A coupled immersed interface and level set method for simulation of interfacial flows steered by surface tension

H. V. R. Mittal*[1], Rajendra K. Ray[1], Hermes Gadêlha[2], and Dhiraj V. Patil[3]

[1]School of Basic Sciences, Indian Institute of Technology Mandi, P.O. Box 175005, Kamand, Mandi, India
[2]Faculty of Engineering, University of Bristol, Clifton BS8 1UB, United Kingdom
[3]Department of Mechanical Engineering, Indian Institute of Technology Dharwad, Dharwad, Karnataka, 580011, India

Abstract

This work presents a methodology to address the problems of bubbles and drops evolving in incompressible, viscous flows due to the effect of surface tension. It is based on the combination of a recently developed immersed interface method to resolve discontinuities, with the level set (LS) method to reproduce the evolving interfaces. The paramount feature of this immersed interface method is the use of Lagrange interpolation enclosing grid points positioned in the vicinity of the interface and few exceptional grid points positioned on the interface. Different problems are considered to assert the accurateness of the proposed methodology, involving both simple and complex interface geometries. Precisely, the following problems are addressed: circular flow with a fixed interface, the dispersion of capillary waves, initially circular, ellipse, star shaped bubbles oscillating to an equilibrium state and circular drops deforming in shear flows. The transient evolution of bubbles/drops in terms of their shapes, pressure profiles, velocity vectors, deformation ratios of major and minor axis is analyzed to observe the effect of surface tension. The proposed methodology is seen to recover the exact numerical equilibrium between the surface tension and pressure gradient in the vicinity of complex interface geometries as well, while recreating the flow physics with an adequate level of accuracy with well representation of overall trends. Moreover, the numerical results yield a good level of agreement with the reference data.

Keywords: Immersed Interface; Level set; Surface tension; Multiphase; Navier-Stokes equations; Shear flows; HOC.

1 Introduction

Recently, there is a growing interest to model and simulate the behaviour of liquid drops or gas bubbles that are immersed in viscous fluids. These studies are motivated by the presence of multiple fluid environments in daily life and by numerous scientific and engineering applications. Examples are fluid mixing, fuel injection in engines, bubble column reactors, particulate flows, suspension rheology, biofluid-dynamics such as emulsion rheology, blood flow, to name only a few. To cite another example, deformation and cleavage of cells generally involves multiple drop deformation processes [1, 2, 3]. Moreover, some emerging and rapidly growing technologies like digital microfluidics are fed from the knowledge of the topology of drop or bubbles evolving through microgeometries [4]. Since the shape and size of the droplet determine important properties such as stability, rheology, and particle morphology, it is important to understand the mechanism of bubble deformation, growth and breakup. The success in numerical simulations of these flows lies in the capability of numerical algorithm to address the following dominant issues:
(a) The fluid interfaces must be accurately computed and well represented.
(b) Discontinuity of material properties (eg. density, viscosity) across these interfaces must be effectively treated while accounting for the forces acting on them like surface tension.
(c) The computations should be stable while having advantages like easy implementation, low in time and cost.
In order to address the first issue, two main groups of numerical techniques have been developed in the literature. Namely, Lagrangian methods (e.g. front tracking [5]) and Eulerian methods (e.g. the level set (LS) [6, 7], the volume-of-Fluid (VOF) methods [8, 9, 10]). The location of the interface is explicitly known in Lagrangian methods and consequently the implementation of surface tension and interfacial boundary conditions is straightforward. However, a major shortcoming of this method is that once the interface topology starts varying, "surgical" procedures are required to smooth and distribute the interfacial elements utilized to represent the interface [11]. Such procedures are complicated when interfaces merge or split, specifically in three dimensions (3D) [12]. On the other hand, in Eulerian methods, the location of an interface is implicitly represented by a scalar function in general on an Eulerian mesh, which serves as an indicator and as such it permits interfaces to merge or break with relative ease. The interface is allowed to evolve on a fixed numerical mesh by solving an advection equation involving this indicator function (for example, the LS function or the VOF function). The methodology proposed in the present work considers a set of governing equations with the interfacial conditions embedded in the field equations as source terms. The equations are discretized in a finite-thickness interfacial zone, within which the fluid properties change smoothly [6]. This approach has been undertaken along with the application of the LS method for modelling the interfaces. The LS method introduced in [7] is a numerical approach for implicitly evolving the interface by means of a smooth (Lipschitz continuous) function (denoted by ϕ) defined on the entire physical domain. By initializing the interface as the zero level set of this function, the interface propagates naturally with the evolution of zero level set. Some of its advantages over other methods include its ability to easily handle topological changes like breakup and merging, well generalization to three

dimensions, and its relatively easy implementation.

The Navier Stokes equations for incompressible viscous flows constitute a mainstay for computing multiphase flows. The numerical methodology adopted herein is the classic Chorin pressure projection scheme [13]. This method consists of splitting the solution procedure into distinct steps in which the velocity and pressure are decoupled by introducing a pressure Poisson equation. A literature survey reveals the existence of many numerical studies related to the modifications of this method. For example, Balcazar et al. [14], Chakraborty et al.[15], Francois and shyy [16], Francois et al. [17], Hysing [18], Huang et al. [19], Li and Lai [20], Leveque and Li [21], Torrecilla et al. [22], Tong and Wang [23], Uh and Xu [24] and the references therein. However all these variants of the projection method are fraught with certain shortcomings, mostly with evaluating pressure, for instance, oscillations in the pressure profile in the vicinity of the interface [25, 22]. Additionally, it is vital to accurately compute interfacial quantities such as curvature and normal vectors because they are used to evaluate the surface tension force. Numerical imbalance of the surface tension force and the associated pressure gradient is likely to induce non physical velocities, commonly known as spurious or parasitic currents [14, 18, 11]. These currents can flourish with time and can degrade simulation results significantly. In particular, one has to be extremely careful in ascertaining that the interface topologies projected by computation are not numerical artifacts arising from errors.

In the present paper, a coupling between LS and a recently developed higher order accurate immersed interface method [26] has been proposed to address the problems of drops or bubbles evolving in incompressible, viscous flows due to surface tension force. The paramount feature of this immersed interface scheme is the use of Lagrange interpolation enclosing adjacent grid points and few special grid points named as "the interfacial points" as nodes. Interfacial points are defined as points in the domain where grid lines intersect the interface. The use of this methodology in conjunction with the LS method allows us to deal with intricate interface geometries with a very simple mesh generation process. For a fixed interface, this methodology has been thoroughly validated in one, two and three dimensions [26]. In the present work, this methodology is extended to solve moving interface problems governed by N-S equations for the first time. Following problems are considered to assert the accurateness of proposed methodology: dispersion of capillary waves for validation, circular flow with a fixed interface; static circular bubble at equilibrium in a fluid at rest, initially ellipse shaped and star shaped bubbles oscillating to an equilibrium state and drops deforming in shear flows. **For the test cases considered here, the methodology is seen to recover the exact numerical equilibrium between the surface tension and pressure gradient in the vicinity of high curvature interface geometries as well. Accurate resolution of pressure is demonstrated without the presence of any oscillations in the vicinity of the interface validating the Young-Laplace's law to a good level of accuracy. Grid refinement analysis of the infinity error norms of pressure and velocities report the convergence rates near to two along with significantly reduced spurious currents. In addition, the primary traits of physical behaviours of drop deformations in shear flows are strongly captured**

and numerical results yield a good level of agreement with the reference data.

2 Mathematical Formulation

The mathematical formulation of the proposed methodology is presented in the present section. The equations depicting the conservation of mass and momentum of two unsteady, viscous, incompressible, immiscible fluids are given by the the Navier Stokes equations defined on a spatial domain Ω with boundary $\partial\Omega$ as [6]:

$$\rho_i \left(\frac{\partial \mathbf{u}_i}{\partial t} + \mathbf{u}_i \cdot \nabla \mathbf{u}_i \right) = \nabla \cdot \mathbf{S}_i + \rho_i \mathbf{g}, \tag{1}$$

$$\mathbf{S}_i = -p_i \mathbf{I} + \mu_i \left(\nabla \mathbf{u}_i + (\nabla \mathbf{u}_i)^T \right) \tag{2}$$

and

$$\nabla \cdot \mathbf{u}_i = 0 \tag{3}$$

where t is the time, ρ is the density, μ is the viscosity, p is the pressure, $\mathbf{u} = (u, v)$ is the velocity field, $\mathbf{g}$ is the acceleration due to gravity, $\mathbf{S}$ is the stress tensor, $\mathbf{I}$ is the identity tensor and the super-index T represents the transpose operator. Fig. 1 shows an illustration of a square domain Ω embedded with an interface Γ separating two immiscible fluids. The apex $i = 1, 2$ represents variables correlating to the regions labelled as fluid 1 and fluid 2 (see Fig. 1) respectively. Assuming no mass transfer between the fluids yields a continuous velocity condition at the interface:

$$\mathbf{u}_1 = \mathbf{u}_2 \quad \text{at} \quad \Gamma. \tag{4}$$

The effect of surface tension is to balance the jump of the normal stresses along the fluid interface. Overlooking the variations of the surface tension coefficient, σ gives the following boundary condition for momentum conservation at the interface [14, 6]:

$$(\mathbf{S}_1 - \mathbf{S}_2) \cdot \mathbf{n} = \sigma \kappa \mathbf{n} \quad \text{at} \quad \Gamma \tag{5}$$

where $\mathbf{n}$ is a unit outward normal vector along Γ. The resulting evolution equations can be mingled into a set of equations for a single fluid in Ω, with the effect of surface tension in terms of a singular source term [14]:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot \left[\mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right) \right] + \rho \mathbf{g} + \sigma \kappa \mathbf{n} \delta_\Gamma \tag{6}$$

$$\nabla . \mathbf{u} = 0 \tag{7}$$

where δ_Γ is the Dirac delta function (ensuring that σ is only applied at the location Γ).

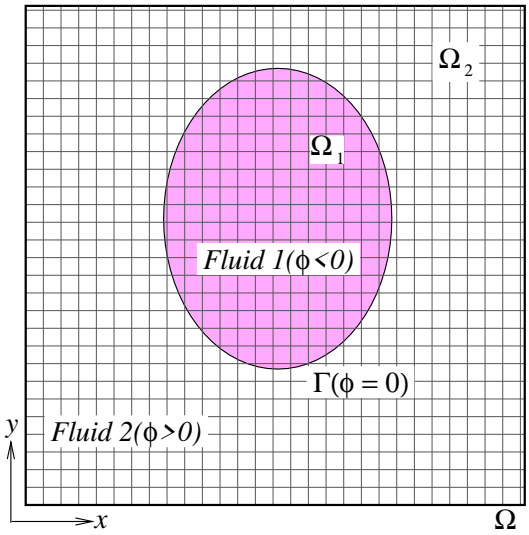


Figure 1: Illustration of a computational domain, Ω showing the evolution of a level set function, $\phi(\mathbf{x}, \mathbf{y})$ with a zero contour marking the interface, Γ.

