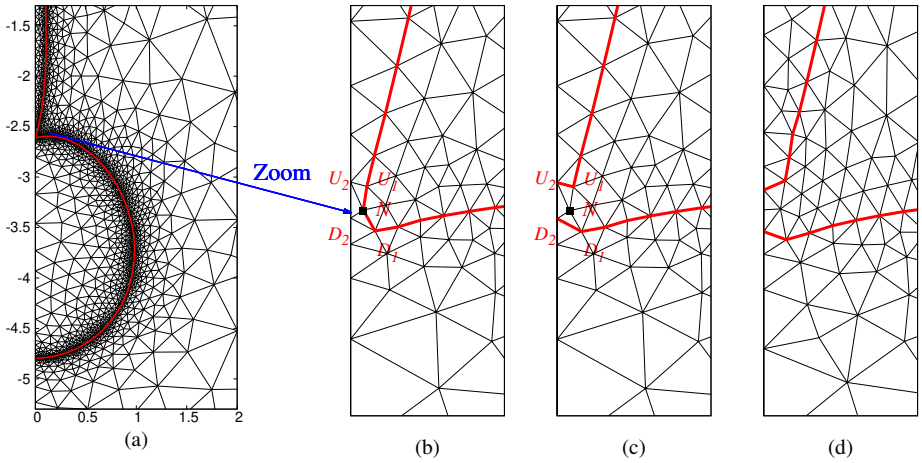


Figure 26: Sequences of mesh topological surgery in a simulation of a jet breakup (the interface is depicted by the thicker lines): (a) a global view of the jet; (b) a close-up shows that the liquid thread consists of only one layer of triangles; (c) identification of the surgery node (marked by a square) and interface reconnection; (d) 100 iterations after interface reconnecting.

can handle any number of jet breakup "on the fly". From the numerical point of view, the above algorithm applies also to the case of droplet coalescence. However, the Naiver-Stokes plus the surface tension are not sufficient to predict this flow. The final stage of the coalescence (the rupture of the gas film) involves the rarefied gas effect and *van der Waals* forces, hence length-scale of order of magnitude 10 nm. liquid thread breakup problem is not very sensitive to the breakup length $l_{\text{break}}$ because the flow is controlled by macro-physical mechanisms (see Fig. 23), it is not so for droplet coalescence; over-estimating this length will lead to wrong outcome. The details are elaborate and beyond the scope of this paper.

## 5. Conclusions

To resolve the intrinsic disparities of length-scales, mesh adaptivity is an absolute necessity for multiphase flow simulations [41–43]. In this work, we use an *interface-conforming* method which explicitly represents the interface by lines of the mesh system in 2D. The interface conformity renders the approximation of the dynamic boundary condition easy and accurate. Through constant remeshing, our adaptive algorithm adjusts the mesh so that it follows the evolution of the interface, even when the interface undergoes large

deformations. The incompressible Navier-Stokes equations are discretized by Taylor-Hood elements and solved with the *Uzawa* algorithm. Special care must be paid to the pressure jump across the interface. We introduce the $P_1d$ space, in which the pressure on the interface has two values. From this configuration we deduce the force balance between interface tension and the associated pressure jump at a circular interface (cf. Sec. 3.4). As a result, the exact *Laplace* solution of a static circular bubble/drop is obtained and later shown to be numerically stable for large timescales. The accuracy of our method is validated through the simulations of bubble/drop oscillations, stationary pendant drops and the dripping faucet. We specifically choose difficult numerical tests ($\mathrm{Oh} \ll 1$) to evaluate the performance of our method to eliminate *spurious* currents. Excellent agreement with corresponding analytic/experimental results demonstrates that our method is of high precision and robustness.

**Acknowledgment**

ZKC would like to thank the China Scholarship Council and the Cambridge Commonwealth, European and International Trust for their financial support. CYL and LSL is supported by the CSRC under the NSAF grant U1530401. And JL would like acknowledge the support from CSRC for his visit to CSRC in the summer of 2018 during which part of this work is performed. The authors would like to thank Mr. Charles Armstrong of ODU for his proofreading the manuscript and editorial assistance.

