Simulating liquid droplets

A quantitative assessment of lattice Boltzmann and Volume of Fluid methods

Mukherjee, Siddhartha; Zarghami, Ahad; Haringa, Cees; van As, Kevin; Kenjereš, Saša; Van den Akker, Harry E.A.

DOI
10.1016/j.ijheatfluidflow.2017.12.001

Publication date
2018

Document Version
Accepted author manuscript

Published in
International Journal of Heat and Fluid Flow

Citation (APA)
Mukherjee, S., Zarghami, A., Haringa, C., van As, K., Kenjereš, S., & Van den Akker, H. E. A. (2018). Simulating liquid droplets: A quantitative assessment of lattice Boltzmann and Volume of Fluid methods. *International Journal of Heat and Fluid Flow*, *70*, 59-78. https://doi.org/10.1016/j.ijheatfluidflow.2017.12.001

Simulating liquid droplets: a quantitative assessment of lattice Boltzmann and Volume of Fluid methods

Siddhartha Mukherjee[a,*], Ahad Zarghami[b], Cees Haringa[a], Kevin van As[a], Saša Kenjereš[a], Harry E.A. Van den Akker[a,c]

[a] *Section of Transport Phenomena, Department of Chemical Engineering, Faculty of Applied Sciences*
Delft University of Technology
Van der Maasweg 9, 2629 HZ Delft, The Netherlands
[b] *Section of Intensified Reaction & Separation Systems, Department of Process and Energy*
Faculty of Mechanical, Maritime and Materials Engineering
Delft University of Technology
Leeghwaterstraat 34, 2628 CD, Delft, The Netherlands
[c] *Bernal Institute, School of Engineering, Faculty of Science and Engineering*
University of Limerick
Limerick, Ireland

Abstract

While various multiphase flow simulation techniques have found acceptance as predictive tools for processes involving immiscible fluids, none of them can be considered universally applicable. Focusing on accurate simulation of liquid-liquid emulsions at the scale of droplets, we present a comparative assessment of the single-component multiphase pseudopotential lattice Boltzmann method (PP-LB, classical and modified) and the Volume of Fluid method (VOF, classical and modified), highlighting particular strengths and weaknesses of these techniques. We show that a modified LB model produces spurious velocities 1-3 orders of magnitude lower than all VOF models tested, and find that LB is roughly 10 times faster in computation time,

[*] Corresponding author
Email address: S.Mukherjee@tudelft.nl (Siddhartha Mukherjee)

while VOF is more versatile. Simulating falling liquid droplets, a realistic problem, we find that despite identical setups, results can vary with the technique in certain flow regimes. At lower Reynolds numbers, all methods agree reasonably well with experimental values. At higher Reynolds numbers, all methods underpredict the droplet Reynolds number, while being in good agreement with each other. Particular issues regarding LB simulations at low density ratio are emphasized. Finally, we conclude with the applicability of VOF vis-à-vis PP-LB for a general range of multiphase flow problems relevant to myriad applications.

Keywords: Multiphase Flows, Pseudopotential Lattice Boltzmann,
Volume of Fluid, Emulsions, Liquid Droplets

1. Introduction

Multiphase flow simulations are becoming commonplace in describing and designing engineering applications, which has led to the development of numerous numerical techniques [1, 2]. In addition, this has provided unprecedented insight into flow physics at the scale of individual droplets and bubbles which in many cases is relevant at the larger scale of reactors or processes. This has in turn greatly benefited emulsion research dealing with liquid-liquid flows [3, 4, 5], where performing experiments to obtain high resolution spatio-temporal data is often not feasible. Many food and personal care products consist of liquid-liquid emulsions with a low density ratio. In the oil industry, separating water from oil (in particular downhole water/oil separation, leaving the water underground) is a tremendously relevant issue [6], as many oil sources not just produce oil and gas but also water in increasing amounts. Also the concept of Enhanced Oil Recovery (EOR) [7] in which steam is injected into oil containing reservoirs to increase oil production results in large amounts of fine liquid-liquid emulsions which need treatment. In the polymer industry, quite a few polymers are produced by means of emulsion polymerization processes [8]. Reliable simulations of these widely different processes require an accurate description of the flow physics at the droplet scale. The existence of myriad simulation techniques presents another caveat - which method is most applicable to a specific problem? This calls for studies that reveal particular strengths and shortcomings of these simulation techniques when directly compared, however in the case of interface resolving multiphase flows, such studies are difficult to come by.

To the best of the authors' knowledge, among the very few other studies comparing a mesoscopic technique to a continuum technique for multiphase flow are the comparison between a two fluid free energy LB approach and a volume of fluid method by Takada et al. [9], between a pseudopotential LB and a front-tracking finite-difference method for rising bubbles by Sankaranarayanan et al. [10] and between a free energy LB formulation and a phase field method by Scarbolo et al. [11]. We compare two widely used techniques based on very different principles, namely the finite volume based Volume of Fluid (VOF) method and the mesoscopic, single-component multiphase pseudopotential lattice Boltzmann method (PP-LB) and focus on specific challenges faced when simulating fluids at low density ratio. Generally, multiphase flows can contain single or multiple fluid components in different phases, for instance a fluid existing simultaneously in its liquid and vapour phase, or two immiscible liquids like oil and water. The most elusive problem here is the accurate representation of the phase separating interface, that essentially emerges from the microscopic interactions at the molecular level, detached from the continuum regime of hydrodynamics, and which can undergo complex deformations during the evolution of the flow.

VOF and PP-LB differ greatly in how they simulate interface dynamics. Both techniques have particular points of strength, but they also bring a set of shortcomings that are easily overlooked - we address them both in the course of our study.

VOF is one of the first techniques developed for multiphase flow simulations based on the finite volume method (FVM) [12]. Using VOF, one solves an additional advection equation for a boolean valued phase indicator function, marking the two *immiscible* fluids. In principle, at the interface, this indicator value changes rapidly from 0 to 1 (or vice-versa) marking the two phase regions. Further, all physical properties are modeled as phase averages, hence working with an effective *single-fluid* formulation of the Navier-Stokes equation. This method has been demonstrated to perform well for a wide range of multiphase flow problems and can reproduce an appreciably sharp interface undergoing arbitrarily large deformations [13].

Also over the past two decades, PP-LB has emerged as a versatile alternative to conventional finite volume techniques for simulating multiphase flows [14]. It was first proposed by Shan and Chen [15, 16] and is based on a mesoscopic kinetic equation for particle distribution functions, and has been used for various fluid mechanics and engineering problems [17, 18]. In this method, particle interactions are modeled by an inter-particle force which causes a single-component fluid to spontaneously segregate into two phases of different densities that change smoothly from one bulk value to another. The interface emerges automatically and is characterized by monitoring the variation in density, and is therefore no longer a mathematical boundary and no explicit interface tracking/capturing technique is required. Moreover, surface tension effects emerge automatically from the underlying Boltzmann dynamics. The single-component PP-LB is the most widely used LB based model due to its simplicity and versatility, remarkable computational efficiency and clear representation of the underlying microscopic physics.

Evidently, VOF and PP-LB simulate multiphase flows very differently, the first being a continuum approach and the latter mesoscopic. The main point of departure is that in VOF, the two (or more) fluids are completely immiscible and interfacial dynamics is modeled with a specified surface tension force. While in single-component PP-LB, the two fluids are modeled as the liquid and vapor phases of the same non-ideal component [19], which coexist due to phase separation [20]. Immiscibility of the two phases in PP-LB is ensured by a repulsive interaction parameter, consequently leading to surface tension effects. Owing to these differences, a direct comparison between the strengths and weaknesses of the two methods can benefit a user by providing motivation for selecting either technique, for one might be better suited to a specific problem than the other. Previously [21], the authors attempted such an intercomparison and it was observed that even after a careful formulation of identical test cases, predicted results can vary between the two methods. In this study, we investigate this discrepancy further to more conclusively remark on the predictive aspects of VOF vis-à-vis PP-LB.

We begin with a description of PP-LB (our own in-house code implementation using FORTRAN 90 as well as an implementation in the open source Parallel Lattice Boltzmann solver Palabos-v1.5r1), and the VOF method (using the standard OpenFOAM and FLUENT VOF solvers, along with modifications to the OpenFOAM solver). We first address the so-called spurious velocities that emerge in both methods as numerical artifacts and can be a limiting constraint on the accuracy of results. We also compare the thickness of the interface as produced by these methods, and show that sharper interfaces are achieved at the cost of higher spurious velocities. Next we simulate falling droplets with a low density ratio (2D and 3D in VOF, and 2D in LB) to compare how well the methods predict the velocity evolution, terminal Reynolds number and droplet shape. This comparison is performed in a small region of the phase-space governing falling droplets, corresponding to the spherical and ellipsoidal regions of the shape regime map of Clift et al. [22] (henceforth called the Clift map). We then discuss the specific challenges faced for the two methods and conclude with our main findings.

2. Numerical Methods

2.1. Single-component PP-LB

The standard lattice Boltzmann equation with a single relaxation time [23] is written as

$$\frac{f_i\left(\mathbf{x}+\mathbf{e}_i\Delta t, t+\Delta t\right)-f_i\left(\mathbf{x},t\right)}{\Delta t}=\frac{1}{\tau}\left(f_i\left(\mathbf{x},t\right)-f_i^{\mathrm{eq}}\left(\mathbf{x},t\right)\right)+\mathbf{S}_i \tag{1}$$

where $f_i\left(\mathbf{x},t\right)$ is the density distribution function associated with discrete velocity direction i, and $\mathbf{S}_i$ represents a general source term added into the lattice Boltzmann equation (which is related to all thermodynamic and hydrodynamic forces). The discrete velocities $\mathbf{e}_i$ in the i^{th} - direction, for the D2Q9 lattice are given by $\mathbf{e}_0=0$ and $\mathbf{e}_i=\lambda_i\left(\cos\theta_i,\sin\theta_i\right)$ with $\lambda_i=1,\theta_i=\left(i-1\right)\pi/2$ for $i=1-4$ and $\lambda_i=\sqrt{2},\theta_i=\left(i-5\right)\pi/2+\pi/4$ for $i=5-8$. The order numbers $i=1-4$ and $i=5-8$ represent the rectangular and the diagonal directions of the lattice respectively. In Eq. 1, f_i^{eq} is the equilibrium distribution function and is calculated as

$$f_i^{\mathrm{eq}}=w_i\rho\left[1+\frac{\left(\mathbf{e}_i\cdot\mathbf{u}^{\mathrm{eq}}\right)}{c_s^2}+\frac{\left(\mathbf{e}_i\cdot\mathbf{u}^{\mathrm{eq}}\right)^2}{2c_s^4}-\frac{\left(\mathbf{u}^{\mathrm{eq}}\cdot\mathbf{u}^{\mathrm{eq}}\right)}{2c_s^2}\right] \tag{2}$$

where $c_s^2 = 1/3$ is the lattice speed of sound, w_i are the weighting factors equal to 4/9 for $i = 0$, 1/9 for $i = 1 - 4$ and 1/36 for $i = 5 - 8$, and $\mathbf{u}^{\text{eq}}$ is the equilibrium velocity. Also, the local mass density, local velocity and the viscosity in the lattice units for each component are calculated as $\rho = \sum_i f_i$, $\mathbf{u} = \left(\sum_i \mathbf{e}_i f_i\right)/\rho$ and $\nu = (\tau - 0.5)/3$ respectively.

The force $\mathbf{F}$ acting on a multiphase system includes external body forces, $\mathbf{F}_{\text{body}}$ (e.g. gravity) and the mean field inter-particle interaction force, $\mathbf{F}_{\text{int}}$, and is written as $\mathbf{F} = \mathbf{F}_{\text{body}} + \mathbf{F}_{\text{int}}$. Based on the original pseudopotential model [16], the so called $\beta-$scheme [24] has been introduced for the interaction force for a single-component multiphase system as follows:

$$\begin{aligned}\mathbf{F}_{\text{int}} = -\beta &\left[\psi(\mathbf{x}, t)\, G \sum_i \psi(\mathbf{x} + \mathbf{e}_i \Delta t, t)\, \mathbf{e}_i \Delta t\right] \\ \frac{1-\beta}{2} &\left[G \sum_i w_i \left[\psi(\mathbf{x} + \mathbf{e}_i \Delta t, t)\right]^2 \mathbf{e}_i \Delta t\right]\end{aligned} \tag{3}$$

where $\Delta t = 1$ is the time interval and G denotes the interaction parameter, with $G < 0$ representing an attractive force between the particles. Compared to the original pseudopotential interaction force, the $\beta-$scheme has more isotropy and by choosing a proper value for β, the thermodynamic inconsistency and magnitude of spurious velocities can be greatly reduced [24, 25, 26]. It has been shown that by setting $\beta = 1.25$ the density ratio in LB matches well with the analytical equation of state (EOS) [25]. Note that by choosing $\beta = 1$, the $\beta-$scheme reduces to the original pseudopotential model. In Eq. 3, $\psi(\mathbf{x}, t)$ is called the pseudopotential function and is calculated by [20]

$$\psi(\mathbf{x}, t) = \sqrt{\frac{2\left(p_{\text{EOS}} - \rho c_s^2\right)}{G}} \tag{4}$$

where p_{EOS} is the pressure calculated from the desired EOS. Here, the

Carnahan-Starling (C-S) EOS is used [20], which is implemented as

$$p_{\text{EOS}} = \rho T \left[\frac{1 + \rho + \rho^2 - \rho^3}{(1-\rho)^3} \right] - \rho^2 \tag{5}$$

The reduced temperature and density are defined as $T_r = T/T_c$ and $\rho_r = \rho/\rho_c$ where $T_c = 0.09432$ and $\rho_c = 0.11911$ are the critical temperature and density related to the C-S EOS, these values have been take from Zarghami et al. [25]. In order to mimic the continuous phase behavior as expressed by the Navier-Stokes equations, usually, a source term representing the mean field inter-particle interaction force is added to the lattice Boltzmann equation (see Eq. 1). One of the most stable and common force implementation schemes is the exact difference method (EDM) which is directly derived from the Boltzmann equation [27], and is given as

$$\mathbf{S}_i = f_i^{\text{eq}} (\rho, \mathbf{u} + \mathbf{F}\Delta t/\rho) - f_i^{\text{eq}} (\rho, \mathbf{u}) \tag{6}$$