2.1 Interface Capturing

The position of the interface is evolved in time using the LS method, introduced in [7]. A smooth level set function $\phi(\mathbf{x}, t)$ is defined over Ω in such a way that $\phi > 0$ in one fluid region (i.e. in Ω_2) and $\phi < 0$ in the other fluid region (i.e. in Ω_1) and the interface position is characterized by $\phi = 0$. The time evolution of the interface is obtained solving the Hamilton-Jacobi equation for ϕ:

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0. \tag{8}$$

At each time, t the fluid domain Ω_i are defined by the sign of ϕ and the parameters required to impose the jump conditions at the interface Γ are obtained from the level set function $\phi(\mathbf{x}, t)$. For some problems, a source term is added to right hand side of eq. (8) to maintain the mass conservation in the solution process ([27]). This source term is the difference between the current mass at each time step during the solution process and the initial mass.

The accurate computation of the curvature, κ and the normal, $\mathbf{n} = \nabla\phi/|\nabla\phi|$ desires the values of ϕ to be nearly a signed distance function, that is $|\nabla\phi| = 1$. The numerical solution of eq. (8) rapidly loses the signed distance property during larger time computations, thus a correction is introduced solving the so-called re-initialization equation [28] for few steps in the pseudo-time Θ:

$$\frac{\partial \phi}{\partial \Theta} + S(\phi_0)\left(|\nabla\phi| - 1\right) = 0. \tag{9}$$

The operator $S(\phi_0) = \phi_0/\sqrt{\phi_0^2 + h}$ is a smoothed sign function, $\phi_0 = \phi(\Theta = 0)$ is the initial zero-level set function and h is the grid spacing.

2.2 Surface Tension Treatment

Implementation of a pertinent surface tension model is necessary for an accurate calculation of curvature and incorporation of pressure jump in to the fluid domain. By the means of ϕ, the singular source term, $\sigma\kappa\mathbf{n}\delta_\Gamma$ in (6) can be expressed as [6, 29]:

$$\sigma\kappa\mathbf{n}\delta_\Gamma = \sigma\kappa\delta(\phi)\nabla\phi \tag{10}$$

The curvature κ can be expressed in terms of ϕ and its derivatives as

$$\kappa(\phi) = \frac{\phi_y^2\phi_{xx} - 2\phi_x\phi_y\phi_{xy} + \phi_x^2\phi_{yy}}{(\phi_x^2 + \phi_y^2)^{3/2}} \tag{11}$$

2.3 Governing equations

This conversion of the singular source term in (10) enables us to reformulate the evolution equations (6)-(7) as:

$$\rho\left(\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) = -\nabla p + \nabla\cdot\left[\mu\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)\right] + \rho\mathbf{g} + \sigma\kappa(\phi)\delta(\phi)\nabla\phi, \tag{12}$$

and

$$\nabla.\mathbf{u} = 0 \tag{13}$$

All quantities in the above-mentioned equations are made dimensionless using the following reference values:

$$\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L},\ \tilde{\mathbf{u}} = \frac{\mathbf{u}}{U}, \tilde{t} = \frac{tU}{L},\ \tilde{p} = \frac{p}{\rho_{\text{ref}}U^2},\ \tilde{\mathbf{g}} = \frac{\mathbf{g}}{g_{\text{ref}}}$$
$$\tilde{\rho} = \frac{\rho}{\rho_{\text{ref}}},\ \tilde{\mu} = \frac{\mu}{\mu_{\text{ref}}}$$

where L is the characteristic length scale; the subscript ref is for the reference values being equal to the values of phase in domain Ω_2. Therefore, the dimensionless form of equations (12) and (13), in which the superscripts $\tilde{}$ are omitted for convenience, are

$$\rho\left(\frac{\partial\mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) = -\nabla p + \frac{1}{\text{Re}}\nabla\cdot\left[\mu\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)\right] + \frac{1}{\text{Fr}}\rho\mathbf{g} + \frac{1}{\text{We}}\kappa(\phi)\delta(\phi)\nabla\phi, \tag{14}$$

$$\nabla.\mathbf{u} = 0 \tag{15}$$

where Re is the Reynolds number, We is the Weber number, and Fr is the Froude number, defined as

$$\text{Re} = \frac{\rho_{\text{ref}}UL}{\mu_{\text{ref}}},\ \ \text{We} = \frac{\rho_{\text{ref}}U^2L}{\sigma},\ \ \text{Fr} = \frac{U^2}{\mathbf{g}_{\text{ref}}L}$$

Since, physical properties change discontinuously across the interface, a Heaviside function, H is utilized to distinguish these fluid properties in each domain, yielding the following equations:

$$\rho = \lambda + (1-\lambda)H \quad \text{and} \quad \mu = \eta + (1-\eta)\text{H}$$

where $\lambda = \rho_1/\rho_2$ is the density ratio, $\eta = \mu_1/\mu_2$ is the viscosity ratio and the value of H is equal to one in Ω_1 and zero elsewhere.

The sharp changes in these physical properties across the interface Γ can pose numerical difficulties [6, 29]. In order to resolve these issues, the interface is given a fixed thickness ϵ proportional to mesh spacing h. Firstly, a regularization for δ is introduced using H. Following [30], the regularized delta function, δ_ϵ (where ϵ is typically set to some integral or fractional multiple of h) is defined as

$$\delta_\epsilon(\phi) = \begin{cases} \frac{1}{2}\left(1 + \cos(\frac{\pi\phi}{\epsilon})\right)/\epsilon, & \text{if } |\phi| < \epsilon \\ 0, & \text{elsewhere} \end{cases}$$

H_ϵ is defined as

$$H_\epsilon(\phi) = \begin{cases} 0 & \text{if } \phi < -\epsilon \\ (\phi+\epsilon)/(2\epsilon) + \sin\left(\frac{\pi\phi}{\epsilon}\right)/(2\pi), & \text{if } |\phi| \leqslant \epsilon \\ 1, & \text{if } \phi > \epsilon \end{cases}$$

The above Heaviside function satisfies the relation $\frac{dH_\epsilon(\phi)}{d\phi} = \delta_\epsilon(\phi)$. Using H_ϵ, the corresponding regularized viscosity, μ_ϵ and regularized density, ρ_ϵ can be written as follows:

$$\rho_\epsilon = \lambda + (1-\lambda)H_\epsilon(\phi) \tag{16}$$

$$\mu_\epsilon = \eta + (1-\eta)H_\epsilon(\phi) \tag{17}$$

With this regularisation, the resulting equations (14)-(15) are well posed [6], written as follows:

$$\rho_\epsilon\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) = -\nabla p + \frac{1}{\mathrm{Re}}\nabla\cdot\left[\mu_\epsilon\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)\right] + \frac{1}{\mathrm{Fr}}\rho_\epsilon\mathbf{g} + \frac{1}{\mathrm{We}}\kappa\delta_\epsilon\nabla\phi, \tag{18}$$

$$\nabla.\mathbf{u} = 0 \tag{19}$$

2.4 Interface Jump conditions

The interface normal velocity jump condition, in absence of mass transfer, can be written as:

$$[\mathbf{u}]\cdot\mathbf{n} = 0, \tag{20}$$

where the operator $[u] = u_2 - u_1$ at Γ and $\mathbf{n} = (n_1 \quad n_2)^T$ is the interface normal unit vector. Enforcing the no-slip condition at the interface, the tangential velocity jump condition can be written as:

$$[\mathbf{u}]\cdot\mathbf{t} = 0, \tag{21}$$

where $\mathbf{t}$ is the interface tangent unit vector. Combining eq. (20) - (21), the velocity jump condition at the interface is $[\mathbf{u}] = 0$. Integrating eq. (18) across Γ and considering the effects of the the surface tension, the following stress jump condition is obtained:

$$\left[\bar{\mathbf{T}}\right] \cdot \mathbf{n} - \sigma \mathbf{n} \left(\nabla \cdot \mathbf{n}\right) = 0, \tag{22}$$

where σ is the constant surface tension, $-\nabla \cdot \mathbf{n} = \kappa$, the interface curvature and $\bar{\mathbf{T}} \cdot \mathbf{n}$ is the stress tensor. In this case the non-uniform surface tension contribution $\nabla\sigma$ is neglected, thus the tangential stress component jump $\left(\bar{\mathbf{T}} \cdot \mathbf{n}\right) \cdot \mathbf{n} = 0$. For a Newtonian fluid, the interface normal component of the jump condition (22) yields:

$$\left[p - 2\mu \frac{\partial \mathbf{u} \cdot \mathbf{n}}{\partial \mathbf{n}}\right] = \sigma\kappa. \tag{23}$$

Equation (23) can be rewritten as follows:

$$[p] - 2\left[\mu\right] \frac{\partial \mathbf{un}}{\partial \mathbf{n}} = \sigma\kappa, \tag{24}$$

where $[p] = p_2 - p_1$ at Γ and $[\mu] = \mu_2 - \mu_1$ at Γ.

3 Numerical Method of Solution

The set of governing equations (18)-(19) posed in Section 2.3 are discretized on a uniform Cartesian grid. Without loss of generality, the solution domain, Ω is considered to be a rectangle with a curved interface, Γ (see Fig. 1). The governing equations are solved by the means of standard projection method [13], which comprises of three different steps. In the first step, an intermediate velocity, $\mathbf{u}^*$ is computed explicitly by ignoring the pressure gradient term

$$\rho_\epsilon^n \left(\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n.\nabla\mathbf{u}^n\right) = \frac{1}{\mathrm{Re}}\nabla \cdot \left[\mu_\epsilon^n \left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)\right] + \frac{1}{\mathrm{Fr}}\rho_\epsilon^n\mathbf{g} + \frac{1}{\mathrm{We}}\kappa\delta_\epsilon\nabla\phi \tag{25}$$

The superscript, n denotes the quantities at n-th time level. In this equation (projector step), the diffusion terms are discretized by using the standard central difference scheme. However, the convection terms are discretized by means of a second order ENO scheme. In the second step of the method, to satisfy the incompressibility constraint, $\nabla.\mathbf{u}^{n+1} = 0$, the pressure at $(n+1)$-th time level, p^{n+1} is defined through the solution of Poisson equation:

$$\nabla^2 p^{n+1} = \frac{\rho_\epsilon^{n+1}}{\Delta t}\nabla.\mathbf{u}^* \tag{26}$$

Numerical solution of the pressure Poisson equation (26) is challenging due to the discontinuities in the variables near the interface. To compute pressure, an efficient immersed interface scheme is enforced, according to [26]. The detailed discussion of this scheme is presented in Section (3.1).

The final step of the method is to compute the velocity field at $(n+1)$-th time level, $\mathbf{u}^{n+1}$ from the following expression:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\Delta t}{\rho_\epsilon^{n+1}} \nabla p^{n+1} \tag{27}$$

The set of equations (25)-(27) comprise to form a N-S solver which are simulated along with the application of LS method to reproduce the evolving interfaces. Third order accurate **total variation diminishing Runge Kutta (TVDRK) scheme** is enforced to solve the LS advection equation (8) and the reinitialisation equation (9) in time. Spatial discretization in the LS method is carried out by using a **fifth order Hamilton Jacobi weighted essentially non-oscillatory scheme (HJ-WENO)** scheme.