## Appendix A. Existence of static equilibrium solution with axisymmetry

Consider a spherical droplet with radius $R$ at rest and free of external forces. The static *Laplace* solution for a spherical droplet is: the velocity $\boldsymbol{u} := (u,\, v) = \mathbf{0}$; the pressure is constant inside the droplet ($p_1$) and outside the droplet ($p_2$); $p_1 - p_2 = 2\sigma/R$. In this case, the weak formulation of the momentum equations (20) and (21) in [18] are reduced to

$$-\int_\Omega \frac{\partial(r\phi)}{\partial r} p \mathrm{d}r\mathrm{d}z = \int_{\Gamma^\mathrm{N}} \sigma\kappa n_r r\phi \mathrm{d}s \;, \tag{A.1a}$$

$$-\int_\Omega \frac{\partial(r\phi)}{\partial z} p \mathrm{d}r\mathrm{d}z = \int_{\Gamma^\mathrm{N}} \sigma\kappa n_z r\phi \mathrm{d}s \;, \tag{A.1b}$$

where $\boldsymbol{n} := (n_r,\, n_z)$ is the unit normal vector on the interface at $(r,\, z)$. The above equations are the axisymmetric counterparts of the two-dimensional Eq. (9). Now let Fig. 6 again represent a portion of a mesh, where the interface is drawn as a thick solid line, $\Gamma^{\mathrm{N}}$. We must show that at the discrete level the above equations hold for every basis function $\phi$ associated with a node in a Taylor-Hood element.

As in the two-dimensional case, consider first the case of node $i$ associated with an interior vertex as shown in Fig. 6. Let $\Omega_i$ denote the shaded region composed of the triangles around node $i$. The basis function $\phi_i$ vanishes outside $\Omega_i$ including on boundary $\partial\Omega_i$, hence $\phi_i = 0$ on $\Gamma_N$. The right hand side of Eqn. (A.1a) is zero and its left hand side:

$$\int_\Omega \frac{\partial(r\phi_i)}{\partial r} p \mathrm{d}r\mathrm{d}z = p_1 \int_{\Omega_i} \frac{\partial(r\phi_i)}{\partial r} \mathrm{d}r\mathrm{d}z = p_1 \int_{\partial\Omega_i} r\phi_i n_r ds = 0 \ . \tag{A.2}$$

We have used the fact that within $\Omega_i$ the pressure is a constant and that $\phi_i = 0$ on $\partial\Omega_i$. Consider node $l$ associated with an interior edge as shown in Fig. 6, and denote $\Omega_l$ the shaded region composed of the 2 triangles around this edge. Eqn. (A.1a) for the basis function $\phi_l$ is readily derived if $i$ is replaced by $j$ and $p_1$ by $p_2$ in the above calculation.

If node $j$ is associated with a vertex on the interface (shown in Fig. 6), let $\Omega_j$ denote the shaded region composed of the triangles around this vertex $j$, and $\Gamma_j$ denote the portion of the interface in $\Omega_j$. $\Omega_j$ is composed of $\Omega_{j,1}$ in fluid 1 and $\Omega_{j,2}$ in fluid 2. The numerical value of the curvature $\kappa = 2/R$ (exact up to the machine accuracy). The pressure is not a constant in $\Omega_j$, but is constant in $\Omega_{j,1}$ and $\Omega_{j,2}$, respectively. Eq. (A.1a) then becomes:

$$\begin{aligned}
-p \int_{\Omega_j} \frac{\partial(r\phi_j)}{\partial r} \mathrm{d}r\mathrm{d}z &= -p_1 \int_{\Omega_{j,1}} \frac{\partial(r\phi_j)}{\partial r} \mathrm{d}r\mathrm{d}z - p_2 \int_{\Omega_{j,2}} \frac{\partial(r\phi_j)}{\partial r} \mathrm{d}r\mathrm{d}z \\
&= -p_1 \int_{\partial\Omega_{j,1}} r\phi_j n_r \mathrm{d}s - p_2 \int_{\partial\Omega_{j,2}} r\phi_j n_r \mathrm{d}s \\
&= (p_1 - p_2) \int_{\Gamma_j} r\phi_j n_r \mathrm{d}s \\
&= \frac{2\sigma}{R} \int_{\Gamma_j} r\phi_j n_r \mathrm{d}s = \int_{\Gamma_j} \sigma\kappa r\phi_j n_r \mathrm{d}s \\
&= \int_{\Gamma^{\mathrm{N}}} \sigma\kappa r\phi_j n_r \mathrm{d}s.
\end{aligned}$$

Note that $\phi_j$ vanishes on the boundary of $\Omega_j$ but not on $\Gamma_j$, on which $n_r$ is of opposite sign in $\Omega_{j,1}$ and $\Omega_{j,2}$. Now consider node $m$ associated with an interface edge as shown in Fig. 6, and denote $\Omega_m$ the shaded region composed of the 2 triangles around this edge. Eq. (A.1a) for the basis function $\phi_m$ is readily derived if $j$ is replaced by $m$ in the above calculation.