The equilibrium velocity $\mathbf{u}^{\text{eq}}$, and the real fluid velocity $\mathbf{U}$ can be calculated as

$$\mathbf{U} = \mathbf{u}^{\text{eq}} = \mathbf{u} + \frac{\Delta t}{2\rho}\mathbf{F} \tag{7}$$

In order to incorporate gravitational and buoyancy forces in the model, the body force can be defined in one of the following ways depending on the specific problem being simulated and the domain boundary conditions

$$\mathbf{F}_{\text{body}} = \rho g \tag{8}$$

$$\mathbf{F}_{\text{body}} = (\rho - \rho_{\text{vap}})g \tag{9}$$

$$\mathbf{F}_{\text{body}} = (\rho - \hat{\rho})g \tag{10}$$

where g is the downward gravitational acceleration, ρ is the local density, ρ_{vap} is the vapor density and $\hat{\rho}$ is the averaged density over the whole computational domain. Since we use a fully periodic domain for the LB simulations, Eq. 10 is the most appropriate implementation of the body force [28, 29, 30]. This ensures that the average value of $\mathbf{F}_{\text{body}}$ in the computational domain is zero and no net momentum is added to the system, so the droplet and surrounding fluid do not keep accelerating downward indefinitely [30]. The recovered macroscopic Navier-Stokes equations from LB are as follows

$$\begin{aligned}\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{u} &= -\frac{\Delta t}{2}\nabla \cdot \mathbf{F} \\ \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) &= -\nabla p + \nu \nabla \cdot \left[\rho \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T\right)\right] + \mathbf{F} \\ &\quad - \frac{\Delta t}{2}\epsilon \frac{\partial \mathbf{F}}{\partial t_1} - \Delta t \nabla \cdot \left[\frac{1}{2}(\mathbf{u}\mathbf{F} + \mathbf{F}\mathbf{u}) + \Delta t \frac{\tau}{\rho}\mathbf{F}\mathbf{F}\right] \end{aligned} \tag{11}$$

One can see that an additional nonlinear relaxation-time dependent term is introduced when recovering the NS equations using the EDM scheme and when applying the Chapman-Enskog expansion. The scheme will be consistent with the macroscopic equations if the temporal and spatial changes of the force vary only slightly (or the force term be constant), and is the last term within brackets of Eq. 11 be negligible. However, the last term may have a great influence on the solution due to the velocity gradient. It has been shown [25, 31] that the term $\rho^{-1}\mathbf{F}\mathbf{F}$ is capable of enhancing numerical stability. This term has a non-zero value only at the phase interface, as a result of the fluid particle interaction (which in itself does not appear in the NS equations). This enables one to simulate high density ratios when using the EDM scheme.

2.2. VOF in OpenFOAM

OpenFOAM (version 2.3.0) [32, 33] which stands for Open Field Operation and Manipulation, is a highly flexible open source C++ library of finite volume based solvers for differential equations, operating on scalar, vector and tensor fields. It provides a VOF solver called *interFoam* - which is employed in this study and has been widely used and validated [34, 35, 36, 37, 38]. It solves the mass conservation and momentum equation for incompressible fluid flow along with the transport of a phase indicator function α which differentiates the two phases. This α is ideally a step function such that it goes from 1 to 0 as one moves from one phase to the other. Once a simulation is initialized with a sharp profile of α, the interface gets diffused over a few cells, which can in principle be confined to an arbitrarily narrow region with mesh refinement, and several methods have been proposed to limit this effect [39]. The interface between the two fluids is taken to be the contour of $\alpha = 0.5$. This approach assumes that each phase moves with the center of mass velocity such that $\mathbf{u} = \mathbf{u}_{\alpha_1} = \mathbf{u}_{\alpha_2}$ and is acted upon by one pressure field. The following equations are solved by interFoam

$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \rho u}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla p^{\star} + \nabla \cdot (2\mu \mathbf{S}) + \mathbf{f}_b + \mathbf{f}_\sigma \tag{12}$$

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\mathbf{u}\alpha) + \nabla \cdot (\mathbf{u}_c \alpha (1 - \alpha)) = 0 \tag{13}$$

where $\mathbf{f}_b$ denotes body forces (like gravity), $\mathbf{f}_\sigma$ is surface tension force (explained below), $2\mu\mathbf{S}$ is the deviatoric stress with $\mathbf{S}$ the rate of strain tensor $\mathbf{S} = \frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^T\right)$. Here $p^{\star}$ is the modified pressure found by removing the hydrostatic component from the pressure, and is calculated as [40]

$$p^{\star} = p - \rho \mathbf{g} \cdot \mathbf{x} \tag{14}$$

where $\mathbf{g}$ and $\mathbf{x}$ are the gravity and position vectors respectively. Therefore the term $\nabla p^{\star}$ expands to

$$\nabla p^{\star} = \nabla p - \rho \mathbf{g} - \mathbf{g} \cdot \mathbf{x} \nabla \rho \tag{15}$$

The last term of Eq. 13 of the phase indicator transport equation $\nabla \cdot (\mathbf{u}_c \alpha(1-\alpha))$ is an *interface compression* term used to maintain a sharp interface between the phases. The calculation of $\mathbf{u}_c$ follows [41], where the compression velocity is given as

$$\mathbf{u}_c = \min\left(c_\alpha \left| \frac{\phi}{|\mathbf{S}_f|} \right|, \left| \frac{\phi}{|\mathbf{S}_f|} \right|_{\max} \right) \tag{16}$$

where $0 < c_\alpha < 1$ limits the compression velocity to below the maximum face flux velocity $\phi/|\mathbf{S}_f|$ [42], where $\mathbf{S}_f$ is the cell face vector. This term is active only in the interface region due to the $\alpha(1-\alpha)$ factor (conceptually similar to the extra term appearing in the LB formulation of the NS equation, see the description following Eq. 6). This technique helps preserve interface sharpness, though it comes at the cost of exaggerating the parasitic currents [37], which will be discussed shortly. All physical properties ψ in this formulation are given as

$$\psi = \psi_1 \alpha + \psi_2 (1-\alpha) \qquad \text{where} \quad \psi \in \{\rho, \mu, c\} \tag{17}$$

The surface tension force in the momentum equation, $\mathbf{f}_\sigma$, is adopted based upon the Continuous-Surface-Force (CSF) formulation [43], given as

$$\mathbf{f}_\sigma = \sigma \kappa \nabla \alpha \qquad \text{with} \quad \kappa = -\nabla \cdot \left(\frac{\nabla \alpha}{|\nabla \alpha|} \right) \tag{18}$$

where σ is a specified constant surface tension. It is the inaccuracies in determination of the curvature κ that lead to parasitic currents in VOF. Lafaurie et al. [44] proposed an improved calculation of the curvature reducing parasitic currents, where the α field is converted to a smoother field $\widetilde{\alpha}$ using a Laplacian filter as follows

$$\widetilde{\alpha}_P = \frac{\sum_{f=1}^{n} \alpha_f |\mathbf{S}_f|}{\sum_{f=1}^{n} |\mathbf{S}_f|} \tag{19}$$

where P denotes the cell index and f denotes the face index. This modified formulation can be used to reduce the parasitic currents by an order of magnitude [37].

Dynamic Mesh Refinement for 3D simulations

For 3D VOF simulations, dynamic mesh refinement [45] as available in OpenFOAM was used to reduce the computational cost. This allows the use of a relatively coarse base mesh, while grid cells around the interface can be dynamically refined based on the value of the α-field (a criterion of $0.1 < \alpha < 0.9$ has been used in this study). Every octagonal cell that satisfies the refinement criterion has each dimension cut in half, resulting in 8 smaller octagonal cells, the result being a $2:1$ refinement. This process is repeated up to two times, resulting in a factor four times finer mesh in the region around the interface.

Pressure-velocity coupling

OpenFOAM provides two pressure-velocity coupling procedures for the iterative solver. First is the well known PISO algorithm [46], which corrects iteratively for the pressure and velocity for the number of times prescribed by the parameter *inner corrector*. The second is the PIMPLE algorithm, which is a combination of the SIMPLE [47] algorithm with PISO. The PIMPLE algorithm has *outer correctors* along with the *inner correctors* of PISO. It solves the entire PISO loop several times, the main difference being that the α field is also corrected by PIMPLE (unlike with PISO). If the time step size is small enough (i.e. a small enough $Co_{\mathrm{max}]}$), then the α field should not change much within a time step and PISO and PIMPLE are expected to give very similar results. We have tested both algorithms in this study.

2.3. VOF in FLUENT

The explicit VOF method in FLUENT (version 15.7) is equivalent to the implementation in OpenFOAM in terms of the momentum; the volume fraction equation differs depending upon the interface reconstruction approach. The FLUENT default is the geometrical reconstruction scheme based upon Youngs [48]. For this scheme, the volume fraction equation simply reads:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\mathbf{u}\alpha) = 0 \tag{20}$$

and a piecewise-linear interface construction (PLIC) scheme is used to determine the face fluxes of α. Alternatively, a “Compressive” scheme can be chosen from the available options (others being “HRIC” and “CICSAM” - which have not been used in this study). The Compressive scheme has been implemented in FLUENT via the volume fraction discretization equation

$$\alpha_f = \alpha_d + \beta \nabla \alpha_d \tag{21}$$

with α_f the face value and α_d the value in the donor cell, further details can be found in Ubbink [49]. Prescribing $\beta = 2$ amounts to the Compressive scheme, as it increases the flux of α being accepted from the donor cell. (while $\beta = 0$ yields a $1^{\mathrm{st}}-$order upwind discretization as $\alpha_f = \alpha_d$). The momentum equation is discretized using $2^{\mathrm{nd}}-$ order upwind and the 3^{rd} order *MUSCL* scheme, which is a blend between second order upwind and central differencing:

$$\begin{aligned} \phi_f = \theta \left[\frac{1}{2} (\phi_0 + \phi_1) + \frac{1}{1} (\nabla \phi_o \cdot \mathbf{r_0} + \nabla \phi_o \cdot \mathbf{r_1}) \right] + \\ (1 - \theta) \left[\phi_0 + \nabla \phi_0 \cdot \mathbf{r_0} \right] \end{aligned} \tag{22}$$

This gives the following FLUENT approaches:

1. 3^{rd}-order MUSCL using “Compressive” Scheme
2. 3^{rd}-order MUSCL using “Geometrical-Reconstruction” Scheme

For all cases, the SIMPLE scheme for pressure-velocity coupling and a staggered grid (PRESTO, i.e Pressure Staggering Option scheme [47]) for pressure interpolation was used. A first order implicit, adaptive time-stepping was used with a Courant criterion of $Co_{\max} = 0.25$. We also tested results from the $2^{\text{nd}}-$ order upwinding scheme, which were identical to those from the MUSCL scheme. Hence all forthcoming FLUENT results shown are those obtained using the MUSCL scheme.

3. Theoretical Discussion

3.1. Dimensional Analysis

We simulate falling droplets, a relevant multiphase flow problem along with rising bubbles, where the dynamics is commonly influenced by buoyancy, drag, gravity, surface tension and other forces depending on the fluid properties. The interplay of these forces results in different droplet/bubble behaviours, depending on droplet/bubble size, density ratio and viscosity ratio of the fluids. Carrying out a dimensional analysis of the relative (slip) velocity for a falling (rising) droplet (bubble) in another fluid shows that the problem is completely described by four non-dimensional parameters [50]: the density ratio ($\rho^{\star}$), the dynamic viscosity ratio ($\mu^{\star}$), the Eötvös number (Eo, also called the Bond number) and the Galilei number (Ga) defined as

$$
\begin{aligned}
\text{Density Ratio: } \rho^\star &= \frac{\rho_i}{\rho_o} \\
\text{Dynamic Viscosity Ratio: } \mu^\star &= \frac{\mu_i}{\mu_o} \\
\text{Eötvös Number: } Eo &= \frac{\Delta\rho g d^2}{\sigma} \\
\text{Galilei Number: } Ga &= \frac{\sqrt{g\rho_o\Delta\rho d^3}}{\mu_o}
\end{aligned}
\tag{23}
$$

where $\Delta\rho = |\rho_o - \rho_i|$, the subscripts i, o denoting properties inside and outside the droplet/bubble region respectively. Also, g, d and σ are the gravitational acceleration, the initial diameter of droplet and surface tension. For our discussion and subsequent formulation of cases based on the Clift map, we also define the Reynolds (Re), Morton (M) and Weber (We) numbers

$$
\begin{aligned}
\text{Reynolds Number: } Re &= \frac{\rho_o u_t d}{\mu_o} \\
\text{Morton Number: } M &= \frac{\mu_o^4 \Delta\rho g}{\rho_o^2 \sigma^3} \\
\text{Weber Number: } We &= \frac{\rho_o u_t^2 d}{\sigma}
\end{aligned}
\tag{24}
$$

where u_t is the terminal velocity of the droplet. Note that Ga is similar to Re, but is defined using the characteristic gravitational velocity ($\sqrt{(\Delta\rho/\rho_o)gd}$) rather than the terminal velocity [51]. This becomes useful in some flow regimes where a falling droplet or rising bubble may never display a steady terminal velocity, and even if it does, the terminal velocity is unknown a priori and depends on several physical quantities; the gravitational velocity scale then is better defined.

3.2. Flow regime map

A well-known regime map based on experimental correlations has been presented by Clift et al. [22], which is used as a common reference for predicting the terminal velocity and final shape of rising air bubbles in water, also applicable to low density ratio droplets. Based on this map, the main regimes depending on the final shape of the bubble are the spherical, ellipsoidal and spherical cap regimes. The surface tension and the viscous forces are dominant in the spherical regime and the bubble size is small ($d < 1.3$ [mm]), the bubble shape remaining spherical or nearly-spherical. Here, the Hadamard-Rybzynski [52, 53] solution for flow past viscous spheres applies, which is given as

$$u_t = \frac{gd^2\Delta\rho}{6\nu_o}\left(\frac{1+k}{2+3k}\right) \tag{25}$$

where $k = \nu_o/\nu_i$ ($k = 1$ in this study). In the ellipsoidal regime which ranges from $0.25 < Eo < 40$, surface tension is the dominant parameter and the bubble size is typically $1.3 < d < 6$ [mm]. Here, the terminal velocity can be approximated by correlations suggested by Mendelson [54]

$$u_t = \left(\frac{2.14\sigma}{\rho_o d} + 0.505gd\right)^{0.5} \tag{26}$$

Finally, the spherical cap regime occurs when the bubble size is large ($d > 6$ [mm]) and $Eo > 40$, and is governed by inertial forces which generate higher deformation by inducing higher pressure on the front and rear of the bubble, and lower pressure at the sides [28]. The wobbling, skirted or dimpled spherical cap sub-regimes may also emerge in case of an unsteady rising bubble.