3.1 Immersed Interface treatment

In order to discretize (26), the grid points in the computational domain are categorized into regular and irregular points [31, 32, 26]. A grid point is said to be an grid irregular point, if the finite difference stencil resulting from the approximation of the derivatives encloses points from both sides of Γ. Otherwise, they are categorized as regular. Our numerical algorithm is based upon enforcing a nine point **Higher Order Compact (HOC) scheme** [32, 33, 34, 35] for discretization at regular points and a higher order accurate immersed interface method at the irregular points. However, as we contemplate on using a nine point HOC scheme on the regular points, if all the eight neighbours of a regular point cannot be found on the same side across Γ, there is a need for redefining such regular points. Subsequently, we further classify a regular point as "semi-regular" (labelled in Fig. 2). A regular point in the subdomain Ω_i is termed as semi-regular if at least one of its eight neighbouring points in the nine point HOC stencil belongs to Ω_j such that $i \neq j, \quad i, j = 1, 2$. The methodology employed at irregular points curtails to standard central differences at semi-regular points. The detailed description of the methodology used for the discretization of (26) at irregular points is presented. The paramount feature of this methodology is the use of interfacial points as nodes in the stencil, which allows us to employ standard finite difference approximations directly. Pertinent interpolation techniques are enforced on both sides of the interface to determine the values at interfacial points. Fig. 2 sketches a magnified view of Ω in the vicinity of Γ. It outlines different classifications of grid points, along with a general interpolation stencil used for computations.

Let (x_i, y_j) be an irregular grid point (see Fig. 2). The discretization at irregular points follows the similar procedure in both x and y-directions. Therefore, the modified formulas are presented only in x-direction on the segment $[x_{i-(N-1)}, x_{i+M}] \times \{y_j\}$ and the subscript j in the notations below is omitted. Denote the interfacial point by "$x_{i+\theta}$" such that

$$x_{i+\theta} = x_i + \theta h, \qquad 0 < \theta < 1.$$

Denoting the approximation at point x_i by p_i, the central difference formula for the derivative

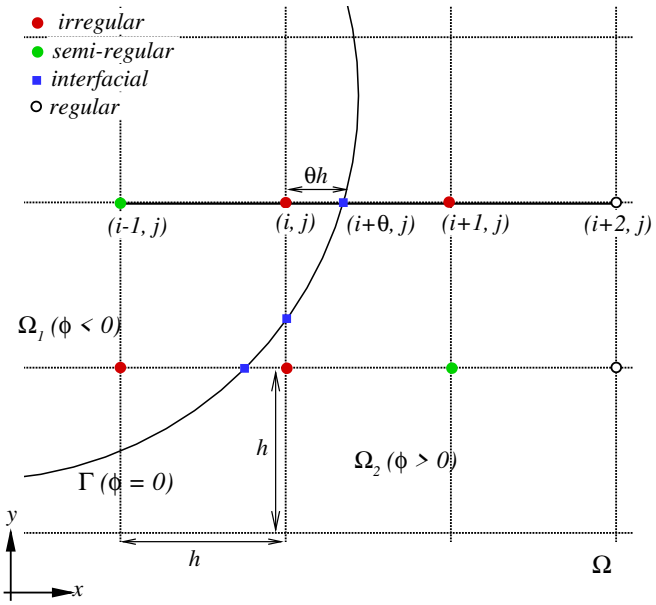


Figure 2: Illustration of the magnified view of computational domain, Ω in the vicinity of the interface, Γ. Different classifications of grid points are highlighted with coloured symbols and a general interpolation stencil used in our computations is outlined.

$\dfrac{\partial^2 p}{\partial x^2}$ enclosing grid points x_{i-1}, x_i and the interfacial point $x_{i+\theta}$ as nodes can be written as:

$$\left.\frac{\partial^2 p}{\partial x^2}\right|_i \approx \left(\frac{p^-_{i+\theta} - p_i}{x_{i+\theta} - x_i} - \frac{p_i - p_{i-1}}{x_i - x_{i-1}}\right) \Bigg/ \left(\frac{(x_{i+\theta} - x_i) + (x_i - x_{i-1})}{2}\right) \tag{28}$$

We now introduce the notation $p^-_{i+\theta}$ to denote the value of p at the interfacial point $x_{i+\theta}$ in Ω_1. Correspondingly, $p^+_{i+\theta}$ denotes the value of p at the interfacial point $x_{i+\theta}$ in Ω_2. $p^-_{i+\theta}$ is calculated by Lagrange polynomial interpolation by using points on both sides of the interface and the interfacial point. The interpolating polynomial on one side of the interface enclosing $(N+1)$ points $\{x_{i-(N-1)}, \cdots, x_{i-1}, x_i, x_{i+\theta}\}$ can be written as

$$L^-(x) = \sum_{k=-(N-1)}^{0} l_k(x) u_{i+k} + l_\theta(x) u^-_{i+\theta} \tag{29}$$

Similarly, the interpolating polynomial on the other side of interface enclosing $(M+1)$ points $\{x_{i+\theta}, x_{i+1}, x_{i+2}, \cdots, x_{i+M}\}$ can be written as

$$L^+(x) = \sum_{k=1}^{M} h_k(x) u_{i+k} + h_\theta(x) u^+_{i+\theta} \tag{30}$$

Here $l_k(x)$ and $h_k(x)$ are the Lagrange basis polynomials and they are respectively given by

$$l_k(x) = \prod_{\substack{-(N-1)\leqslant l\leqslant 0,\ l=\theta \\ l\neq k}} (x - x_{i+l}) \Bigg/ \prod_{\substack{-(N-1)\leqslant l\leqslant 0,\ l=\theta \\ l\neq k}} (x_{i+k} - x_{i+l})$$

and

$$h_k(x) = \prod_{\substack{l=\theta,\ 1\leqslant l\leqslant M \\ l\neq k}} (x - x_{i+l}) \Bigg/ \prod_{\substack{l=\theta,\ 1\leqslant l\leqslant M \\ l\neq k}} (x_{i+k} - x_{i+l})$$

Differentiating polynomial equations (29) and (30) term by term and substituting into derivative jump condition ($[p_x]_\Gamma$) at the interface yield

$$\left\{ \sum_{k=1}^{M} h'_k(x_{i+\theta}) p_{i+k} + h'_\theta(x_{i+\theta}) p^+_{i+\theta} \right\} - \left\{ \sum_{k=-(N-1)}^{0} l'_k(x_{i+\theta}) p_{i+k} + l'_\theta(x_{i+\theta}) p^-_{i+\theta} \right\} = [p_x]_\Gamma \,. \tag{31}$$

Note that derivative jump conditions are nothing but the derivative of equation (24) in the direction normal to the interface. Equation (31) is a discretized version obtained by resolving the normal jump along x-direction. It is easy to see that jump relation (24) translates to

$$p^+_{i+\theta} - p^-_{i+\theta} = [p]_\Gamma \,, \tag{32}$$

so that (31)-(32) simplifies to

$$\begin{bmatrix} h'_\theta(x_{i+\theta}) & -l'_\theta(x_{i+\theta}) \\ 1 & -1 \end{bmatrix} \begin{bmatrix} p^+_{i+\theta} \\ p^-_{i+\theta} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{33}$$

where

$$b_1 = [p_x]_\Gamma - \left\{ \sum_{k=1}^{M} h'_k(x_{i+\theta}) - \sum_{k=-(N-1)}^{0} l'_k(x_{i+\theta}) \right\} p_{i+k}$$

$$\text{and} \qquad b_2 = [p]_\Gamma \,.$$

Solution of (33) yields the following general formula for $p^-_{i+\theta}$

$$p^-_{i+\theta} = \frac{1}{D} \left\{ \sum_{k=-N+1}^{M} a_k p_{i+k} + [p_x]_\Gamma - h'_\theta(x_{i+\theta}) \, [p]_\Gamma \right\} \tag{34}$$

where

$$D = h'_\theta(x_{i+\theta}) - l'_\theta(x_{i+\theta})$$

$$a_k = \begin{cases} l'_k(x_{i+\theta}) & (k = -(N-1),\ \ldots\ , 0) \\ h'_k(x_{i+\theta}) & (k = 1\ , \ldots,\ M). \end{cases}$$

The same procedure is followed at the node x_{i+1}. Here, once again a central difference type discretization is employed as in equation (28) by enclosing the nodes $x_{i+\theta}$, x_{i+1} and x_{i+2}. In this case, $p^+_{i+\theta}$ needs to be approximated by the means Lagrange polynomial interpolation. At the boundary $\partial\Omega$, pressure is computed by solving the zero pressure gradient condition

$$\nabla p \cdot \hat{n} = 0, \tag{35}$$

$\hat{n}$ being a unit normal to the boundary. The values of p arising out of the Neumann boundary condition (35) are approximated by a fifth order one sided formula (see [36]). For example, on the right boundary (denoted by the index b), the following backward difference formula is enforced:

$$p_{b,j} = \frac{1}{25}\left[48p_{b-1,j} - 36p_{b-2,j} + 16p_{b-3,j} - 3p_{b-4,j}\right] + O(h^5),$$

and likewise on the other three boundaries.

3.2 Time step

The time step, Δt must adhere to the CFL condition at each iteration due to the presence of convective terms. Therefore, the following restriction is imposed [29]:

$$\Delta t \leqslant \min_{\Omega}\left(\frac{h}{||\mathbf{u}^n||}, \frac{h^2\rho^n}{\mu^n}, \left(\frac{\rho_1+\rho_2}{4\pi\sigma}\right)^{1/2} h^{3/2}\right) \tag{36}$$

3.3 Calculation Algorithm

This section summarizes how the solution algorithm proceeds iteratively to advance the solution of velocity field, pressure field, and level set function forward from time step t^n to t^{n+1}:

1. Calculate Δt (Section 3.2).
2. Compute $\mathbf{u}^*$ from $\mathbf{u}^n$ (Eq. (25)).
3. Solve the pressure Poisson equation (26) to calculate p (Section 3.1).
4. Compute $\mathbf{u}^{n+1}$ from $\mathbf{u}^*$ (Eq. (27)).
5. Solve the level set evolution equation (8) to advect ϕ by using $\mathbf{u}^{n+1}$.
6. Iterate the reinitialisation equation (9) to correct ϕ (Section 2.1).
7. Return to Step 1 till $t > t_{\text{final}}$.

4 Numerical Examples

In order to affirm the validity and efficiency of the proposed methodology, a series of numerical experiments are conducted on flows involving simple and intricate interface geometries. Precisely, the following problems are addressed : circular flow with a fixed interface geometry

to test the N-S solver; problem of the dispersion of capillary waves for validation and finally, transient evolution of initial circular, ellipse and star shaped bubbles to equilibrium static shapes. All the computations are performed on a Laptop with i5-2430M Processor with 4 GB RAM using a double precision floating point arithmetic on grid sizes ranging from 20×20 to 320×320. The performance of our methodology is evaluated in terms of the infinity norm of the errors in velocity, $\|\mathbf{u}\|_\infty$ and pressure, $\|\mathbf{p}\|_\infty$ such that $\|\mathbf{u}\|_\infty = \left(\dfrac{\|u\|_\infty + \|v\|_\infty}{2}\right)$.