Finally, if the $r$-derivative is replaced by the $z$-derivative in the above calculations, we derive the $z$-momentum equation, Eq. (A.1b). We have therefore completed the proof for the axisymmetric case.

## References


[1] Zhihua Xie. A two-phase flow model for three-dimensional breaking waves over complex topography. *Proc. R. Soc. A*, 471(2180):20150101, 2015.

[2] K. Yokoi, D. Vadillo D, J. Hinch, and I. Hutchings. Numerical studies of the influence of the dynamic contact angle on a droplet impacting on a dry surface. *Phys. Fluids*, 21(7):072102, 2009.

[3] X. Chen, D. Ma, V. Yang, and S. Popinet. High-fidelity simulations of impinging jet atomization. *Atomization Sprays*, 23(12), 2013.

[4] O. Desjardins, V. Moureau, and H. Pitsch. An accurate conservative level set/ghost fluid method for simulating turbulent atomization. *J. Comput. Phys.*, 227(18):8395–8416, 2008.

[5] S. Popinet. Numerical models of surface tension. *Annu. Rev. Fluid Mech.*, 50:1–28, 2018.

[6] R. Scardovelli and S. Zaleski. Direct numerical simulation of free-surface and interfacial flow. *Annu. Rev. Fluid Mech.*, 31(1):567–603, 1999.

[7] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114(1):146–159, 1994.

[8] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y. Jan. A front-tracking method for the computations of multiphase flow. *J. Comput. Phys.*, 169(2):708–759, 2001.

[9] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti. Modelling merging and fragmentation in multiphase flows with SURFER. *J. Comput. Phys.*, 113:134–147, 1994.

[10] J. Li. Calcul d'Interface Affine par Morceaux. *C. R. Acad. Sci. Paris*, 320 Série IIb:391–396, 1995.

[11] H. Ding, P.D.M. Spelt, and C. Shu. Diffuse interface model for incompressible two-phase flows with large density ratios. *J. Comput. Phys.*, 226(2):2078–2095, 2007.

[12] S. Afkhami, S. Zaleski, and M .Bussmann. A mesh-dependent model for applying dynamic contact angles to VOF simulations. *J. Comput. Phys.*, 228:5370–5389, 2009.

[13] S. Popinet. GERRIS: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *J. Comput. Phys.*, 190(2):572–600, 2003.

[14] P. Yue, C. Zhou, J. J. Feng, C. F. Ollivier-Gooch, and H. H. Hu. Phase-field simulations of interfacial dynamics in viscoelastic fluids using finite elements with adaptive meshing. *J. Comput. Phys.*, 219(1):47–67, 2006.

[15] B. Bejanov, J-L. Guermond, and P.D. Minev. A grid-alignment finite element technique for incompressible multicomponent flows. *J. Comput. Phys.*, 227(13):6473–6489, 2008.

[16] S. Quan and D.P. Schmidt. A moving mesh interface tracking method for 3d incompressible two-phase flows. *J. Comput. Phys.*, 221(2):761–780, 2007.

[17] J. Li, M. Hesse, J. Ziegler, and A. W. Woods. An arbitrary Lagrangian Eulerian method for moving-boundary problems and its application to jumping over water. *J. Comput. Phys.*, 208:289–314, 2005.

[18] J. Li. An arbitrary Lagrangian Eulerian method for three-phase flows with triple junction points. *J. Comput. Phys.*, 251:1–16, 2013.

[19] S. Popinet and S. Zaleski. A front-tracking algorithm for accurate representation of surface tension. *Int. J. Numer. Methods Fluids*, 30(6): 775–793, 1999.

[20] S. Popinet and S. Zaleski. Bubble collapse near a solid boundary: A numerical study of the influence of viscosity. *J. Fluid Mech.*, 464:137–163, 2002.