Whether this regime map is also applicable to falling droplets can be ascertained by comparing the conditions under which bubbles and droplets have the same dynamics and evolution history, i.e. they rise or fall to a similar relative position over the same time and acquire the same shape and velocity. As reported in recent literature [50, 55], this is ensured by the conditions

$$
\begin{aligned}
Ga_d &= Ga_b \\
\rho_d^\star &= \frac{\rho_b^\star}{2\rho_b^\star - 1}
\end{aligned}
\tag{27}
$$

where the subscripts d, b refer to the droplet and bubble respectively. Further in accordance to the Boussinesq approximation, the influence of circulation inside the fluid particle on its surface should be considered [55], which gives the condition $\rho_b^\star = 2 - \rho_d^\star$ and $\mu_b^\star = \mu_d^\star$. Once $\rho_d^\star > 2$, no exact equivalent of a bubble can be found [50, 56, 57]. Essentially, the dynamics of droplet motion becomes qualitatively different from an equivalent bubble when the density ratio is far from unity. This has been briefly demonstrated below in Fig. 1, where identical cases of rising bubbles and falling droplets have been shown side by side, comparing the steady state shape and the background vorticity field, for density ratios $\rho^\star = 1.2, 3$ & 20 and with $\mu^\star = 1$. These simulations were performed in OpenFOAM in 2D at a resolution of 40 grid cells per droplet diameter, and an adaptive time stepping following the maximum Courant number criterion of $Co_{\max} = 0.25$. It is seen that the vorticity tends to concentrate more strongly in the lighter density fluid. As $\rho^\star$ deviates further from unity, the equivalent bubble and droplet begin to differ.

4. Stationary liquid droplets in vapor

4.1. Simulation details

We first wish to address the well-known *spurious* (or *parasitic*) currents that are known to arise in VOF and PP-LB alike, and which have been de-

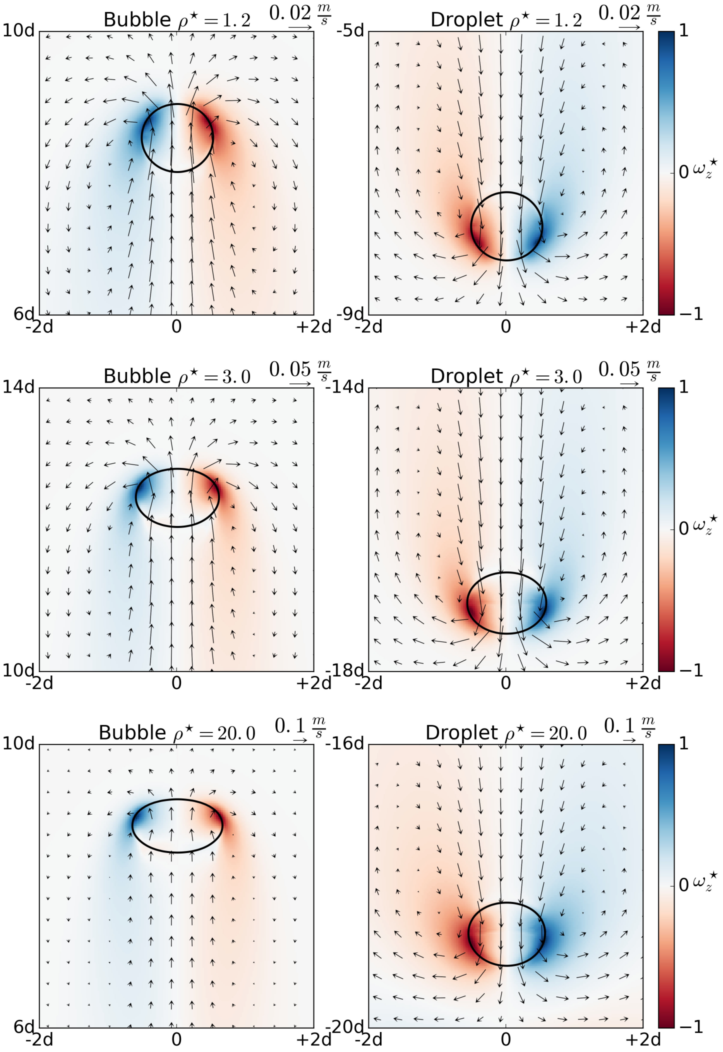


Figure 1: Comparison of equivalent bubble-droplet pairs for density ratio $\rho^\star = 1.2, 3$ & 20, shown along with the normalized vorticity field $\omega_z^\star = \omega_z / |\omega_z|_{\max}$ at steady state. In these cases, the bubble is seen to deform more than the droplet, while the droplet generates more intense and longer trailing vorticity fields. As the density ratio gets farther from unity, the bubble-droplet pair begins to drastically differ. These results are obtained from simulations performed in OpenFOAM in 2D.

tailed before in literature [58, 59]. As a brief reminder, these are *unphysical* velocity fields that arise due to the continuum surface force implementation for modeling surface tension effects in Eulerian based multiphase flow simulations like VOF, which do not disappear with grid refinement. This is a consequence of simulating a curved interface using an orthogonal grid. In PP-LB, these arise from an insufficient isotropy of the gradient operator. The standard method of investigating spurious currents is to simulate a stationary droplet suspended in vapor, without the influence of any body forces. In such a situation, although all initial velocity fluctuations should eventually settle to zero by the effect of viscosity, very coherent velocity structures are obtained particularly neighbouring the fluid interface. We simulate liquid droplets (ρ_l) suspended in vapor (ρ_v), centered in a periodic domain of 150 × 150 uniform orthogonal cells, with a viscosity ratio $\nu^\star = 1$ (as we use single-component LB) and increase the density ratio $\rho^\star = \rho_l/\rho_v$.

LB parameters: These simulations are performed using our in-house code. The droplet diameter is set to 60 lattice units initially, with the density inside and outside the droplet varying smoothly over 6 lattice units using a hyperbolic tan profile. These densities are initialized very close to the corresponding liquid and vapor densities found from the coexistence curve [25]. The density ratio is varied by changing the reduced temperature T_r. Two sets of simulations with $\beta = 1$ (i.e. classical multiphase PP-LB) and $\beta = 1.25$ (so-called β scheme) are performed (refer Eq. 3). The relaxation time $\tau = 1$, which gives a LB viscosity of $\nu = 0.16$.

OpenFOAM-VOF parameters: In VOF, the droplet diameter is initialized with a diameter of $d = 2$ [mm], at 60 grid cells per diameter, which is a sharp initial profile of α which becomes slightly diffused once the simulation begins. Surface tension values have to be provided in the VOF simulations as a parameter (refer Eq. 18). These are obtained from the corresponding LB simulations using the Laplace law ($\Delta p = 2\sigma/R$) at steady state, where Δp is the pressure difference between the center of the droplet and the edge of the domain. Further, three sets of simulations with OpenFOAM are performed

1. Classical VOF, i.e. without interface compression ($c_\alpha = 0$)
2. VOF with interface compression ($c_\alpha = 1$), which produces a sharp interface but increases the spurious currents as it steepens the α field gradients at the interface (Klostermann et al. [60] report c_α does not influence the magnitude of spurious currents much, however Hoang et al. [37] demonstrate that a higher c_α generates stronger spurious currents)
3. VOF with interface compression ($c_\alpha = 1$), with an added α smoothing step during calculation of the curvature, which is called *VOFsmooth* after Hoang et al. [37] who implemented it in OpenFOAM, based upon Lafaurie et al. [44]. This case is aimed to specifically demonstrate the utility of the α smoothing function, despite a compressive scheme to preserve a sharp interface.

Euler time integration (with adaptive time stepping following the limiting Courant criteria $Co_{\max} = 0.25$), with *Gauss linear* interpolation of gradient terms and *Gauss vanLeer* interpolation of the advection terms is used. *Backward* time integration was also tried, however it did not influence these results much. A single PISO loop was used with 10 corrector steps, and the tolerance criterion for convergence was kept at 10^{-7} for the pressure term and 10^{-8} for the velocity.

FLUENT-VOF parameters: The FLUENT simulations are performed for two sets of cases, the first with geometrical-reconstruction and the other with an interface compression technique similar to OpenFOAM. The residuals of velocity and momentum were kept to their default values of 10^{-5} and it was checked that the average velocity in the domain was constant at this value within a time step. Time integration performed was *first order implicit* (equivalent to the OpenFOAM *Euler* integration) - other than which is not allowed for explicit VOF in FLUENT and *least squares* interpolation for the gradient terms was used. An adaptive timestepping with a maximum timestep size of 10^{-4} [s] was used along with the same Courant criterion of $Co_{\max} = 0.25$.

4.2. Spurious Currents

Spurious velocity vectors after a long simulation time are shown for the case $\rho^{\star} = 3$ in Fig. 2 with contours of $\alpha = 0.5$ for the VOF methods implemented in OpenFOAM (FLUENT results are qualitatively similar) and for LB ($\beta = 1.25$). The normalized vorticity fields ($-1 < \omega_z^{\star} < 1$, red to blue, where $\omega_z^{\star} = \omega_z/|\omega_z|_{\max}$) have also been shown. Fig. 2a shows classical VOF which has a slightly diffused interface and produces only a small magnitude of spurious velocities near the interface, velocity vectors distant from the interface being almost of length zero. Fig. 2b shows how using interface compression ($c_\alpha = 1$) steepens the magnitude of spurious velocities, which now appear with longer vectors. Upon using VOFsmooth (Fig. 2c) these vectors again become smaller in magnitude, while the scheme preserves a sharp interface. Lastly, Fig. 2d shows the spurious velocities in LB, which after having attained a steady state do not exhibit fluctuations (unlike in VOF) and attain a very symmetric profile. Note that the overall shape and magnitude of the spurious velocity field in LB can change depending on the discretization of the force term into conservative or non-conservative forms, as has been explained by Connington and Lee [66]. These results are representative of how the velocity fields look in VOF and LB. In the VOF simulations we observe that for high $\rho^\star$ values (i.e. also high σ) the droplet can perform a random walk in the domain driven by large asymmetric spurious currents, as has also been reported in literature [37].

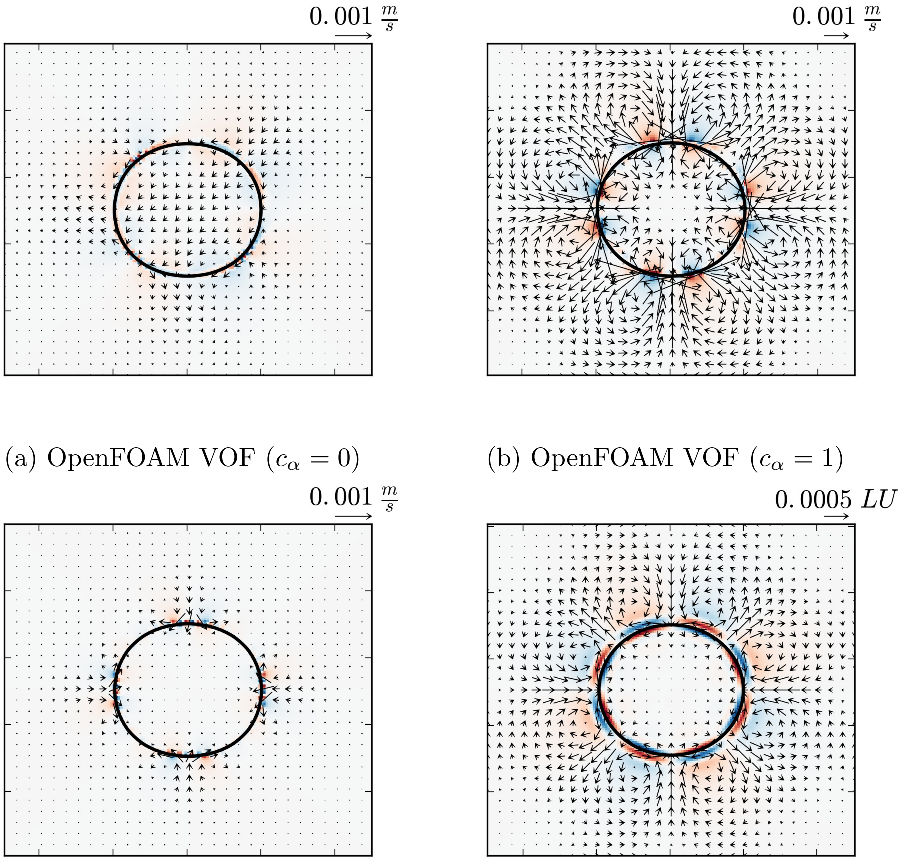
Since there is no steady state behaviour in the VOF simulations for the global average spurious velocity or the maximum spurious velocity magnitude, a single value is ascertained for each simulation by time averaging the maximum spurious velocity magnitude $|u|_{\max}$ between 0.2 to 1.0 [s] to get $\langle |u|_{\max} \rangle$, as shown in Fig. 3. It was checked that this time average is representative, as performing a similar averaging over a 10 times longer signal between 0.2 to 10.0 [s], for one of the cases, yielded a value within $2-4\%$ of the shorter time average.

Finally to compare the magnitude of these spurious velocities between different techniques, a "Spurious Reynolds number" Re_{sp} is defined as follows

$$Re_{\text{sp}} = \frac{\langle |u|_{\max} \rangle \, d}{\nu} \tag{28}$$

where $\langle |u|_{\max} \rangle$ is the time averaged maximum spurious velocity for the VOF simulations ($\langle |u|_{\max} \rangle$ is taken to be the steady state value for LB simulations). Re_{sp} over increasing $\rho^\star$ is shown in Fig. 4.