4.1 Circular Flow

In order to ascertain the accuracy of the proposed methodology to solve N-S equations, an example with a fixed circular interface is chosen [37], which admits the exact solution

$$u(x,y,t) = \begin{cases} (1-e^t)\left(\dfrac{y}{r} - 2y\right), & \text{if} \quad \phi > 0 \\ 0, & \text{if} \quad \phi \leqslant 0 \end{cases},$$

$$v(x,y,t) = \begin{cases} (1-e^t)\left(\dfrac{-x}{r} + 2x\right), & \text{if} \quad \phi > 0 \\ 0, & \text{if} \quad \phi \leqslant 0 \end{cases},$$

$$p(x,y,t) = \begin{cases} \sin(\pi x)\sin(\pi y), & \text{if} \quad \phi > 0 \\ 0, & \text{if} \quad \phi \leqslant 0 \end{cases}.$$

Here $r = \sqrt{x^2+y^2}$ and the interface is a circle centered at origin with radius 0.5 in a square computational domain of size $[-1,1]\times[-1,1]$ with an equal grid spacing, h. Dirichlet boundary conditions are used at the outer boundaries.

Fig. 3 plots the variation of absolute error in the curvature along the circumferential angle, 0° to 360° of the interface computed by using the formula given in equation (11). It is intended to verify the accurate calculation of curvature by considering three different grids: $32 \times 32, 64 \times 64$ and 128×128. Minimal error of order ranging from $O(10^{-6})$ to $O(10^{-3})$ is reported. It is noteworthy to mention that all the numerical results presented hereafter are simulated by using the numerical approximation to the curvature in equation (11). In order to affirm the adequateness of different grid sizes tested, Table 1 shows errors in velocity, $\|\mathbf{u}\|_\infty$ and pressure, $\|p\|_\infty$ for $\mu_2 = 1$, $\mu_1 = 0.1$ at $t = 2$ and compares them with the results of method proposed in [37]. It can be seen that our computed results are relatively better which certainly confirms the efficiency of the proposed methodology.

4.2 Dispersion of Capillary waves

The preceding test example permits us to predict the accurateness in the calculation of curvature and the performance of N-S solver. In the present example of capillary waves ([38, 39, 40, 41]), the performance of the full methodology, i.e. coupling amongst the curvature calculation, N-S solver and interface advection is validated. The problem of small amplitude oscillations associated with an initially sinusoidal interface between two superimposed

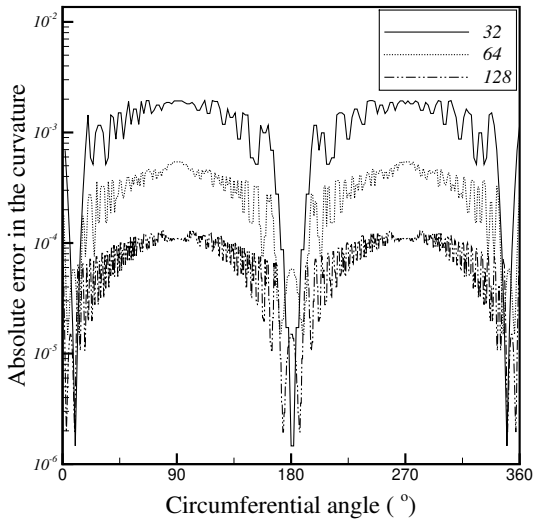


Figure 3: Absolute error in the computed curvature along the circumferential angle, 0° to 360° of the interface for three different grids, $32 \times 32, 64 \times 64$ and 128×128.

Table 1: Comparison of the errors in the dimensionless velocity, $\|\mathbf{u}\|_\infty$ and pressure, $\|p\|_\infty$ on grid refinement with the method proposed in [37].

n	$\|\mathbf{u}\|_\infty$		$\|p\|_\infty$	
	Present	[37]	Present	[37]
32	1.224×10^{-3}	1.915×10^{-3}	6.760×10^{-3}	7.516×10^{-3}
64	3.286×10^{-4}	4.498×10^{-4}	1.502×10^{-3}	2.110×10^{-3}
128	8.882×10^{-5}	1.101×10^{-4}	4.016×10^{-4}	6.103×10^{-4}
256	1.211×10^{-5}	2.811×10^{-5}	9.616×10^{-5}	1.681×10^{-4}

viscous, incompressible fluids of infinite depth and lateral extent is addressed. A sinusoidal perturbation is induced to a horizontal interface between two fluids initially at rest. Fig. 4 sketches the interface configurations at initial and equilibrium states in a rectangular domain of length, L and height, H. The balance amongst surface tension, inertia and viscosity leads to the oscillations of the interface around its equilibrium position. Numerical simulations are performed on a equally spaced grid of size 64×190 on $\Omega = [-1.5, 0.5] \times [-1.5, 1.5]$ with a time step; $\Delta t = 1 \times 10^{-4}$. The initial wavelength of the wave is chosen as $\lambda = 1$ with an initial amplitude, $a_0 = 0.1$. The kinematic viscosites of both the fluids are assumed to be the same. Gravitational forces are assumed to be absent and flow regime is set by treating the horizontal walls as no slip boundaries and vertical walls as periodic boundaries. The following non dimensional parameters are adopted: $\rho_1/\rho_2 = 100, 1000$; $Re = 100$, $We = 1$. Fig. 5 sketches the transient variation of interface curves in terms of the time evolution of their respective normalized amplitudes, computed on 64×190 grid. The decay of oscillations

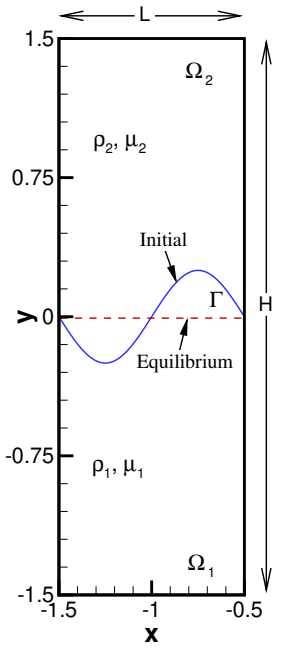


Figure 4: Illustration of interface configurations at initial and equilibrium states in computational domain.

is apparent from the clear fluctuations of amplitude curves in this figure. The numerical solution agrees well with the analytical solution of Prosperetti [40], which certainly confirms the accurateness and efficiency of the proposed methodology.

4.3 Static Bubble

The present example addresses the problem of a stationary circular bubble in static equilibrium without gravity ([14, 15, 18, 22, 23]). In the absence of viscous, gravitational or external forces, the circular interface with surface tension should remain motionless with a pressure jump at the interface. This implies an exact equilibrium between the surface tension force and pressure gradient. Therefore, the analytical solution is a zero velocity field and a constant pressure jump at the interface, according to Young-Laplace condition, expressed as:

$$[p]_{\text{exact}} = \sigma \kappa_{\text{exact}}. \tag{37}$$

Here, the exact curvature is given by $\kappa_{\text{exact}} = 1/r$ such that r is the radius of the bubble. The pressure field arises from a constant value $p_{\text{out}} = p_0$ outside the bubble to a value $p_{\text{in}} = p_0 + \sigma\kappa_{\text{exact}}$ inside the bubble. The pressure inside the bubble, p_{in} corresponds to the maximum pressure in Ω, and the pressure outside the bubble, p_{out} corresponds to the minimum pressure in Ω.

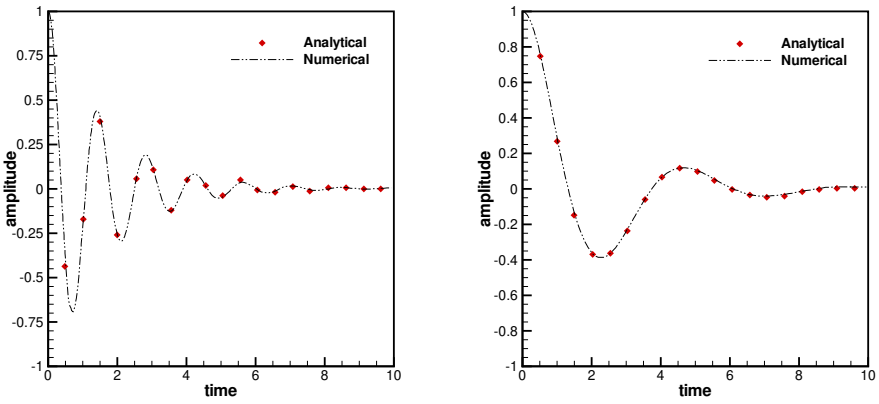


Figure 5: Comparison of the transient evolution of the normalized amplitudes of capillary waves for (a) $\rho_1/\rho_2 = 10$, and (b) $\rho_1/\rho_2 = 100$; with analytical solution of Prosperetti [40], plotted on 64×190 grid.

A bubble with radius $r = 0.5$ is positioned in the center of a square domain, Ω of size $[-1, 1] \times [-1, 1]$. No-slip conditions are imposed everywhere on the boundary of Ω. The coefficients of surface tension, the density and the viscosity inside and outside the bubble are all set to unity. This corresponds to a Weber number of $We = 1$ and Reynolds number of $Re = 1$. However, circulating flow near the interface, known as spurious velocity currents arise because at a discretized level, accurate calculation of curvature and the balance between interfacial tension and pressure jump are not trivial problems[42]. In the computational results that follow, these spurious currents are measured by errors in the dimensionless velocity, $\|\mathbf{u}\|_\infty$.

Initially, in Fig. 6, we probe the influence of interface thickness, ϵ on the computed results in order to check the numerical diffusion in the transition zone, $|\phi| \leqslant \epsilon$. A careless choice of ϵ could lead to an inappropriate shift in the zone of surface tension force [15]. Fig. 6 sketches the plots of errors in the dimensionless velocity for different values of ϵ, computed on a 80×80 grid, at time, $t = 10$. The spurious currents reduce to $O(10^{-6})$ when $2.5h \leqslant \epsilon \leqslant 3.5h$ and they rise thereafter. This suggests that the interface thickness has an influence on spurious currents, which are relatively strong when $\epsilon \geq 3.5h$ and as such, $\epsilon = 2.5h$ is chosen in our computations. Fig. 7 sketches the pressure distribution on the whole domain, Ω and absolute error between the numerical pressure and exact pressure, computed on 80×80 grid. As expected, a piecewise constant pressure distribution in inner and outer domains is observed. No oscillation in the pressure profile resulting from this computation is seen near the interface as against some reported in the literature [14, 22]. Fig. 7(b) reports maximum error near the interface which is attributed to the numerical approximation of pressure due to jump discontinuity at the interface. Table 2 further consolidates how well the computed pressure fulfils Young-Laplace's law as reflected by the infinity norm, $\|p\|_\infty$. One can clearly see that, for both pressure and velocity, the norm decreases rapidly with

grid refinement, to converge with nearly second order accuracy. Fig. 8 shows a typical distribution of spurious velocity currents around the interface, computed on a 80×80 grid. Remarkably, the magnitude is $(O(10^{-7}))$ which is lower than those reported by the methods proposed in [22].