[21] Y. Renardy and M. Renardy. Prost: a parabolic reconstruction of surface tension for the volume-of-fluid method. *J. Comput. Phys.*, 183(2):400–421, 2002.

[22] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, and M. W. Williams. A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. *J. Comput. Phys.*, 213(1):141–173, 2006.

[23] M. Herrmann. A balanced force refined level set grid method for two-phase flows on unstructured flow solver grids. *J. Comput. Phys.*, 227(4): 2674–2706, 2008.

[24] S. Popinet. An accurate adaptive solver for surface-tension-driven interfacial flows. *J. Comput. Phys.*, 228:5838–5866, 2009.

[25] T. Abadie, J. Aubin, and D. Legendre. On the combined effects of surface tension force calculation and interface advection on spurious currents within volume of fluid and level set frameworks. *J. Comput. Phys.*, 297:611–636, 2015.

[26] A. Prosperetti. Free oscillations of drops and bubbles: the initial-value problem. *J. Fluid Mech.*, 100:333–347, 1980.

[27] D. H. Peregrine, G. Shoker, and A. Symon. The bifurcation of liquid bridges. *J. Fluid Mech.*, 212:25–39, 1990.

[28] J. Etienne., E. J. Hinch, and J. Li. A Lagrangian-Eulerian approach for the numerical simulation of free-surface flow of a viscoelastic material. *J. Non-Newtonian Fluid Mech.*, 136:157–166, 2006.

[29] D. G. Dritschel. Contour surgery: A topological reconnection scheme for extended integrations using contour dynamics. *J. Comput. Phys.*, 77:240, 1988.

[30] Olivier Pironneau. *Finite Element Methods for Fluids*. Wiley Chichester, 1989.

[31] P. M. Gresho and R. L. Sani. *Incompressible Flow and the Finite Element Method*, volume 2. Isothermal Laminar Flow. Wiley, New York, 2000.

[32] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *J. Comput. Phys.*, 100(2):335–354, 1992.

[33] D. Fuster, G. Agbaglah, C. Josserand, S. Popinet, and S. Zaleski. Numerical simulation of droplets, bubbles and waves: state of the art. *Fluid Dyn. Res.*, 41(6):065001, 2009.

[34] P. Coullet, L. Mahadevan, and C. S. Riera. Hydrodynamical models for the chaotic dripping faucet. *J. Fluid Mech.*, 526:1–17, 2005.

[35] J. Eggers and T. D. Dupont. Drop deformation in a one-dimensional approximation of the navier-stokes equation. *J. Fluid. Mech.*, 262:205, 1994.

[36] J. F. Padday and A. R. Pitt. The stability of axisymmetric menisci. *Phil. Trans. R. Soc. Lond. A*, 275(1253):489–528, 1973.

[37] N. Fuchikami, S. Ishioka, and K. Kiyono. Simulation of a dripping faucet. *J. Phys. Soc. Jpn.*, 68(4):1185–1196, 1999.

[38] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, and S. Zaleski. Volume-of-fluid interface tracking with smoothed surface stress methods for three-dimensional flows. *J. Comput Phys.*, 152(2):423–456, 1999.

[39] J. Eggers. Nonlinear dynamics and breakup of free surface flows. *Rev. Mod. Phys.*, 69:865–929, 1997.

[40] Edward D. Wilkes, Scott D. Phillips, and Osman A. Basaran. Computational and experimental analysis of dynamics of drop formation. *Phys. Fluids*, 11(12):3577–3598, 1999.

[41] J. Li. Macroscopic model for head-on binary droplet collisions in a gaseous medium. *Phys. Rev. Lett.*, 117(21):214502, 2016.

[42] Zhihua Xie, Dimitrios Pavlidis, Pablo Salinas, James R. Percival, Christopher C. Pain, and Omar K. Matar. A balanced-force control volume finite element method for interfacial flows with surface tension

using adaptive anisotropic unstructured meshes. *Comput. Fluids*, 138: 38–50, 2016.

[43] Changjuan Zhang, Jie Li, Li-Shi Luo, and Tiezheng Qian. Numerical simulation for a rising bubble interacting with a solid wall: Impact, bounce, and thin film dynamics. *Phys. Fluids*, 30(11):112106, 2018.

[44] Horace Lamb. *Hydrodynamics*. Cambridge University Press, Cambridge, UK, 1993.