LB with $\beta = 1.25$ is seen to produce Re_{sp} 1-3 orders of magnitude lower than all VOF formulations, while the classical LB approach ($\beta = 1.0$) becomes comparable to VOF around $\rho^\star = 300$. Generally among the VOF methods, interface sharpening ($c_\alpha = 1$) in OpenFOAM increases Re_{sp} by almost an order of magnitude as compared to the classical VOF ($c_\alpha = 0$).



(a) OpenFOAM VOF ($c_\alpha = 0$)

(b) OpenFOAM VOF ($c_\alpha = 1$)

(c) OpenFOAM VOFsmooth ($c_\alpha = 1$)

(d) LB $\beta = 1.25$

Figure 2: Spurious velocity vectors for $\rho^\star = 3$ are shown for the three VOF methods implemented in OpenFOAM (FLUENT results being qualitatively similar to OpenFOAM) and for LB ($\beta = 1.25$). The normalized vorticity fields ($-1 < \omega_z^\star < 1$, red to blue, where $\omega_z^\star = \omega_z/|\omega_z|_{\max}$) have also been shown. The classic VOF method in OpenFOAM ($c_\alpha = 0$) has a slightly diffused interface and produces small spurious velocities near the interface. Using VOF with $c_\alpha = 1$ sharpens the interface while increasing the magnitude of the spurious velocity vectors. VOFsmooth with $c_\alpha = 1$ shows the utility of the α smoothing function which reduces the spurious velocities while maintaining a sharp interface. Lastly, in LB these emerge as symmetric, non-fluctuating counter rotating vortices.

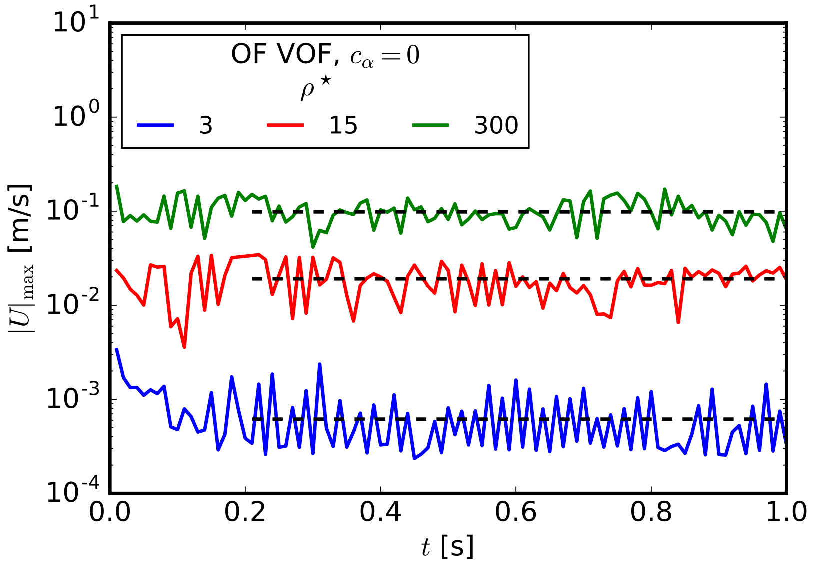


Figure 3: The averaging performed on the maximum spurious velocity magnitude $|u|_{\max}$ (between times 0.2 and 1.0 [s]) to ascertain a representative spurious velocity magnitude $\langle |u|_{\max} \rangle$ for VOF simulations is shown for the OpenFOAM simulations for classical VOF.

Upon smoothing the interface during curvature calculation, Re_{sp} can be brought down an order of magnitude, and this effect is more prominent at higher density ratios. The two FLUENT VOF methods produce very similar Re_{sp} numbers, which at high density ratios remain close to the VOFsmooth values. The differences between the FLUENT and OpenFOAM results here cannot be ascribed to the difference in the tolerances used for the solvers (i.e. 10^{-8} and 10^{-5} for velocity in OpenFOAM and FLUENT respectively). This is because the minimum magnitude of the spurious currents ($\mathcal{O}(10^{-3})$) is still 100 times larger than the 10^{-5} tolerance. We performed a test with the same solver in OpenFOAM for the two tolerance values of 10^{-5} and 10^{-8}, and the spurious velocity magnitudes were within 5% of each other,

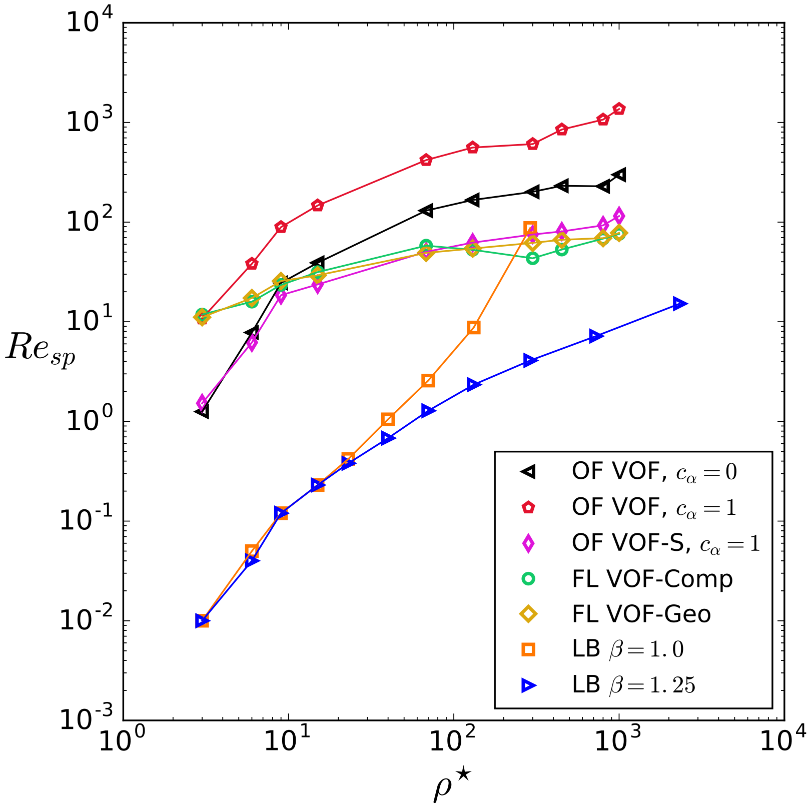


Figure 4: Spurious Reynolds number Re_{sp} over density ratio $\rho^\star$, for different solver cases is shown. LB with $\beta = 1.25$ produces Re_{sp} $1 - 3$ orders of magnitude lower than all VOF formulations. Among the VOF methods, using interface compression in OpenFOAM increases Re_{sp} by an order of magnitude as compared to classical VOF, which can be remedied by using VOFsmooth which reduces Re_{sp} significantly while maintaining a sharp interface. The FLUENT VOF methods produce very similar Re_{sp}, that are close to the VOFsmooth values for higher $\rho^\star$. Note that all these simulations were performed in 2D.

whereby not influencing the results in Fig. 4.

Due to the many differences between LB and VOF simulations, it is not possible to directly comment on why LB produces much lower spurious currents than VOF purely in terms of tolerances, numerical schemes etc. At low density ratios ($\rho^\star \sim 3 - 60$), where classical LB performs as well as the modified $\beta-$scheme, this difference is stark as Re_{sp} is almost 1000 times lower in LB than in VOF. This can partly be ascribed to the thicker interfaces in LB at low density ratios. Further, the non-dimensional LB units cannot be mapped to a physical set of units in this case as there are insufficient variables for a stationary droplet for a mapping. Since LB with $\beta = 1.25$ produces Re_{sp} 1000 times lower than all VOF methods, it appears better suited for simulating large density ratio flows (like air-water) at low Re - which would be difficult with VOF as spurious currents will be of the same order of magnitude or more than the physical velocity scales. Also, while simulating heat and mass transfer problems, spurious currents will increase scalar transport across the interface, where having lower spurious velocities can be a decisive advantage for a simulation technique.

In addition in LB simulations, the density ratio $\rho^\star$ and interface thickness (both of which are determined by the reduced temperature T_r used in the EOS) and β influence the magnitude of Re_{sp}, while in VOF, it is $\rho^\star$ and the value of the surface tension parameter σ. For instance in VOF, for the same value of σ (8×10^{-4} [N/m]), changing $\rho^\star$ from $\mathcal{O}(1)$ to $\mathcal{O}(100)$ only changes Re_{sp} by a factor of ~ 3, while increasing σ (from 8×10^{-4} to 2×10^{-2} [N/m], in accordance to the LB simulations) increases Re_{sp} by a factor ~ 100. Another aspect is that Re_{sp} can increase up to $2 - 5$ times in VOF based upon the specific choice of solvers - specially if using higher order discretization schemes (which is not shown here but was observed).

It is worthwhile to note that an alternative FVM method with an exact interface representation using body-fitted coordinates, as presented in the PROST scheme of Renardy and Renardy [61], can virtually eliminate these parasitic currents. Though such a formulation comes with its own limitations, namely a higher computational cost of re-meshing the domain at each time step along with interpolation of values to new cell faces. In PP-LB, upon increasing isotropy by introducing mid-range interactions, spurious velocities can be made to practically vanish as demonstrated by Sbragaglia et al. [62]. Such a formulation in turn breaks the ease of parallelization of the LB code, as the interactions are no longer local and successively distant lattice nodes begin to influence an otherwise compact computational molecule centered at each lattice node.

4.3. Interface Thickness

Focusing on the sharpest reproducible interface, the phase indicator function α has been shown in Fig. 5 across the droplet interface (see schematic in the figure), for $\rho^\star = 15$ & 1000, for OpenFOAM VOF with $c_\alpha = 1$, FLUENT VOF with geometrical reconstruction and LB with $\beta = 1.25$ (in LB the density field has been normalized to a phase indicator). As a reminder, the VOF simulations are initialized with a sharp α profile, and the LB simulations start with a droplet with the interface smoothed over 6 lattice units. These interfaces then diffuse during the simulation, and the final interface thicknesses are independent of the initialization.

It is seen that FLUENT produces the sharpest interface with the geometrical reconstruction scheme, an interface of $2-3$ grid cells, closely followed by OpenFOAM with interface compression where an interface of $3-4$ grid cells is obtained, over all $\rho^\star$ values. LB produces thicker interfaces, ranging from $8-11$ lattice units for low $\rho^\star$ to $3-5$ for a higher $\rho^\star$. This $\rho^\star$ dependence comes from the pseupotential force at the interface which more strongly repels the lighter phase when $\rho^\star$ is high, and is weakly repelling for

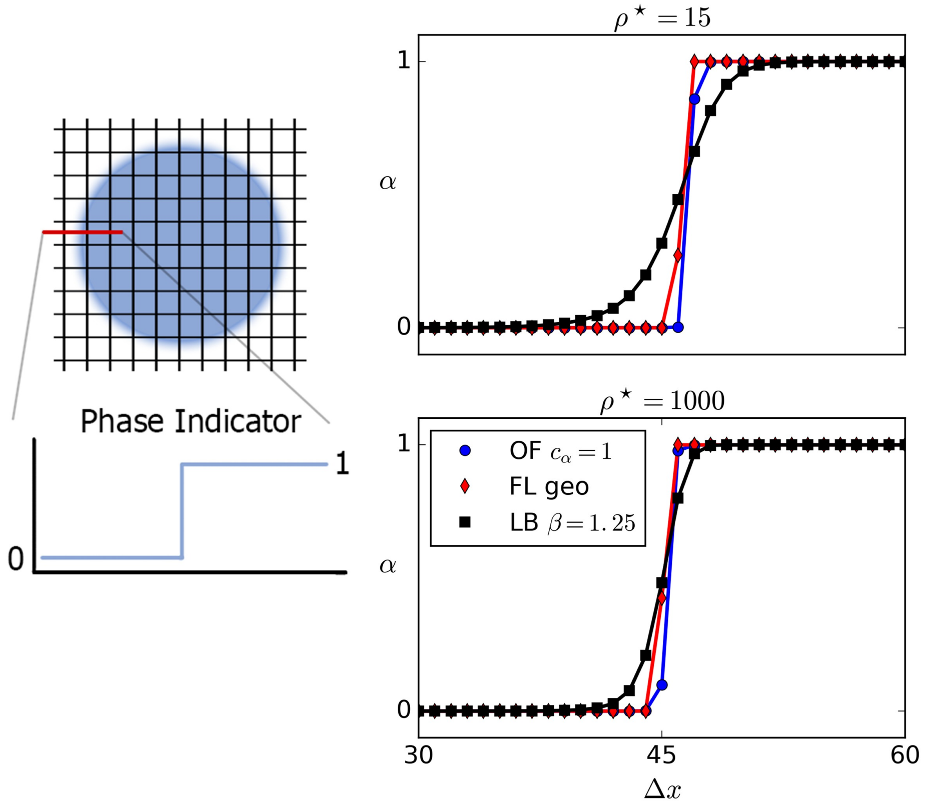


Figure 5: Comparison of the interface thickness at its sharpest in the stationary droplet simulations, shown as the phase indicator function α across one side of the droplet (refer schematic) for OpenFOAM VOF with interface compression ($c_\alpha = 1$), FLUENT with geometrical reconstruction and LB ($\beta = 1.25$). It is seen that FLUENT with geometrical reconstruction produces the sharpest interface, closely followed by OpenFOAM with interface compression. In LB, the interface becomes sharper with an increasing density ratio.

lower $\rho^\star$ values. Also, it has been shown that the interface thickness in LB is affected by the choice of the equation of state [63].

5. Falling droplets in liquid-liquid systems

5.1. Case formulation

Here onward, we focus on a realistic problem of a single liquid droplet, falling through a column of another immiscible liquid under gravity. We are interested in a comparison for low density ratio systems, which are highly relevant to liquid-liquid emulsion research. The cases are formulated by fixing the Eo and M numbers, which have been selected to coincide with a low Re and Eo region of the Clift map. First, a droplet is initialized in LB with an initial approximate diameter $d = 100\ \Delta x$, which is let to relax to an equilibrium value. Then, by using the Laplace law, σ is calculated. Using this value and Eo, g_{LB} is obtained. Finally, using the value of M, ν_{LB} is estimated which can be controlled by changing τ. Equivalent VOF simulations are formulated by keeping Eo and M identical to the LB simulations by varying σ and d, while keeping ρ_l, ρ_d and g constant, values of which have been listed in Table 1. Here onward the subscripts l, d denote the surrounding liquid and droplet liquid respectively. The five cases are listed in Table 2. Note that we perform these simulations for the lowest density ratio possible in our single-component PP-LB implementation, i.e. $\rho^\star = 3$, since droplets at even lower density ratios begin to dissolve under non-stationary conditions. This is because at very low density ratios, the interaction force between the phases becomes weaker in LB as phase separation is fragmented, and consequently the droplet dissolves away if it starts moving. It is worth mentioning that by choosing a low density ratio the spurious Reynolds number in VOF $\mathcal{O}(0.1)$ also remains much smaller in magnitude than the flow Reynolds number $\mathcal{O}(1-10)$.