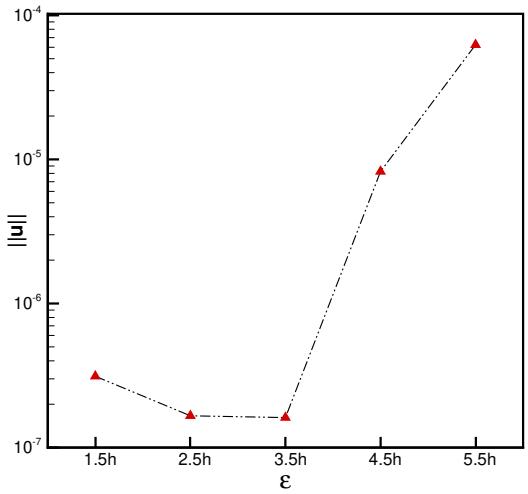


Figure 6: The effect of interface thickness, ϵ on the infinity norm of the error in velocity, $\|\mathbf{u}\|_\infty$, computed on 80×80 grid.

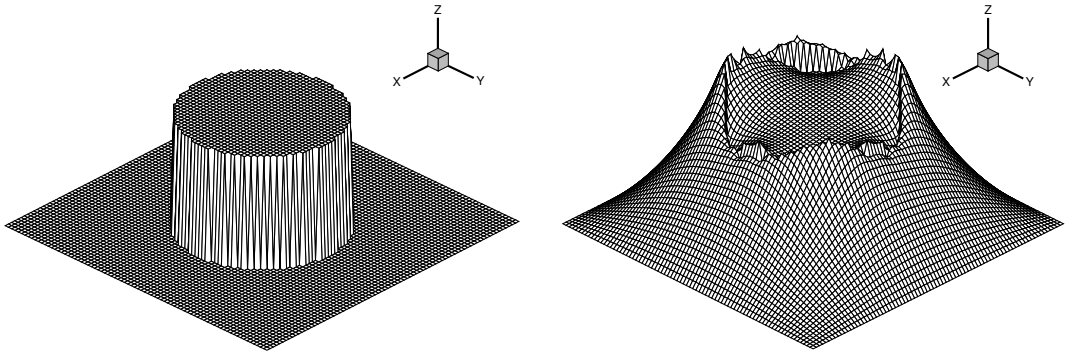


Figure 7: (a) Pressure distribution and (b) Absolute error between the numerical pressure and exact pressure on the whole domain, Ω computed on 80×80 grid.

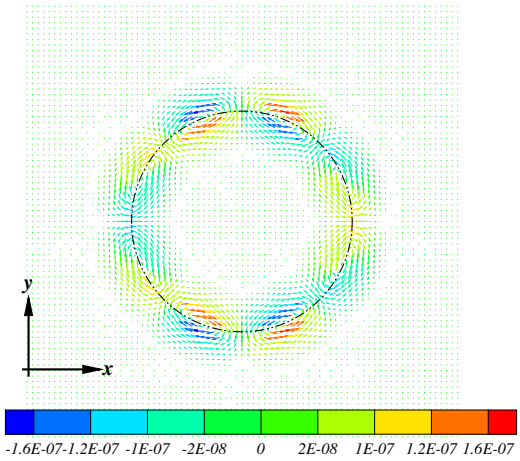


Figure 8: Spurious velocity currents around the bubble interface in terms of velocity vectors, plotted on 80×80 grid.

Table 2: Errors in the dimensionless velocity, $\|\mathbf{u}\|_\infty$ and pressure, $\|p\|_\infty$ with their respective orders of accuracies with grid refinement.

n	$\|\mathbf{u}\|_\infty$	order	$\|p\|_\infty$	order
20	1.976×10^{-6}		2.471×10^{-3}	
		1.76		1.89
40	5.814×10^{-7}		6.680×10^{-3}	
		1.85		1.93
80	1.615×10^{-7}		1.758×10^{-3}	

4.4 Oscillating Bubble

In the present section, the problem of an initially non-spherical viscous bubble with subsequent oscillating decay to equilibrium circular shape is addressed ([3, 17, 20, 21, 23, 24, 37, 43]). The transient motion and evolution of the bubble till numerical equilibrium is analyzed in order to evaluate the performance of proposed methodology. Two different initial interface configurations: ellipse and star shaped are considered.

Ellipse shaped interface

First, an initial ellipse shaped bubble with semi-major axis and semi-minor axis, $r_x = 0.75$ and $r_y = 0.5$ respectively, placed in the center of Ω is considered. The interface configurations at initial and equilibrium states are sketched in Fig. 9. The bubble tends to acquire a circular shape at equilibrium with radius $r_e \approx 0.61237$ (labeled as equilibrium in Fig. 9). We begin with the detailed discussion of the performance of our scheme using following non-dimensional parameters: $\rho_1 = \rho_2 = 1$, $\mu_1 = \mu_2 = 0.1$, $Re = 10$ and $We = 10$. Then, the

performance is evaluated by reducing the value of We to 1. Firstly, a grid refinement analysis

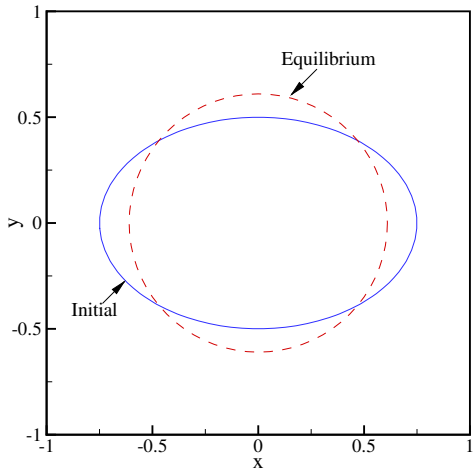


Figure 9: Illustration of interface configurations at initial and equilibrium states in computational domain.

is performed to analyze the order of accuracy with the results of finest grid, 320×320 chosen as reference value. Table 3 compiles the grid refinement analysis at $t = 0$, where the errors are measured in infi norm. One can easily see both velocity and pressure errors decaying at a rate of nearly $O(h^2)$. Fig. 10 shows the transient oscillations of the bubble and their profiles in terms of velocity vectors at times $t = 0.5, 2.5, 5.0, 7.0, 10.0$ and 15.0 for $Re = 10$ and $We = 10$. The maximum velocity magnitudes first increase till time $t = 2.5$ up to $O(10^{-2})$ and then decrease up to $O(10^{-4})$ till $t = 15$. Fig. 11(a) sketches the transient evolution of pressure along the bubble centerline at initial and equilibrium states, computed on 160×160 grid. As expected a piece-wise constant pressure distribution is observed in the inner and outer domains at the equilibrium circular shape. The pressure distribution over the whole domain is illustrated in Fig. 11(b). The pressure profile clearly exhibits no oscillations near the interface as against reported by few [14, 18]. The numerical equilibrium is found to agree well with the true equilibrium using the proposed methodology. For instance, at t = 15, the error between r_x and r_e is only $\mathbf{3.12 \times 10^{-4}}$ and the error between r_y and r_e is only $\mathbf{2.30 \times 10^{-4}}$. Further, the absolute error in area is $\mathbf{5.33 \times 10^{-4}}$ which indicates fairly little leakage of about 0.01%.

The transient variation of bubble shapes is further evaluated in terms of the time evolution of r_x and r_y. The evolution of r_x and r_y is plotted in Fig. 12 for different values of We in order to analyze the effect of surface tension. The decay of oscillations is apparent from the clear fluctuating pattern of r_x and r_y in these figures. Further, it can be noticed that increase in the value of We reduces the motion of the bubble, which is reflected in fewer number of oscillations with smaller amplitudes and longer time to converge to equilibrium. This observation is further supported by the transient evolution of total kinetic energy, K.E. $= \frac{1}{2}\int \rho \mathbf{u}.\mathbf{u} dv$, sketched in Fig. 13. All these observations certainly confirm the

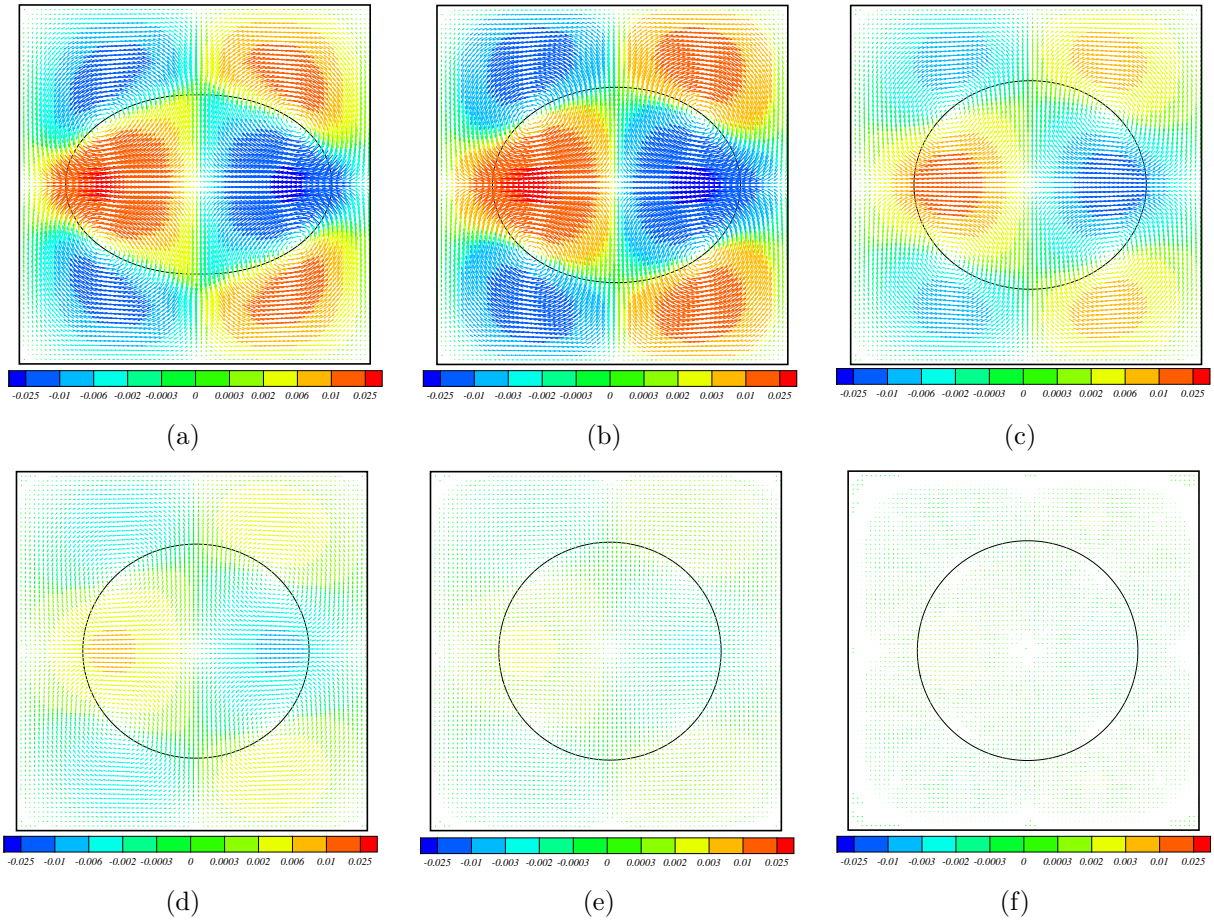


Figure 10: Transient oscillations of the bubble and their velocity profiles at times $t = (a)0.5, (b)2.5, (c)5.0, (d)7.0, (e)10.0$ and (f) 15.0 for $\rho_1/\rho_2 = 1$, $\mu_1/\mu_2 = 1$, $Re = 10.0$ and $We = 10.0$.

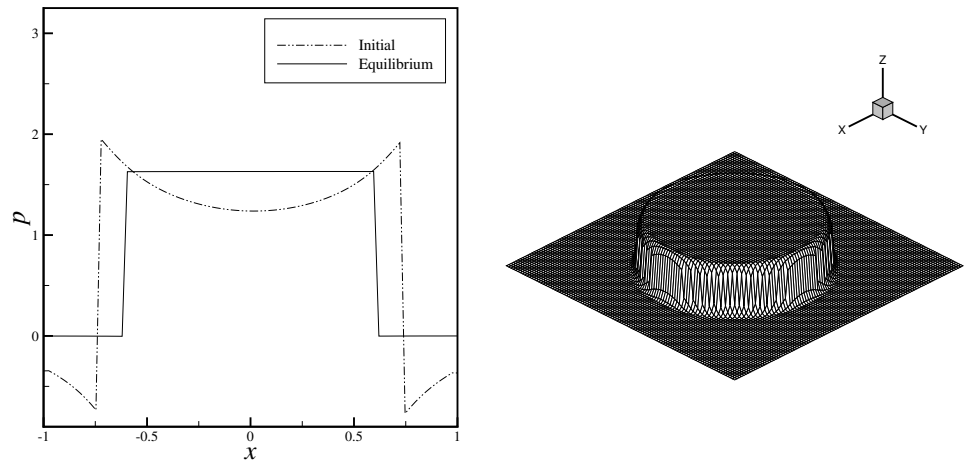
accurateness and capability of the proposed methodology in simulating surface tension driven interfacial flows.

Star shaped interface

In the final example, we illustrate the usefulness of proposed methodology to model surface tension effects in the vicinity of high curvature, complicated interface (star-shaped) geometry in the benchmark test. Fig. 14 sketches the interface configurations at initial and equilibrium states where the initial shape is defined in cylindrical coordinates (r, θ) as:

$$r(\theta) = r_\circ(1 + \epsilon \sin(\zeta\theta)), \qquad 0 \leqslant \theta \leqslant 2\pi.$$

The bubble tends to converge to an equilibrium circular shape due to the effects of surface tension.



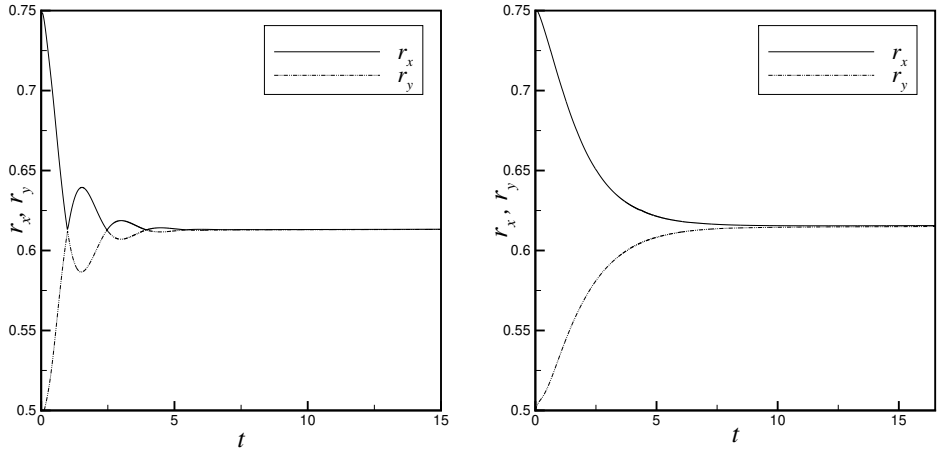
(a) (b)

Figure 11: (a)Transient evolution of pressure along the bubble centerline at initial and equilibrium states and (b) the distribution of numerical pressure over the whole domain, Ω at equilibrium for $\rho_1/\rho_2 = 1$, $\mu_1/\mu_2 = 1$, $Re = 10$ and $We = 10$; computed on 80×80 grid.

Table 3: Errors in the dimensionless velocity, $\|\mathbf{u}\|_\infty$ and pressure, $\|p\|_\infty$ with their respective orders of accuracies on grid refinement at time $t = 0$.

n	$\|\mathbf{u}\|_\infty$	order	$\|p\|_\infty$	order
40	1.291×10^{-2}		2.237×10^{-2}	
		1.96		1.76
80	3.322×10^{-3}		6.580×10^{-3}	
		2.01		1.84
160	8.263×10^{-4}		1.827×10^{-3}	

Flow regime is set by adopting the following non-dimensional parameters: $\rho_1 = \rho_2 = 1$, $\mu_1 = \mu_2 = 1$, $Re = 1$ and $We = 10$. Fig. 15 sketches transient oscillations of the bubble shapes and their velocity profiles at times $t = 0.5, 2, 7, 10, 14$. The maximum velocity magnitudes first rise till time $t = 5$ up to $O(10^{-1})$ and then fall up to $O(10^{-4})$ till $t = 16$. Fig. 11(a) sketches the transient evolution of pressure along the bubble centerline at initial and equilibrium states, computed on 160×160 grid. As expected a piece-wise constant pressure distribution is observed in the inner and outer domains at the equilibrium circular shape. The pressure distribution over the whole domain is illustrated in Fig. 11(b). The pressure profile clearly exhibits negligible oscillations near the interface which proves the capability of our scheme in efficiently resolving the pressure jump in the vicinity of complex, high curvature interfaces as well. Table 4 shows grid refinement analysis of error infinity norm of our computed results. The infinity norm of error in the solution is small and the accuracy does not seem to be affected by arbitrary shaped interface as long as grid resolution is high enough to allow for one sided interpolation. Moreover, one can easily see convergence



(a) (b)

Figure 12: Transient evolution of the semi-major axis, r_x and semi-minor axis, r_y for (a) $We = 1$ and (b)$We = 10$ when $\rho_2/\rho_1 = 1$, $\mu_2/\mu_1 = 1$ and $Re = 10$, computed on 160×160 grid.

rate near to second order for both the parameters.

Table 4: Errors in the dimensionless velocity, $\|\mathbf{u}\|_\infty$ and pressure, $\|p\|_\infty$ with their respective orders of accuracies with grid refinement at time $t = 0$.

n	$\|\mathbf{u}\|_\infty$	order	$\|p\|_\infty$	order
40	1.486×10^{-2}		3.421×10^{-2}	
		1.89		1.95
80	4.026×10^{-3}		8.832×10^{-3}	
		2.03		1.98
160	9.824×10^{-4}		2.226×10^{-3}	

4.5 Deforming drops

In this subsection, the problem of a liquid drop implanted in another immiscible liquid subjected a to shear flow is addressed. Fig. 17 shows an illustration of square computational domain, containing a liquid drop, where upper and bottom walls move with opposite velocities thereby inducing a plane Couette flow. It is admitted from the literature that, in such flow configurations, the drop reciprocates by deforming and by developing an internal circulation in the direction of the flow [44, 45, 46]. In present computations, the radius of the drop is set to one fourth of the domain height, $r = 0.25H$ where $L/H = 1$. The physical properties of the fluids and surface tension, σ of the drop interface are assumed to be uniform and constant during the motion.

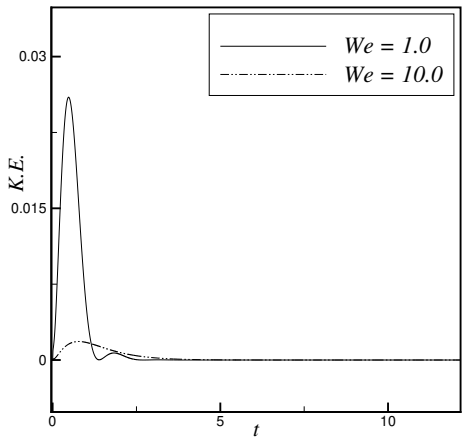


Figure 13: Transient evolution of total Kinetic energy for $We = 1$ and $We = 10$ when $\rho_2/\rho_1 = 1$, $\mu_2/\mu_1 = 1$, $Re = 10.0$, computed on 160×160 grid.

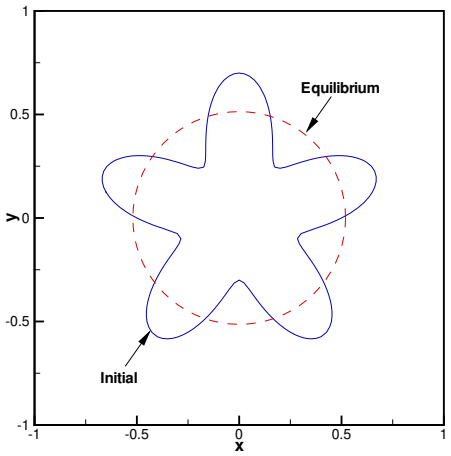


Figure 14: Illustration of interface configurations at initial and equilibrium states in computational domain.