Quantity	ρ_d	ρ_l	ν_d	ν_l	g
Units	[kg/m^3]	[kg/m^3]	[m^2/s]	[m^2/s]	[m/s^2]
Value	300.0	100.0	5.0×10^{-6}	5.0×10^{-6}	9.81

Table 1: Physical properties used in the VOF simulations.

Case	Eo	M	Re [22]	d [m]	σ [N/m]
1	0.98	1.75e-04	4	4.54e-04	4.12e-04
2	1.82	3.26e-04	8	5.57e-04	3.35e-04
3	2.18	5.29e-04	7	5.63e-04	2.85e-04
4	6.27	1.47e-03	13.5	8.05e-04	2.03e-04
5	3.50	9.50e-05	24	9.49e-04	5.05e-04

Table 2: Parameters for the five cases of a single liquid droplet falling through an immiscible liquid ($\rho^\star = 3$), along with the corresponding Re prediction by marking Eo and M on the Clift map.

5.2. Simulation details

The simulation domains with boundary conditions are shown in Fig. 6. The horizontal extent of our simulation domains is $8d \times 30d$ where d is the droplet diameter for the VOF simulations. This is considered sufficiently large such that the walls do not influence the velocity evolution [28, 64], given that what we wish to study is effectively a droplet freely falling in an infinite medium, i.e. to look at the droplet from a stationary frame of reference. The LB domain is larger at $10d \times 40d$, for two reasons. First is to ensure a large enough domain despite droplet expansion during the initialization phase. Secondly, our current LB implementation works with a fully periodic domain, while the VOF domains use free-slip horizontal boundaries and no-slip vertical boundaries. Since the cases we simulate have a strong left-right symmetry with no unsteady features in the droplet wake, the free-slip boundary condition should be equivalent to a periodic boundary condition in the horizontal direction (as the horizontal extent is large enough). To minimize the influence of the vertical boundary conditions being different in VOF compared to LB, the LB domain is kept large enough in the vertical direction such that the droplet reaches terminal velocity at a sufficient distance from the bottom periodic boundary. This ensures that the droplet does not cross the boundary and its wake is captured correctly in LB.

In OpenFOAM, the PISO algorithm has been used, while the PIMPLE algorithm with residual control was also tested. FLUENT uses the SIMPLE algorithm and only the tolerances for the final residual values have to be provided, which are kept the same as in OpenFOAM at a value of 10^{-7} for pressure and 10^{-8} for velocity to consistent. Here we have changed the tolerance from its default value in FLUENT (i.e. 10^{-5}, as was used in the spurious velocity study) to 10^{-8}, as an extra precaution to eliminate any differences that may arise if these values are different. The VOF simulations use second order time integration (*backward* scheme in OpenFOAM), with *vanLeer* discretization of convective terms and *linear* interpolation of gradient terms.

The parameters used in the LB simulations are presented in Table 3, and these simulations are performed using the open source Parallel Lattice Boltzmann Solver (Palabos-v1.5r1). The value of the reduced temperature $T_r = 0.96$, which gives a density ratio of $\rho^\star \approx 3$. The droplets are initialized with a diameter of 100 lattice units, with the density inside and outside set to 0.21 and 0.07, as obtained from the C-S coexistence curve at $T_r = 0.96$ [25]. This system is first allowed to relax to equilibrium and during this period, the droplet can expand from its original size where the magnitude of the expansion depends on the total mass in the system which needs to relax to equilibrium. Hence, this transient phase is dependent on the domain size, and if the droplet expands significantly, the horizontal and vertical extents that were chosen may be insufficient. Changing the value of β from $\beta = 1.25$ to $\beta = 1.65$ can help mitigate the droplet expansion and has been shown in the Appendix in Fig. B.17. Thereafter, d, g_{LB} and τ are calculated and the body force is added. We perform simulations for both values of β. Note that this is specifically a problem while simulating low density ratio (or T_r close to 1) systems with single component LB. This does not occur at higher density ratios ($T_r < 0.8$), where the phase separation is strong.

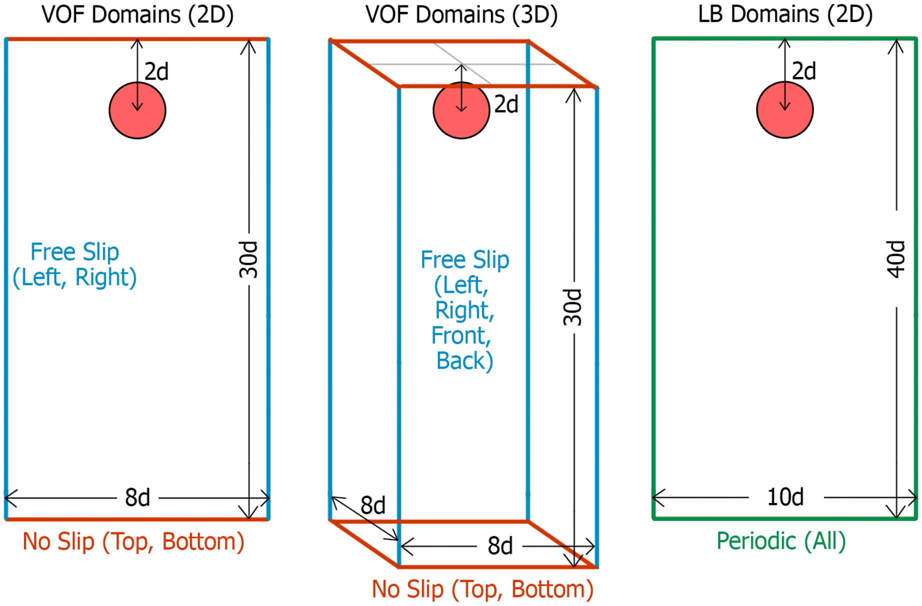


Figure 6: Schematic showing the computational domains and boundary conditions used in the VOF and LB simulations of a single liquid droplet falling through an immiscible liquid (with a density ratio $\rho^\star = 3$). Note that d is the droplet diameter in each case.

Case	1	2	3	4	5
$\beta = 1.25$					
τ	0.9129	0.8503	0.8779	0.8747	0.7578
$g[\times 10^{-7}]$	0.61463	1.1383	1.3634	3.9214	2.1951
$\beta = 1.65$					
τ	0.8958	0.8753	0.9049	0.9014	0.74716
$g[\times 10^{-7}]$	0.79861	1.1601	1.3896	3.9966	2.8522

Table 3: LB parameters for the falling droplet cases. For these simulations $T_r = 0.96$ and $\rho^\star = 3$. All simulations are performed on a domain $N_x \times N_y = 1000 \times 4000$ using the Parallel Lattice Boltzmann Solver (Palabos-v1.5r1).

Nomenclature

- OpenFOAM VOF simulations performed in 2D with mesh resolution of N cells per droplet diameter, $N \in \{20, 40, 80\}$, are labeled **OF2Ddx20** etc. For these simulations, we use $c_\alpha = 0.5$, to prevent interface diffusion in high *Re* cases, while not aggravating the spurious currents too strongly. Simulations using the VOFsmooth solver are suffixed with an 'S', Eg. OF2Ddx20S.

- OpenFOAM VOF simulations performed in 3D using dynamic mesh refinement are labeled as **OF3D**, with a suffix 'c' or 'm' for *coarse* or *medium* respectively, with 5 and 7.5 base cells per diameter and a factor 4 refinement at the interface. For our chosen simulation domain the number of grid cells is above a million for the 7.5 base cells with mesh refinement, which we do not refine further so as not to make the computational cost

- Fluent VOF simulations performed in 2D are named similar to their

OpenFOAM counterparts, as **FL2Ddx20** etc.

- LB simulations were performed using the open source Parallel Lattice Boltzmann Simulator (Palabos-v1.5r1) and are labeled **LB2D**.
- When not using a suffix ('dx20', 'm' etc.), we refer to results from the finest resolution simulations.

Before we present the results, it is worthwhile to mention that we compare 2D VOF and LB simulations, along with 3D VOF, to 3D experimental results from the Clift map. The 2D droplets can be thought of as infinite fluid cylinders, without any variation in the third direction. By balancing drag force with buoyancy, the terminal velocity can be expressed as $v_\infty = (\Delta\rho V g/0.5\rho_c A c_D)^{1/2}$, where $\Delta\rho$, V, ρ_c, A and c_D are the density difference, volume of the droplet, density of the continuous (outer) fluid, crossectional area and coefficient of drag respectively. For the Re values in this study, the drag coefficient c_D on a (solid) cylinder and sphere are approximately 4.5 and 7 (for Case 1, $Re = 4$) and 1.8 and 2.5 (for Case 5, $Re = 24$) respectively [65]. Using these values, it can be said that the 2D VOF results will predict a velocity 25% higher than the 3D value, particularly for Case 1. Apart from this, if the flow does not consist of highly 3D features like unsteady vortices, we expect the 2D simulations to be comparable to the 3D results. Note that the internal circulation inside a droplet reduces the drag experienced by it in comparison to a solid body by almost 30% [22], so the c_D values mentioned above do not directly apply to liquid droplets, and will need a further correction factor.

5.3. Results

The steady state Re numbers are shown together on a section of the Clift map in Fig. 7, where the solid black line marks the boundary between the spherical (below) and elliptical (above) shape regimes. Further, analytical solutions for the spherical and elliptical regimes (refer to Eq. 25 and Eq. 26) are shown, along with the final shapes from the OF2Ddx80 simulations for reference. This figure presents an overview of the simulations performed, also showing the small part of the falling droplet phase space that has been explored. The same is also shown in Fig. 8, where the cases are presented separately for additional clarity.

All techniques agree well with each other and the Clift map for Cases 4 and 5, while the spread between predictions is higher at lower Re. For VOF, this may be ascribed to a higher sensitivity of results to spurious velocities, as at low Re spurious currents may be comparable to the physical velocity. Further, the horizontal extent of the domain has a higher influence on lower Re droplets, while at higher Re values the domain edges have little influence.

We look more closely at Case 1 and Case 5, corresponding the smallest and the largest Re in this study.

Case 1

This corresponds to the spherical region of the Clift map. The steady state droplet shapes with the velocity and normalized z-vorticity fields ($-1 < \omega_z^\star < 1$, from red to blue) are shown in Fig. 9, the OF3D results are plotted on a cross-section at the center of the domain. Note that these plots show only a small region around the droplet to focus on the local flow characteristics. It was checked that the velocity further from the droplet, near the domain edges, is negligible. If the domain is not sufficiently large in the

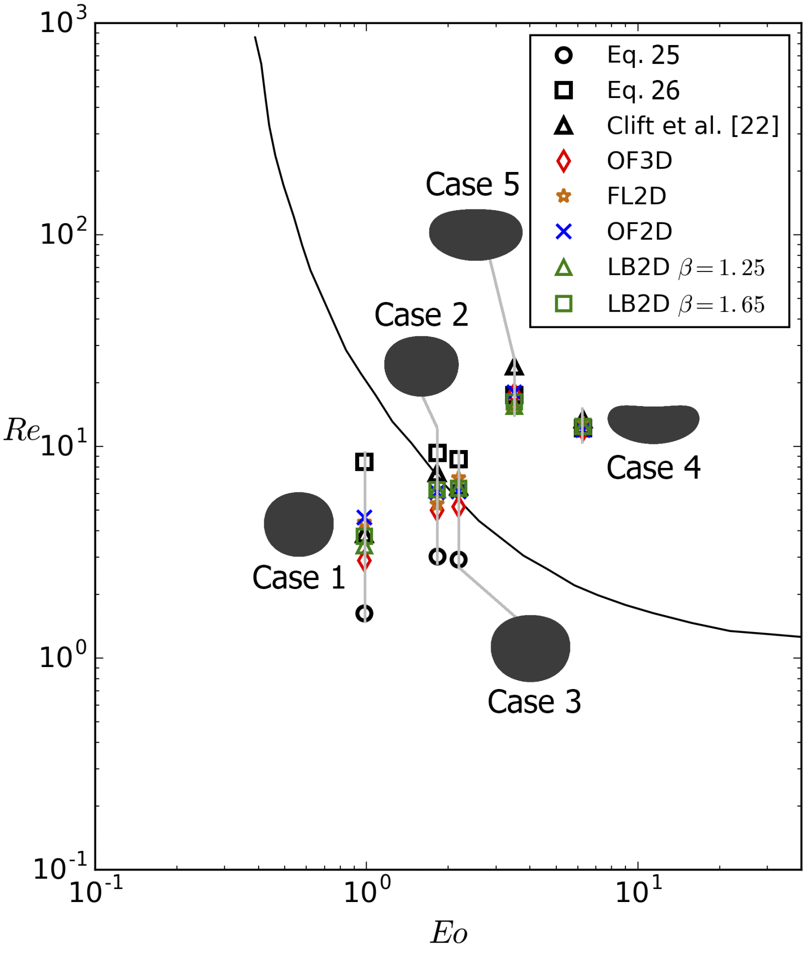


Figure 7: Terminal Re numbers for single liquid droplets falling through an immiscible liquid ($\rho^\star = 3$), calculated from simulations with the finest grid resolution shown with the Eo number along with the final shape from the OF2Ddx80 simulations. Results for individual cases (i.e. a fixed Eo and M number from Table 2) are connected with a vertical gray line, which also forks to the droplet shape for Cases 2, 3 and 5 for clarity. Lastly, the solid black curve divides the $Re - Eo$ phase-space into the spherical (below the line) and ellipsoidal (above) shape regions, and is taken from Clift et al. [22].

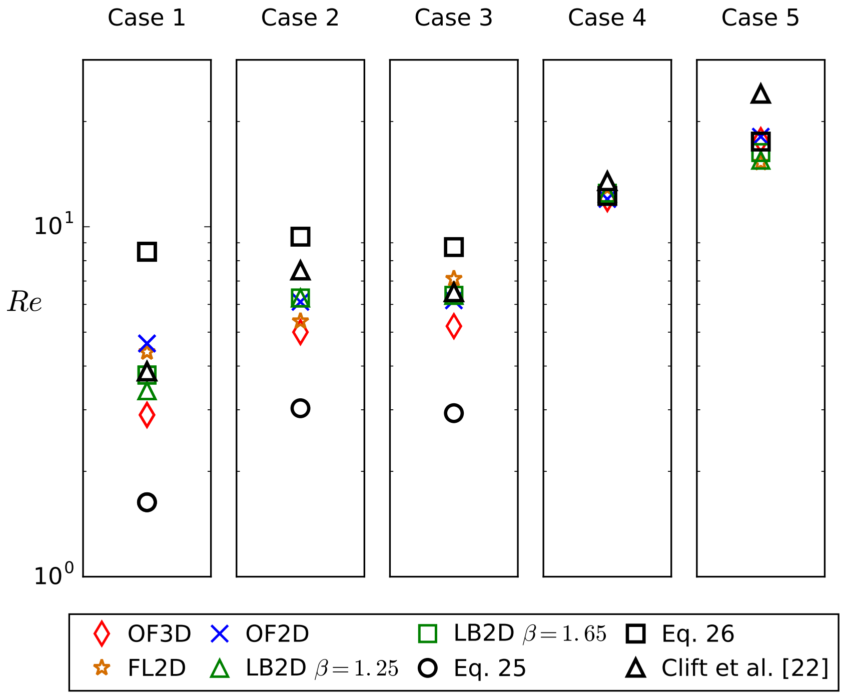


Figure 8: Terminal *Re* numbers from the different simulations shown separately for each Case listed in Table 2 for additional clarity. The spread between the predictions is larger at lower *Re* numbers, while at higher *Re* values all simulations agree well with each other and the Clift map. There is excellent agreement between VOF and LB results.

horizontal direction, the velocities near the lateral walls can be significant and should be taken into account to study the falling droplet from a stationary frame of reference. The simulations produce a trailing vorticity field at steady state, with the highest concentrations outside the lower droplet region, close to the interface, clearly marking two counter rotating regions of the flow. The shapes produced are also very similar and the droplets remain mostly spherical for all solvers.

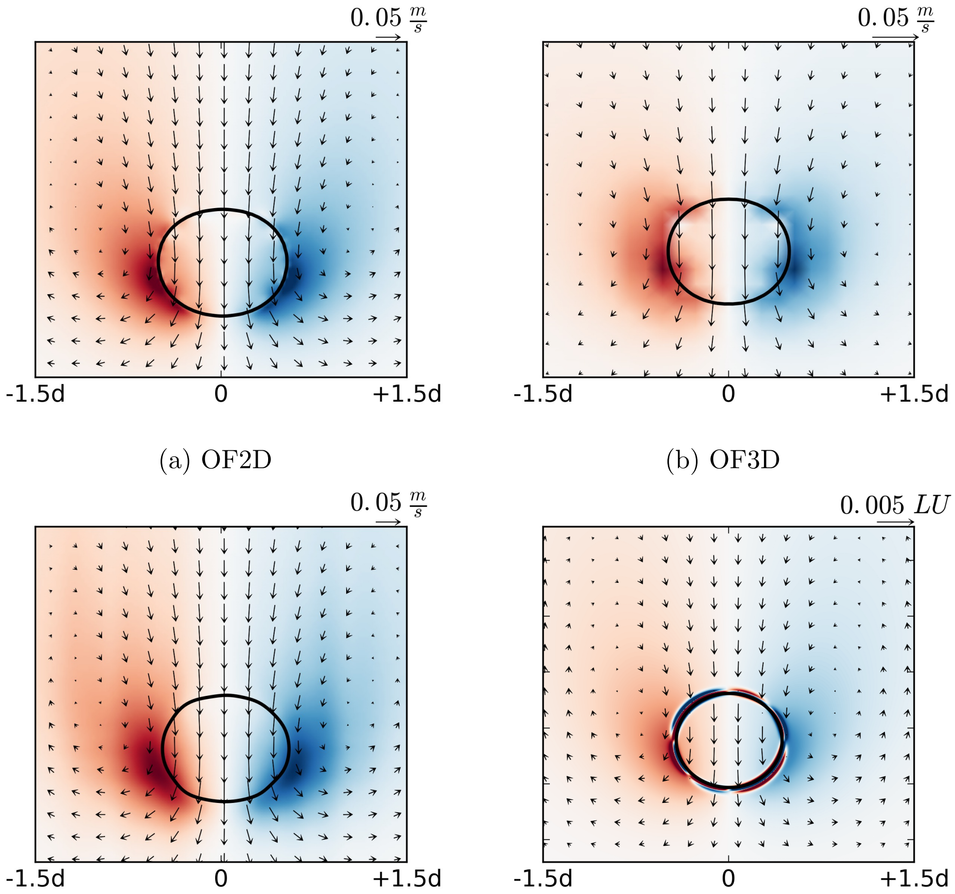
The vorticity field in LB is less intense than in VOF, but its features are similar. Some spurious vorticity regions can be seen localized near the interface which do not diffuse into the bulk. The spurious vorticity produced in Palabos is slightly different from the spurious vorticity in the LB simulations using our in-house code (Fig. 2d). This difference comes from the exact discretization of the force term in the LB equation, the effect of which has been detailed in Connington and Lee [66]. What is important here is that the spurious currents do not influence our results directly or significantly as they are symmetric and tend to cancel out. They might have an effect on the droplet shape, suppressing acute deformations, but we do not investigated that in this study.

The evolution of the Reynolds Re number of the droplet has been shown in Fig. 10. For calculating Re, the average velocity in the droplet region is considered as follows

$$u = \frac{\sum \alpha u_y}{\sum \alpha} \tag{29}$$

where $\alpha = 1$ in the droplet region and 0 outside it and u_y is the magnitude of the vertical velocity component. The evolution is shown over dimensionless time $t^\star = t/\sqrt{d/g}$ and the gray band marks the value from the Clift map with 5% uncertainty.

The 2D VOF simulations attain very similar terminal Re values, close to the 3D value extrapolated from the Clift map. The uncertainty in the results, given the magnitude of spurious Re, is expected to be around $2-20\%$ for the coarse to fine mesh resolutions. Since the magnitude of spurious velocities increases from $\mathcal{O}(0.01)$ to $\mathcal{O}(1)$ from dx20 to dx80, mesh convergence of the results cannot be achieved for this case. This behaviour of OpenFOAM VOF has been observed before for flows with a low Capillary number [60]. The LB2D results are very close to the Clift map for both values of β, and the spurious currents in LB have very little influence on the droplet



(a) OF2D

(b) OF3D

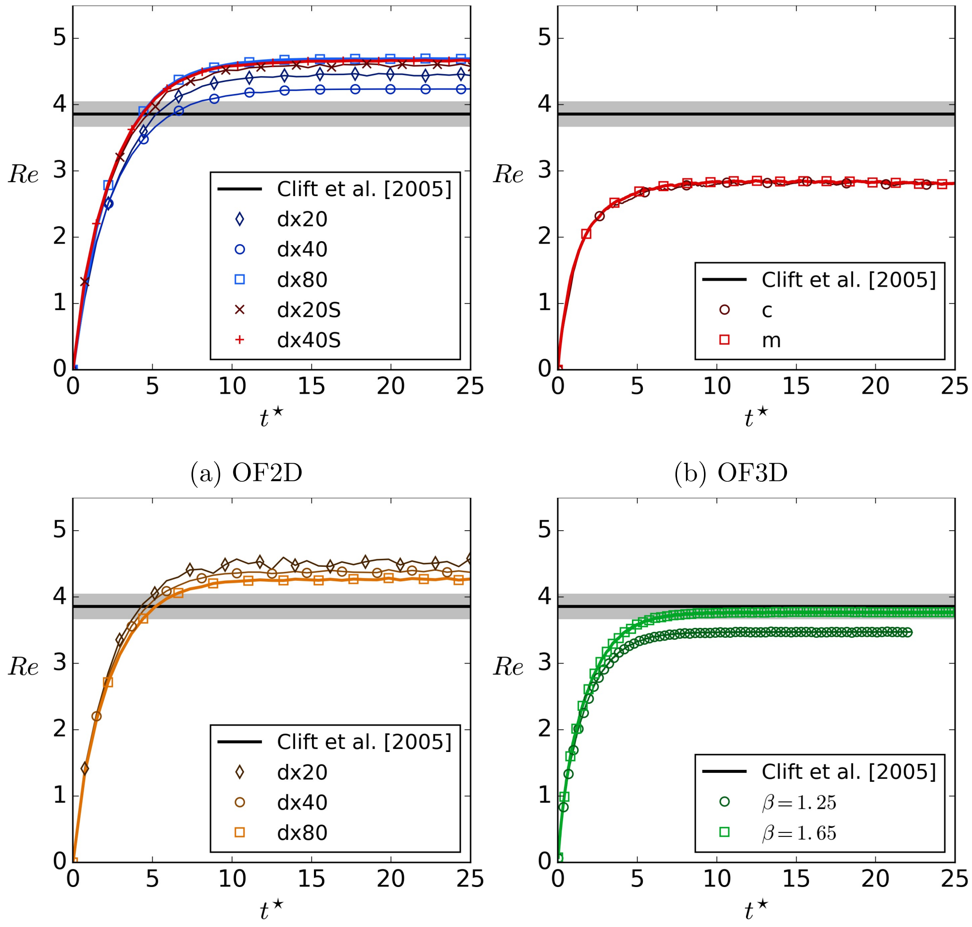
(c) FL2D

(d) LB2D ($\beta = 1.65$)

Figure 9: Steady state droplet shape with the normalized vorticity field $\omega_z^\star$ ($-1 < \omega_z^\star < 1$, red to blue, where $\omega_z^\star = \omega_z / |\omega_z|_{\max}$) is shown for Case 1, corresponding to $Re = 4$. Note that the region shown is a small part of the actual computational domain around the droplet.

Re evolution. A comparison between the steady state shape for the two β values is shown in Fig. B.19 .

OF3D under-predicts the droplet *Re* almost by $\sim 30\%$. We investigated this further by performing tests on a smaller domain to reduce computational



(a) OF2D

(b) OF3D

(c) FL2D

(d) LB2D

Figure 10: Evolution of the *Re* number for Case 1 for all simulations shown along with the value from the Clift map with 5% errorbars.

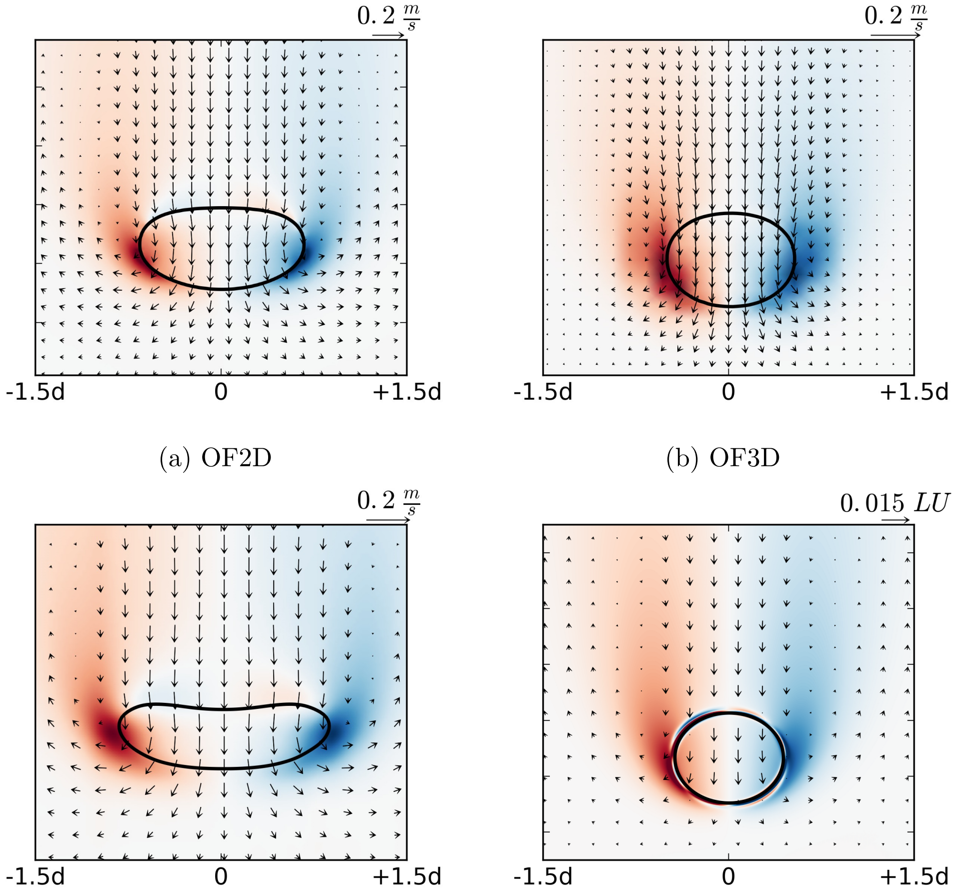
time for a representative problem. We find that compared to a uniform 3D mesh (of grid resolution equal to the finest grid size obtained upon 2 successive dynamic mesh refinements), the dynamic mesh simulations with a coarse base mesh always under-predict the velocity. This can be ascribed to a loss in mesh orthogonality in the transition between coarse and refined mesh regions. The interpolation of values from cell center to cell faces occurs along a line that is not orthogonal to the face itself. This leads to numerical errors and can cause a different kind of spurious currents than those arising out of inaccurate curvature calculation in VOF. Further due to refinement near the interface, curvature induced spurious currents are also aggravated, and it has been reasoned by Magnini et al. [67] that these can be seen as capillary waves on the interface, which effectively cause a deviation from a smooth interface profile resembling a kind of surface roughness that increases drag - something we observe in the consistent underprediction of the velocity. Upon adding additional refinement criterion, for instance refinement in the droplet wake based upon a vorticity threshold, the results improve though they still do not coincide with results from the uniform 3D mesh.