Flow regime is set by treating the vertical walls as periodic boundaries and by adopting the following non dimensional parameters: $Re = 100$ and $Ca = 0.1, 0.2, 0.3, 0.4$. The Capillary number, $Ca = We/Re$, represents the relative magnitude of viscous and capillary forces along the interface. The density and viscosity ratios are assumed to be unity. The transient variation of drop shapes is evaluated in terms of deformation parameter, D_{Df} expressed as

$$D_{Df} = \frac{r_x - r_y}{r_x + r_y}$$

The transient evolution of D_{Df} is plotted in Fig. 18(a) for different values of Ca in order to analyse the effect of surface tension. A first inspection reveals that increase in the value of Ca increases the magnitude of D_{Df}. Further, it is noticed that, when $Ca = 0.1$, D_{Df}

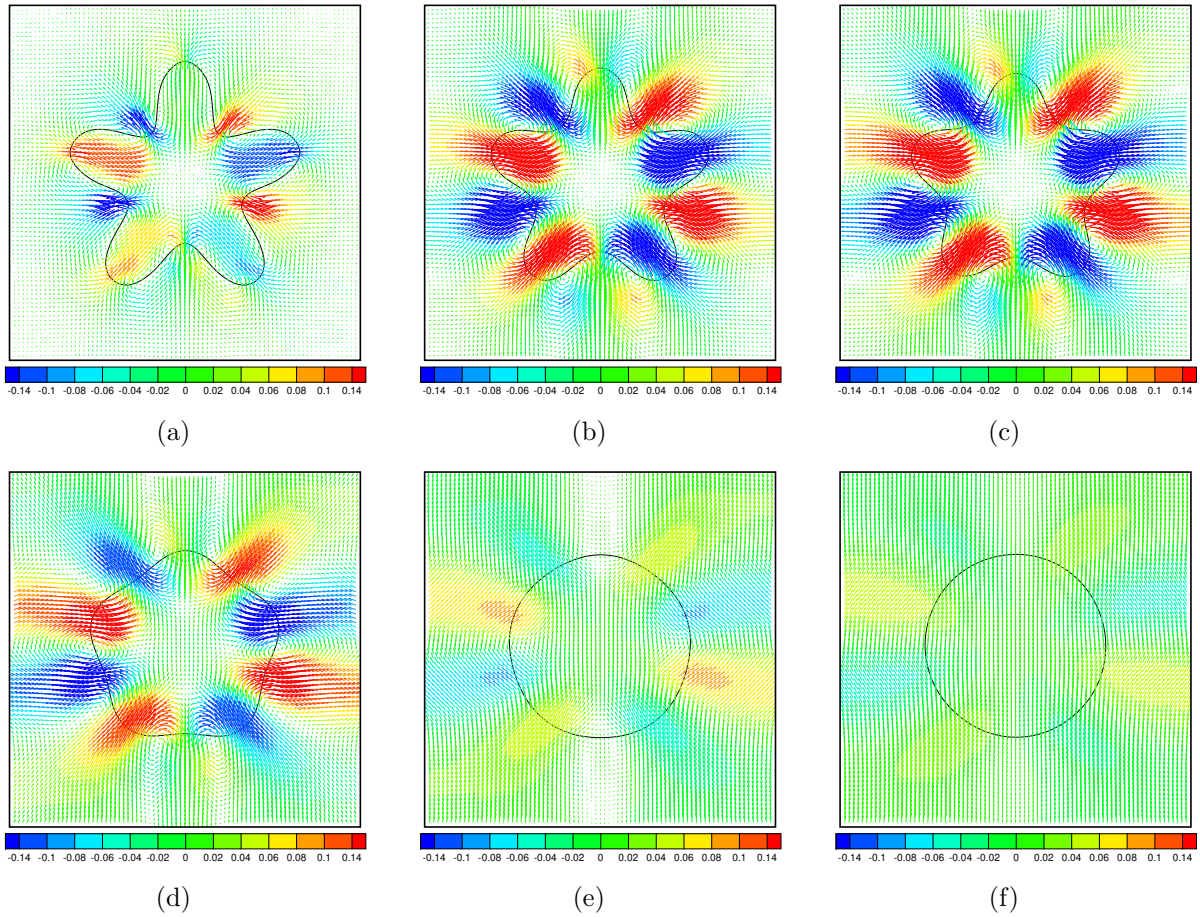
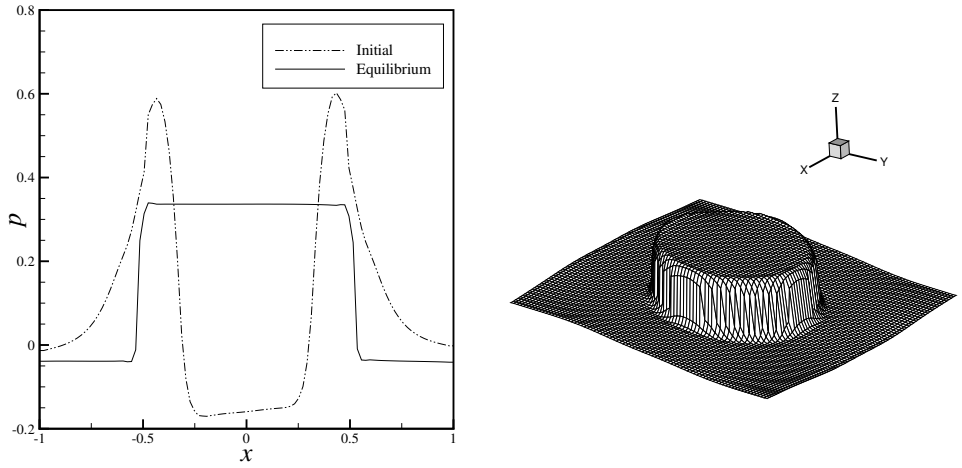


Figure 15: Transient oscillations of the bubble and their velocity profiles at times $t = 0.5, 2, 7, 10, 14$ and 16 for $\rho_1 = \rho_2 = 1$, $\mu_1 = \mu_2 = 1$, $Re = 1$ and $We = 20$.

tends to a constant value near time $t = 2$ which signifies that the drop tends to acquire a steady shape. However, when $Ca > 0.1$, D_{Df} continues to increase which signifies that the drop continues to elongate and does not acquire a stationary shape. Fig. 18(b) sketches the contours of drop shapes at time $t = 3$ for different values of Ca. When $Ca = 0.1$, the shape of the drop resembles an ellipse. However, for high values of $Ca(Ca > 0.1)$, the drops are exceptionally deformed and they display a significant deviation from a steady elliptic shape. Delibrately, the minor axis shortens and the major axis lengthens and consequently they tend to acquire a sigmoidal shape while rotating in clockwise direction. In order to investigate the structure of the flow inside and outside the drop, **streamline patterns** are sketched in Fig. 19 for $Ca = 0.1$ and 0.4. The following Poisson equation for stream-function is solved to serve the purpose:

$$\Delta\psi = u_y - v_x.$$

When $Ca = 0.1$, the drop interface is tangential to the local streamlines affirming that the drops have attained steady shapes. The drops are wrapped by dividing streamlines that end



(a) (b)

Figure 16: (a) Time evolution of pressure along the centerline at times $t = 0.5, 2.5, 5.0, 7.0, 10.0$ and 15.0, and (b) the distribution of numerical pressure over the whole domain, Ω at $t = 15$ for $\rho_1/\rho_2 = 1$, $\mu_1/\mu_2 = 1$, $Re = 10$ and $We = 10$, computed on 160×160 grid.

at stagnation points positioned intermediary between drops and nearby the x-axis. However, when $Ca = 0.4$, the streamlines cross the drop interface demonstrating that drops continue to elongate and manage to rotate in clockwise direction due to the effect of the vorticity of the incoming flow. Next, in order to understand the inertial effects, Fig. 20 sketches the contours of drop shapes at time $t = 2$ for $Ca = 0.2$ and different values of $Re = 10, 50, 80$ and 100. It can be seen that inertial flow plays a significant role in deciding equivalent drop shapes and drops turn out to elongate more with increasing Re. This points to the fact that an increase in Re leads to an easier deformation. All these obervations are coherent with previous studies [45, 46] which certainly confirms that the proposed methodology can strongly capture primary traits of drops deformating in shear flows.

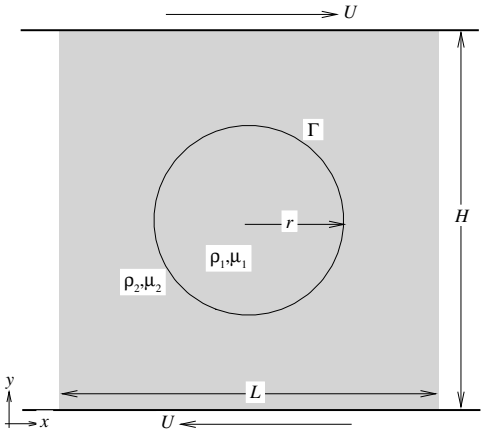
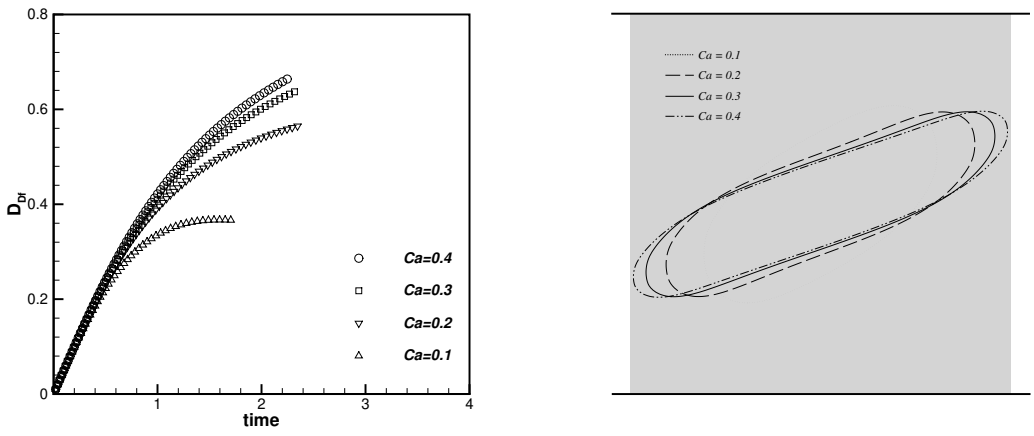


Figure 17: Illustration of a square computational domain of length, L, and height, H, containing a liquid drop of radius, r, such that both upper and bottom walls of the domain move with opposite velocities, $\pm U$.

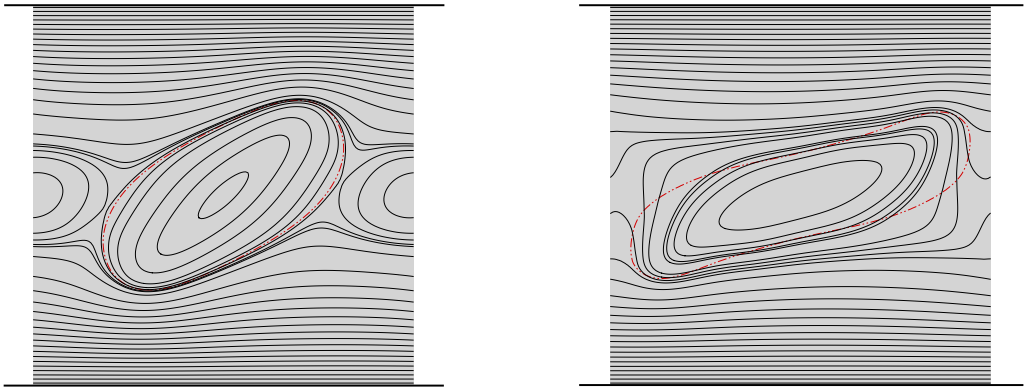


(a) (b)

Figure 18: (a) The transient evolution of drop shapes in terms of deformation parameter, D_{Df} and (b) the contours of drop shapes at time $t = 3$; for different values of Ca and $Re = 100$, computed on 180×180 grid.

5 Conclusion

In this paper, a coupling between a novel immersed interface method and level set method is proposed to simulate surface tension driven interfacial flows. The proposed methodology highly simplifies the grid generation process as it permits us to use a uniform, fixed Cartesian grid. First, an example with a fixed circular interface is presented to predict the accurateness



(a) (b)

Figure 19: **Streamline patterns** for (a) $Ca = 0.1$ and (b) $Ca = 0.4$; $Re = 100$, computed on 180×180 grid.