Case 5

This case corresponds to the ellipsoidal regime of the Clift map. The final droplet shapes along with normalized vorticity fields are shown in Fig. 11. It is seen that compared to the 2D VOF cases, the OF3D droplet undergoes lesser deformation, which is a consequence of the under resolved velocity field owing to a coarse base mesh. The droplet deformation is caused due to the pressure difference between high and low pressure regions around the droplet. A region of strong vorticity corresponds to lower pressure, creating a larger deforming force. Due to the lower velocity in OF3D, this pressure difference is lower, resulting in lesser deformation. The LB2D droplet does not flatten as much as the 2D VOF droplets, and in this case, the droplet shrinks slightly from its initial size. Although we achieve a steady state here, for even higher *Re* numbers, moving droplets in LB can begin to dissolve away completely due to the diffused interface at low density ratios. Further, it is seen that the vorticity field in LB remains confined in the lighter phase, whereas in VOF, the vorticity field is very close to the interface and also extends to within a small region inside the droplet.

The evolution of Re numbers has been shown in Fig. 12. All simulations predict similar terminal Re values, though all fall below the prediction of the Clift map. Here the OF2D results were found to be within 10% of each other, as the spurious velocities have a lower influence on the results. The terminal Re in LB for both $\beta = 1.25$ and 1.65 is smaller than the OF2D value, but comparable to the FL2D values. It was observed (not shown here) that on smaller periodic domains in LB, the droplet does not reach terminal velocity as the droplet wake becomes comparable to the domain length. Using a large enough vertical extent hence is indispensable for accurate predictions. The fact that all the simulations predict a velocity lower than the Clift map value ($Re = 24$) also might indicate that the bubble-droplet equivalence breaks down at higher Re numbers.

There are a few things worth noting regarding the results from the LB simulations. By increasing the value of β from 1.25 to 1.65, the density ratio obtained in LB deviates from the analytical solution of the selected EOS (see Fig. B.16). Since PP-LB follows the selected EOS, large deviations from the EOS lead to unstable simulations. Therefore, the maximum value of β is limited due to the stability condition of the model. A stability analysis shows that the simulations become unstable when g_{LB} is larger than $\mathcal{O}(10^{-5})$. This places constraints on the droplet diameter that is required for simulating a particular Re. Also since the interface width is relatively large at $\rho^{\star} = 3$ ($\approx$ 10 lattice units), at higher Re values it can further diffuse adding to numerical inaccuracy. This can be remedied to an extent by using a larger droplet diameter, which in turn makes the domain size computationally



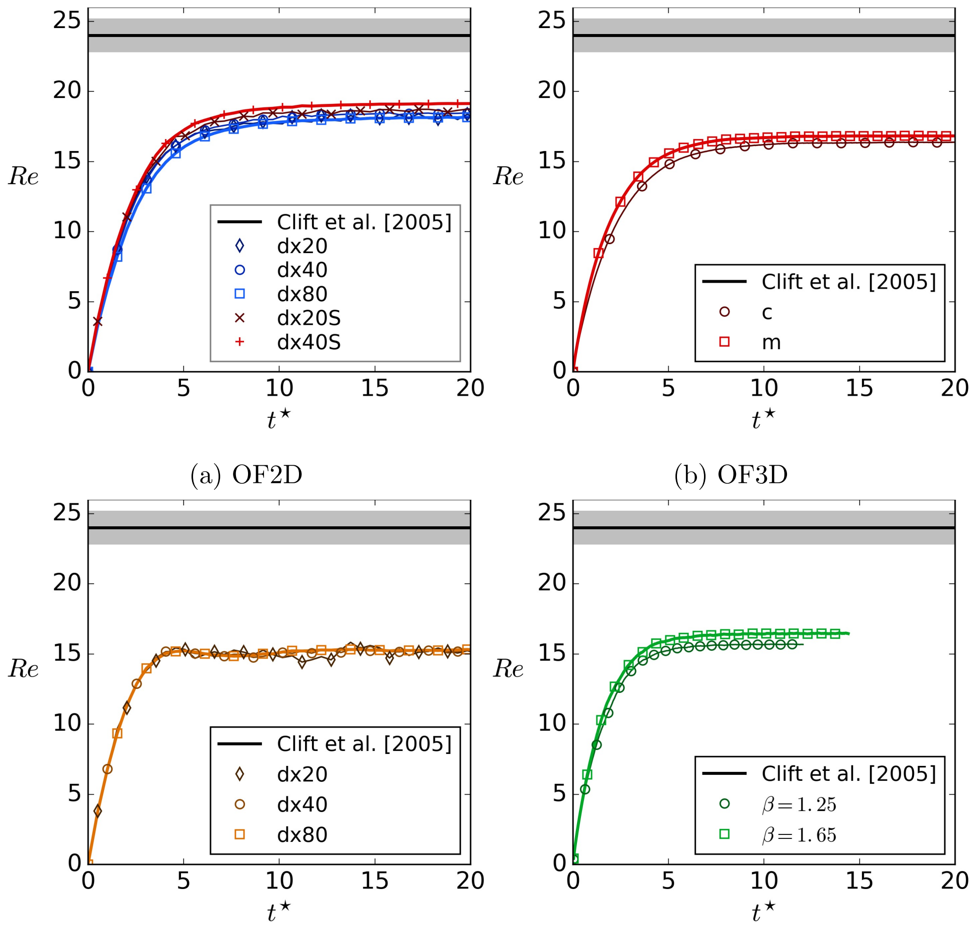
(a) OF2D

(b) OF3D

(c) FL2D

(d) LB2D ($\beta = 1.65$)

Figure 11: Steady state droplet shape with the normalized vorticity field $\omega_z^\star$ ($-1 < \omega_z^\star < 1$, red to blue, where $\omega_z^\star = \omega_z / |\omega_z|_{\max}$) is shown for Case 5, corresponding to $Re = 24$. The FL2D droplets is seen to deform the most, followed by OF2D. OF3D produces a lower deformation of the droplet, which is ascribed to an under-resolved velocity leading to a lower pressure difference between the horizontal and vertical droplet extents which acts towards deforming the droplet. The LB2D droplet does not flatten as much as the 2D VOF droplets and remains mostly spherical, and its size shrinks slightly from the initial value.



(a) OF2D

(b) OF3D

(c) FL2D

(d) LB2D

Figure 12: Evolution of Re for Case 5 for all simulations shown along with the Clift map value with 5% errorbars.

expensive.

Lastly, we compare the deformation of the droplet which can be expressed as the standard Taylor deformation parameter $D = (L-B)/(L+B)$, where L and B are the lengths of the major and minor axes of the deformed droplet. Fig. 13 shows D at steady state for the different cases using simulation results from the finest mesh.

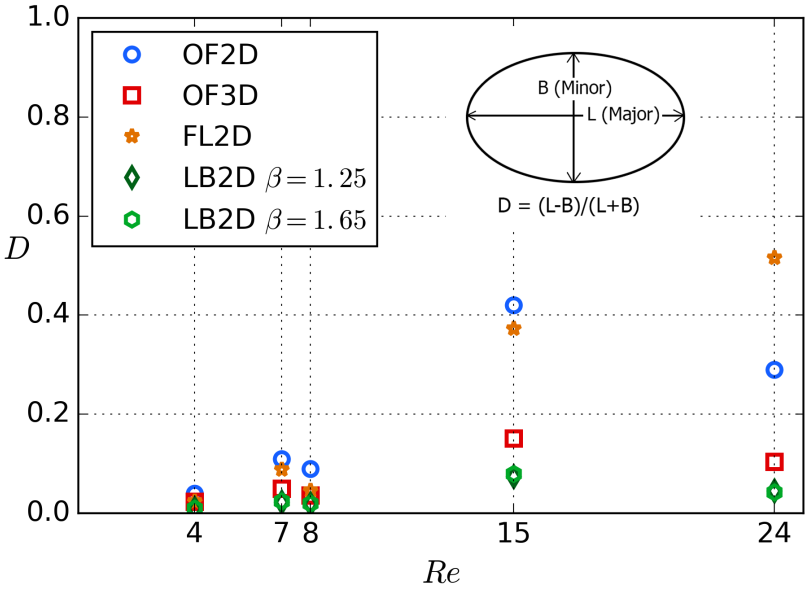


Figure 13: Taylor deformation parameter D at steady state for all cases.

For the cases $Re \in \{4, 7, 8\}$, the VOF simulations predict a small value of D, i.e. the droplets deform very little from their initial spherical shapes. OF3D predicts a low D for all Re numbers, which is ascribed to the under-resolved vorticity field. FL2D and OF2D predict higher deformations for $Re \in \{15, 24\}$. OF2D predicts the highest deformation for Case 4 ($Re = 15$), i.e. the case corresponding to the highest Eo. LB2D consistently predicts a low D value across all Re values. Though the Re prediction with LB turns out to be accurate, our implementation seems to be incapable of capturing droplet deformation more accurately.

6. Discussion

It is widely held that no single flow simulation technique is universally applicable, primarily because each technique may have been developed with the view of studying a particular class of flow problems. However, it is generally also sought that a newly proposed model or method be as universal as possible, i.e. to say it should be robust. Upon comparing VOF and PP-LB, we encountered several challenges unique to each method in trying to formulate a problem that can be commonly studied. We believe these points must not be overlooked, so as to better equip users with an understanding of the limitations they might face upon choosing either method. We also indicate flow regimes where either method may be better suited, along with method specific issues that have to be dealt with. Note that the following discussion draws from our investigation of low density ratio flows which are widely encountered in emulsion research, ranging from oil and gas applications to food processing and biotechnology. The same methods when applied to high density ratio multiphase flows might be faced with challenges distinct to a different flow regime, so we exercise caution while extrapolating the possible outcomes of doing so, and refrain from making overly generalized statements.

Regarding VOF

The VOF method can be considered quite robust in that it can handle a wide range of free surface flows, without constraints on the density and viscosity ratio between the fluids or the flow Reynolds number, though extreme values will pose difficulties. As surface tension is independent of density ratio in VOF, very low density ratios like $\rho^{\star} \sim 1.1$ (and lower) can also be simulated as the phase fraction formulation of two fluids ensures strong immiscibility.

The computational cost in VOF increases rapidly upon using successively finer grids, particularly due to the iterative pressure and velocity solution procedure of the finite volume method, with an added transport equation for the phase indicator. Modifications like dynamic mesh refinement hence become indispensable when simulating fully three dimensional flows, but such techniques may also influence the results adversely - for instance the consistent under-prediction of terminal velocity reported in this paper, and the additional numerical errors in interpolation of values between coarse and refined mesh regions.

For high density ratio systems at low Reynolds numbers (like falling rain droplets, or very small air bubbles in water), spurious currents can become comparable to the physical velocity scale when using the CSF implementation for the surface tension force, and classical VOF might fail to predict accurate results. Modifications like smoothing of the phase indicator field before computation of curvature, the use of body fitted coordinates or an improved surface tension force implementation will become necessary while simulating these flow regimes.

Another limitation of VOF is the time stepping method, as fully implicit schemes might not be available for transient flow solvers, as is the case with the VOF implementations in FLUENT and OpenFOAM. A crucial aspect to simulation accuracy is the residual control algorithm. In OpenFOAM, the PISO algorithm iteratively solves for velocity and pressure (*inner* correction loops), while the VOF field α is solved once per time step (at the beginning of the PISO loop). Contrarily, upon using PIMPLE (with *outer* and *inner* correction loops), the α field is corrected with ever outer loop as well. With a small enough time step and sufficient number of PISO loops, the results from PISO and PIMPLE can be similar. However, we found that some falling droplet (or rising bubble) simulations might not converge at all when using the PIMPLE algorithm, whereby PISO is the only option, and it is generally more stable and faster than the PIMPLE algorithm. The values of the final and relative tolerances of the residuals and under relaxation factors should be carefully tested and chosen, and these can vary with the problem being simulated. In commercial software like FLUENT, the user only prescribes a final convergence criteria, which makes it easier to run simulations at the loss of full control over the solution algorithm.

Regarding single component PP-LB

LB is remarkably faster than finite volume based solvers (roughly 10 times when running both on a single processor for an identical problem) mainly because an iterative Poisson solver is not required, which is time consuming. Further, all computations are highly localized in classic PP-LB (i.e. when considering only nearest neighbour interactions), the advantage of this becomes immediately apparent upon parallelization of the code. The Palabos simulations on 4 million grid cells, for around 1 million iterations, were run on 24 processors and took approximately 24 hours of wall-clock time, while the finest resolution VOF simulations on 1.5 million grid cells run on 16 processors took around 10 days.

Density ratio in single component PP-LB is dependent on T_r, and simulating flows with $T_r \approx 1$ such that $\rho^\star \approx 1$ is not possible as the interface width tends to infinity when $T_r = 1$, which sets a minimum achievable density ratio. For this reason, using single-component multiphase PP-LB to simulate two immiscible liquids might not always be suitable. In this formulation, the two liquids are essentially different thermodynamic phases of the same fluid, co-existing at a prescribed sub-critical temperature (i.e. $T_r < 1$). When the density ratio is low (as in this study), the two phases are separated by a relatively thick interface and they remain miscible particularly under dynamic conditions. Owing to this, phase volumes may change over time - an artifact being small regions of the low density phase dissolving away when the droplet diameter and interface width become comparable. Note that the total mass of the fluid is conserved, only its distribution between the two phases changes. Using a multi-component formulation can help ensure stronger immiscibility between two fluid components by employing a repulsive interaction, which can also help tune surface tension more accurately.

The current LB implementation has several other limitations, for instance a minimum bound on the value of the relaxation time, where values below $\tau = 0.55$ lead to unphysical droplet expansion and wobbling, which gave inaccurate results. The value of τ also depends on the droplet size and the density ratio [25]. Increasing surface tension in the system can enable simulations at higher values of τ, however that increases the required g_{LB} and terminal velocity which should be kept in mind while modeling such a system. The value of g_{LB} is closely tied to the droplet diameter, and it increases for smaller droplets. With a stability analysis we find that simulations with $g_{\mathrm{LB}} > \mathcal{O}(10^{-5})$ become unstable, which further limits the smallest droplet size achievable - the computational domain consequently becomes large. The LB method is sensitive to the value of Re and Eo, and at relatively higher values unphysical droplet expansion or contraction is observed. With these considerations, our LB implementation was limited to low Re and low Eo cases, and several modifications like the incorporation of longer range interaction forces [62] and choosing advanced collision operators such as multiple relaxation time (MRT) [73, 74] might be necessary, along with extensive testing of parameters, to make single-component PP-LB applicable to more complex flow problems, like those involving high density and viscosity ratios between the liquids, or high Re problems.