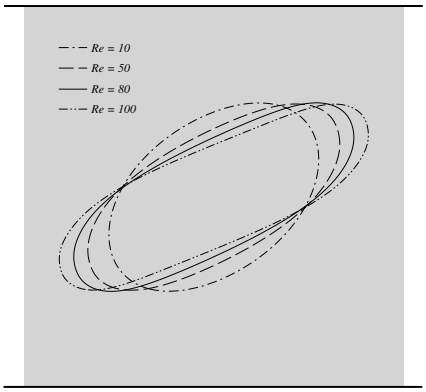


Figure 20: The contours of drop shapes at time $t = 2$ for $Ca = 0.2$ and different values of Re, computed on 180×180 grid.

in the calculation of curvature and the performance of N-S solver. Next, the algorithm is validated by the comparing the results with analytical solution by simulating the problem of small amplitude oscillations associated with the dispersion of capillary waves. Finally, the problems of initially circular, ellipse shaped and star shaped bubbles with an oscillating decay to a numerical equilibrium; and circular drops deforming in shear flows are addressed. The transient evolution of bubbles/drops in terms of their shapes, pressure profiles, velocity vectors, deformation ratios of major and minor axis is analyzed to observe the effect of surface tension. The methodology is shown to regain the exact numerical equilibrium between

the surface tension and pressure gradient in the vicinity of high curvature interface geometries as well. It demonstrates accurate resolution of pressure nullyfying the presence of any oscillations in the vicinity of the interface and Young-Laplace's law is validated to a good level of accuracy and with a significantly reduced spurious currents. Grid refinement analysis of the infinity error norms of pressure and velocities report the convergence rates near to two for the test cases. Moreover, primary traits of physical behaviour of drop deformations in shear flows are strongly captured by the proposed methodology. Numerical results show that the proposed methodology yields a good level of agreement with the reference data. It is noteworthy to mention that the immersed interface method employed in the present computations has already been validated for a fixed interface in three dimensions [26]. This opens up a possibility of their extension to address the problems of moving interfaces in three dimensions, which is the objective of our next paper.

References

[1] Greenspan, H.P., 1977. On the deformation of a viscous droplet caused by variable surface tension. Studies in Applied Mathematics, 57(1), pp.45-58.

[2] Greenspan, H.P., 1977. On the dynamics of cell cleavage. Journal of theoretical biology, 65(1), pp.79-99.

[3] Li, Z. and Lubkin, S.R., 2001. Numerical analysis of interfacial two–dimensional Stokes flow with discontinuous viscosity and variable surface tension. International Journal for Numerical Methods in Fluids, 37(5), pp.525-540.

[4] Stone, H.A., Stroock, A.D. and Ajdari, A., 2004. Engineering flows in small devices: microfluidics toward a lab-on-a-chip. Annu. Rev. Fluid Mech., 36, pp.381-411.

[5] Unverdi, S.O. and Tryggvason, G., 1992. A front-tracking method for viscous, incompressible, multi-fluid flows. Journal of computational physics, 100(1), pp.25-37.

[6] Chang, Y.C., Hou, T.Y., Merriman, B. and Osher, S., 1996. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows. Journal of computational Physics, 124(2), pp.449-464.

[7] Osher, S. and Sethian, J.A., 1988. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. Journal of computational physics, 79(1), pp.12-49.

[8] Klein, M., Ketterl, S. and Hasslberger, J., 2019. Large eddy simulation of multiphase flows using the volume of fluid method: Part 1Governing equations and a priori analysis. Experimental and Computational Multiphase Flow, 1(2), pp.130-144.

[9] Ketterl, S., Reimann, M. and Klein, M., 2019. Large eddy simulation of multiphase flows using the volume of fluid method: Part 2A-posteriori analysis of liquid jet atomization. Experimental and Computational Multiphase Flow, 1(3), pp.201-211.

[10] Scardovelli, R. and Zaleski, S., 1999. Direct numerical simulation of free-surface and interfacial flow. Annual review of fluid mechanics, 31(1), pp.567-603.

[11] Raessi, M., Mostaghimi, J. and Bussmann, M., 2010. A volume-of-fluid interfacial flow solver with advected normals. Computers & Fluids, 39(8), pp.1401-1410.

[12] Osher, S. and Fedkiw, R., 2006. Level set methods and dynamic implicit surfaces (Vol. 153). Springer Science & Business Media.

[13] Chorin, A.J., 1967. A numerical method for solving incompressible viscous flow problems. Journal of computational physics 2(1), pp. 12-26.

[14] Balcazar, N., Lehmkuhl, O., Jofre, L., Rigola, J. and Oliva, A., 2016. A coupled volume-of-fluid/level-set method for simulation of two-phase flows on unstructured meshes. Computers & Fluids, 124, pp.12-29.

[15] Chakraborty, I., Biswas, G. and Ghoshdastidar, P.S., 2013. A coupled level-set and volume-of-fluid method for the buoyant rise of gas bubbles in liquids. International Journal of Heat and Mass Transfer, 58(1), pp.240-259.

[16] Francois, M. and Shyy, W., 2003. Computations of drop dynamics with the immersed boundary method, part 1: numerical algorithm and buoyancy-induced effect. Numerical Heat Transfer: Part B: Fundamentals, 44(2), pp.101-118.

[17] Francois, M.M., Cummins, S.J., Dendy, E.D., Kothe, D.B., Sicilian, J.M. and Williams, M.W., 2006. A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. Journal of Computational Physics, 213(1), pp.141–173.

[18] Hysing, S., 2012. Mixed element FEM level set method for numerical simulation of immiscible fluids. Journal of Computational Physics, 231(6), pp.2449-2465.

[19] Huang, J. J., Haibo H., and Wang S.L., 2015. Phase-field-based simulation of axisymmetric binary fluids by using vorticity-streamfunction formulation. Progress in Computational Fluid Dynamics, an International Journal, 15(6) pp.352–371.

[20] Li, Z. and Lai, M.C., 2001. The immersed interface method for the Navier–Stokes equations with singular forces. Journal of Computational Physics, 171(2), pp.822–842.

[21] Leveque, R.J. and Li, Z., 1997. Immersed interface methods for Stokes flow with elastic boundaries or surface tension. SIAM Journal on Scientific Computing, 18(3), pp.709–735.

[22] Mier–Torecillam M.D., Idelsohn, S.R. and Oate, E., 2011. Advances in the simulation of multi-fluid flows with the particle finite element method. Application to bubble dynamics. International journal for numerical methods in fluids, 67(11), pp.1516–1539.

[23] Tong, A.Y. and Wang, Z., 2007. A numerical method for capillarity-dominant free surface flows. Journal of Computational Physics, 221(2), pp.506-523.

[24] Uh, M. and Xu, S., 2012. The immersed interface method for two-fluid problems.

[25] Balcazar Arciniega, N.V., 2014. Numerical simulation of multiphase flows: level-set techniques.

[26] Mittal, H.V.R. and Ray, R.K., 2018. Solving Immersed Interface Problems Using a New Interfacial Points-Based Finite Difference Approach. SIAM Journal on Scientific Computing, 40(3), pp.A1860-A1883.

[27] Yuan, H.Z., Shu, C., Wang, Y. and Shu, S., 2018. A simple mass-conserved level set method for simulation of multiphase flows. Physics of Fluids, 30(4), p.040908.

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

[29] Gutirrez, E., Favre, F., Balczar, N., Amani, A. and Rigola, J., 2018. Numerical approach to study bubbles and drops evolving through complex geometries by using a level setMoving meshImmersed boundary method. Chemical Engineering Journal, 349, pp.662-682.

[30] Peskin, C.S., 1977. Numerical analysis of blood flow in the heart. Journal of computational physics, 25(3), pp.220-252.

[31] Mittal, H.V.R., 2016. A class of higher order accurate schemes for fluid interface problems (Doctoral dissertation, Ph. D. thesis, Indian Institute of Technology, Mandi, India).

[32] Mittal, H.V.R., Kalita, J.C. and Ray, R.K., 2016. A class of finite difference schemes for interface problems with an HOC approach. International Journal for Numerical Methods in Fluids, 82(9), pp.567-606.

[33] Mittal, H.V.R., Ray, R.K. and Al-Mdallal, Q.M., 2017. A numerical study of initial flow past an impulsively started rotationally oscillating circular cylinder using a transformation-free HOC scheme. Physics of Fluids, 29(9), pp. 093603.

[34] Mittal, H.V.R., Al-Mdallal, Q.M. and Ray, R.K. 2017. Vortex shedding modes from a rotationally oscillating circular cylinder. Ocean Engineering, 146, pp. 324–338.

[35] Mittal, H.V.R., Al-Mdallal, Q.M. 2018. A numerical study of forced convection from an isothermal cylinder performing rotational oscillations in a uniform stream. International Journal of Heat and Mass Transfer, 127, pp. 357–374.

[36] Anderson, D.A., Tannehil, J.C., Pletcher, R.H. 1984. Computational Fluid mechanics and Heat Transfer. Hemisphere Publishing Corporation, New York.

[37] Tan, Z., Le, D.V., Li, Z., Lim, K.M. and Khoo, B.C., 2008. An immersed interface method for solving incompressible viscous flows with piecewise constant viscosity across a moving elastic membrane. Journal of Computational Physics, 227(23), pp.9955-9983.

[38] Denner, F., Evrard, F., Serfaty, R. and van Wachem, B.G., 2017. Artificial viscosity model to mitigate numerical artefacts at fluid interfaces with surface tension. Computers & Fluids, 143, pp.59-72.

[39] Ge, Z., Loiseau, J.C., Tammisola, O. and Brandt, L., 2018. An efficient mass-preserving interface-correction level set/ghost fluid method for droplet suspensions under depletion forces. Journal of Computational Physics, 353, pp.435-459.

[40] Prosperetti, A., 1981. Motion of two superposed viscous fluids. The Physics of Fluids, 24(7), pp.1217-1223.

[41] Popinet, S., 2009. An accurate adaptive solver for surface-tension-driven interfacial flows. Journal of Computational Physics, 228(16), pp.5838-5866.

[42] Connington, K. and Lee, T., 2012. A review of spurious currents in the lattice Boltzmann method for multiphase flows. Journal of mechanical science and technology, 26(12), pp.3857-3863.

[43] Xu, S. and Wang, Z.J., 2006. An immersed interface method for simulating the interaction of a fluid with moving boundaries. Journal of Computational Physics, 216(2), pp.454-493.

[44] Li, Z., Ito, K. and Lai, M.C., 2007. An augmented approach for Stokes equations with a discontinuous viscosity and singular forces. Computers & Fluids, 36(3), pp.622-635.

[45] Sheth, K.S. and Pozrikidis, C., 1995. Effects of inertia on the deformation of liquid drops in simple shear flow. Computers & fluids, 24(2), pp.101-119.

[46] Farokhirad, S., Lee, T. and Morris, J.F., 2013. Effects of inertia and viscosity on single droplet deformation in confined shear flow. Communications in Computational Physics, 13(3), pp.706-724.