A general concern in LB is the maximum value of the velocity in lattice units, which should be sufficiently smaller than the lattice speed of sound (i.e. the low Mach number limit of $Ma < 0.2$). The speed of sound for the LB-EOS is $c_s = 1/\sqrt{3}$, or more generally for an arbitrary EOS $c_s^2 = \partial p/\partial \rho$. This ensures that the incompressible Navier-Stokes equations are obeyed. The error due to a non-zero Ma has been shown to be $\mathcal{O}(Ma^3)$ [75]. Lastly, it has been shown that the discrete kinetic version of the BGK equation, and notably LBM, provide semi-quantitative results [76]. Therefore, a percentage of error can be expected in classical LB modeling, the bounds of which depend upon the flow regime being simulated.

7. Conclusions

Multiphase flow simulations have become common tools being applied to a wide range of processes dealing with immiscible fluids. Notwithstanding, comparisons between different simulation techniques are rare and often qualitative. We have presented a quantitative comparison between an implementation of the single-component multiphase pseudopotential lattice Boltzmann method (PP-LB), which is a widely used mesoscopic technique, and the finite volume based Volume of Fluid (VOF) method. By simulating stationary and falling droplets, we compare how the methods fare for different aspects relevant to multiphase flows with a low density ratio - a regime crucial to emulsion research. The main findings are listed below.

1. A modified implementation of the LB method with the so-called $\beta-$scheme produces spurious currents $1-3$ orders of magnitude lower than all VOF implementations. This can be relevant when simulating problems involving evaporation or heat and mass transfer across interfaces, or high density ratio flows at low Re. Among the VOF methods, interface compression aggravates spurious currents while artificially smoothing the interface during curvature calculation reduces them. In VOF, spurious currents also increase with mesh refinement, and if the physical velocity and spurious velocity become comparable (for instance in our $Re = 4$ case), mesh converged results cannot be obtained.
2. In terms of computational time, LB simulations are faster than their VOF counterparts by an order of magnitude (when running comparable problems, i.e. same Re, Eo, $Nx \times Ny$ etc). The iterative Poisson solver inherent to finite volume methods makes VOF slower in comparison. LB, on the other hand, has more demanding memory requirements as it needs to store more variables per lattice node as compared to VOF (since the density f_i in LB is directional with 9 components for a D2Q9 lattice, 27 for D3Q27 etc, at every lattice node).
3. In comparing interface thickness, the FLUENT VOF with geometrical reconstruction produces the sharpest interface of $2-3$ grid cells, closely followed by OpenFOAM with interface compression ($c_\alpha = 1$). The interface width in LB remains diffuse over $5-10$ lattice units depending on the density ratio (qualitatively varying inversely).
4. The falling droplet simulations show that despite an identical case setup, the 2D results from FLUENT and OpenFOAM tend to vary

slightly. For the $Re = 24$ case, FLUENT produces a more flattened droplet with higher deformation than OpenFOAM. The differences in solutions between OpenFOAM and FLUENT derive from the different treatment of the interface (compressive scheme against an explicit geometrical reconstruction), the difference in the solution algorithm (PISO against SIMPLE) and the exact accuracy control methods. What is striking is that upon simulating a widely studied, *purely numerical* rising bubble benchmark proposed by Hysing et al. [77], OpenFOAM and FLUENT VOF results are found to agree well within 2% (refer to Appendix A). These results were also less sensitive to the choice of discretization schemes. Hence, agreement with a numerical benchmark alone cannot be considered sufficient proof for the reliability of a solver when applied to physical problems, which calls for additional realistic benchmark studies.

5. The 3D VOF simulations using dynamic mesh refinement in OpenFOAM predict a lower terminal Re number, particularly for the $Re < 10$ cases. This is because a spherical droplet at a low flow rate experiences more drag in comparison to the 2D droplets which can be effectively considered as infinite cylinders which have a lower drag coefficient. The results are also aggravated by the coarse mesh resolution in 3D (7.5 base cells per diameter), to keep the computation cost modest. The velocity field is hence under-resolved, which also prevents the droplets from deforming for the higher Re cases as much as the 2D counterparts. More generally, results from simulations with dynamic mesh refinement have a degree of inaccuracy compared to a uniform orthogonal 3D mesh with the same resolution as the finest grid refined cell. This is due to two reasons - first the interpolation errors due to

lack of orthogonality in the transition between coarse and refined mesh regions. Secondly, spurious currents are aggravated in the fine mesh region surrounding the droplet interface. These can be interpreted as capillary waves that increase the drag on the droplet by causing deviations from a smooth interface profile.

6. In LB, the droplet expands during the transient initialization phase when the densities relax to equilibrium. Several factors contribute to this, including the domain size (which corresponds to the amount of mass in the domain, refer to Fig. B.17 in the Appendix), the strength of the pseudopotential force (given by β), and the system parameters themselves (T_r and $\rho^\star$). This is because at lower density ratios (T_r close to 1) the droplet interface is quite diffused as the phase separation is weak. This can be remedied to an extent by increasing the strength of the pseudopotential force (refer to Appendix B, Fig. B.19), also the domain size should be large enough to account for these effects. With these measures, LB is in very good agreement with VOF and the Clift map values in predicting Re. However, LB consistently underpredicts the droplet deformation, even at higher Eo values. This indicates that our current PP-LB implementation is limited to flows of low Eo and Re numbers, and several modifications to the technique are required to make it more generic and robust as compared to VOF. Lastly, while simulating dynamic multiphase problems at low density ratios using *single component* PP-LB, several precautions should be taken depending on the flow regime, as the two thermodynamic phases remain miscible.

We observe that formulating a multiphase flow problem that can be commonly studied by PP-LB and VOF is a non trivial task due to several fundamentally different aspects of the two methods. The translation of parameters from one method to the other to keep the simulations comparable can be meticulous. Also, issues particular to each method can narrow down the operable phase space of the chosen problem considerably as we have elaborated in our Discussion section. Despite precautions and careful case formulation, we see that different numerical techniques produce different results - since the exact numerics between methods is not identical. This urges us to strongly state that such comparisons are important particularly for the formulation of newer-generation benchmark studies that detail the strengths and limitations of the increasing number of novel simulation techniques being proposed in literature.

Acknowledgments

The research leading to these results has been done in cooperation with the Dutch Institute for Sustainable Process Technology (ISPT) as part of the LBM project, the Netherlands Organisation for Scientific Research (NWO) and Shell, the Netherlands. The authors would like to thank Dr. Orest Shardt (University of Limerick, Ireland) for the useful discussions and insight regarding the work.

Appendix A. Comparison between OpenFOAM and FLUENT

We saw from our falling droplet comparison that VOF results from OpenFOAM and FLUENT are not identical. The final velocity from both methods for Case 1 ($Re = 4$) and Case 5 ($Re = 24$) are within 15%. For Case 5, the velocity evolution profiles have slightly different features (for instance an overshoot for FL2D around $t^\star = 5$, refer Fig. 12), and the final shape in FLUENT is more deformed than in OpenFOAM. The major difference between the two solvers is the treatment of the interface, where explicit geometrical reconstruction and an implicit compressive scheme are conceptually and numerically very different . It should be noted that the solution algorithms SIMPLE and PISO are not equivalent as well, and may also lead to differences in the solution. To investigate this further, we simulated a well established numerical benchmark for rising bubbles [77] where various multiphase techniques are tested for predicting bubble shape and velocity for two *purely numerical* 2D test cases. These are namely TC1 and TC2, formulated using parameters listed in Table A.4. We performed the simulations for the finest mesh adopted in the study, i.e. 160 cells per bubble diameter and only show the velocity evolution here (along with the final bubble shape for reference). For comparison, we take the Hysing et al. [77] results as benchmark solutions, along with the results obtained by Klostermann et al. [60] who performed the same benchmark study using OpenFOAM to test the compressive VOF scheme.

The numerical schemes used in OpenFOAM are *backward* time integration, *vanLeer* discretization for velocity and α advection, and *linear* interpolation of gradient terms. Other parameters are kept the same as in Klostermann et al. [60]. The FLUENT simulations were performed with the geometrical reconstruction technique, and the schemes are the same as in Section 2.3.

Fig. A.14 shows the velocity evolution of the bubble over time for TC1, where the FLUENT and OpenFOAM results are very close to each other (also, we are able to reproduce the points of Klostermann et al. [60] identically). Fig. A.15 shows the velocity evolution for TC2 and again Open-

	ρ_1	ρ_2	μ_1	μ_2	g	σ	Re	Eo
TC1	1000	100	10	1	0.98	24.5	35	10
TC2	1000	1	10	0.1	0.98	1.96	35	125

Table A.4: Physical properties of the two test cases formualted by Hysing et al. [77], for other details we refer the reader to the paper.

FOAM and FLUENT results are seen to be very close together until $t = 2$ [s], whereafter they differ slightly (by $\approx 1.5\%$).

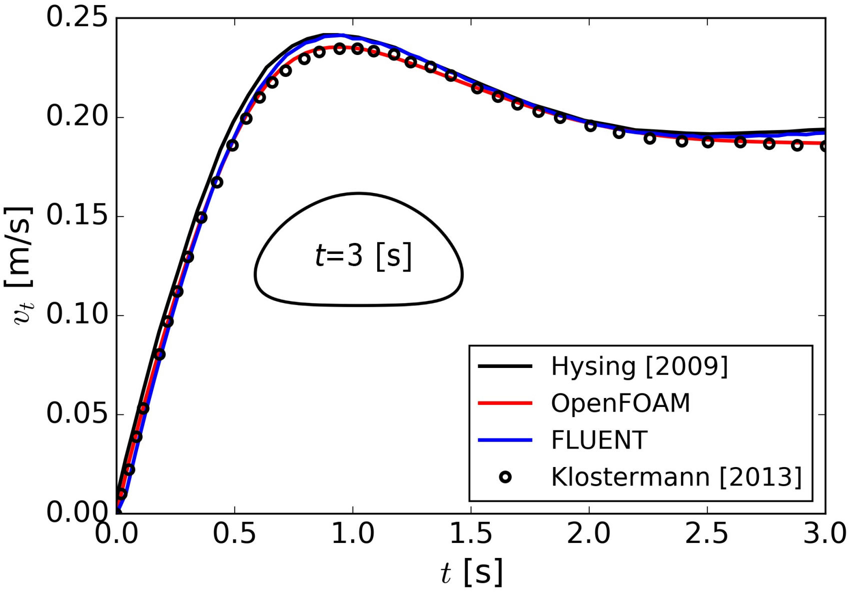


Figure A.14: Velocity evolution for the bubble for TC1 of Hysing et al. [77]. The OpenFOAM and FLUENT VOF solutions are seen to be very close, and our OpenFOAM points coincide precisely with Klostermann et al. [60].

It appears that the VOF techniques in OpenFOAM and FLUENT agree quite well when reproducing this numerical benchmark. However, as shown

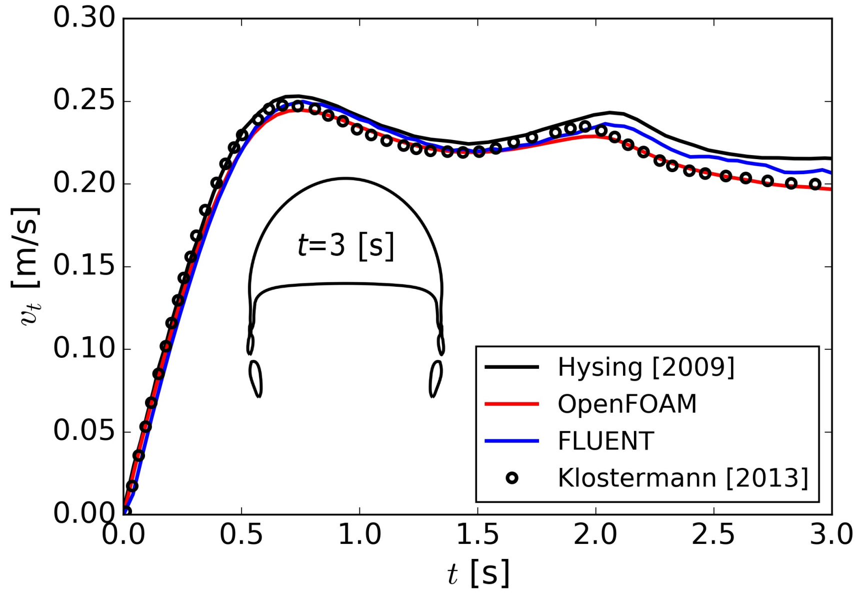


Figure A.15: Velocity evolution for the bubble for TC2 of Hysing et al. [77]. The OpenFOAM and FLUENT VOF solutions are seen to be very close until $t = 2$ [s], after which they differ by roughly 1.5%, and again our OpenFOAM data points coincide precisely with Klostermann et al. [60].

in this paper, when applied to a more generic physical problem (like the falling droplets), the transient as well as long time behaviour can be different. Though widely used to test multiphase techniques, the Hysing et al. [77] benchmark employs a very small domain of $2d \times 4d$, whereby the walls are very close to the edge of the bubble. This might have an influence on the bubble rising characteristics, as generally the horizontal domain width for rising bubbles is taken to be $8d - 12d$ depending on the Re, whereas the height can be anywhere between $12d - 30d$ depending on approach to steady state. This goes on to show the need for more realistic numerical benchmarks to be able to comment on the reliability of different simulation techniques -

something we have attempted to do in a small way.

Appendix B. Influence of β on droplet shape in LB

Fig. B.16 shows a comparison between the analytical coexistence curves for the C-S EOS and numerical results obtained with the proposed method with $\beta = 1.25$ & 1.65. It is clear that the numerical results agree well with the theoretical data over a wide range of reduced temperatures (or density ratios).

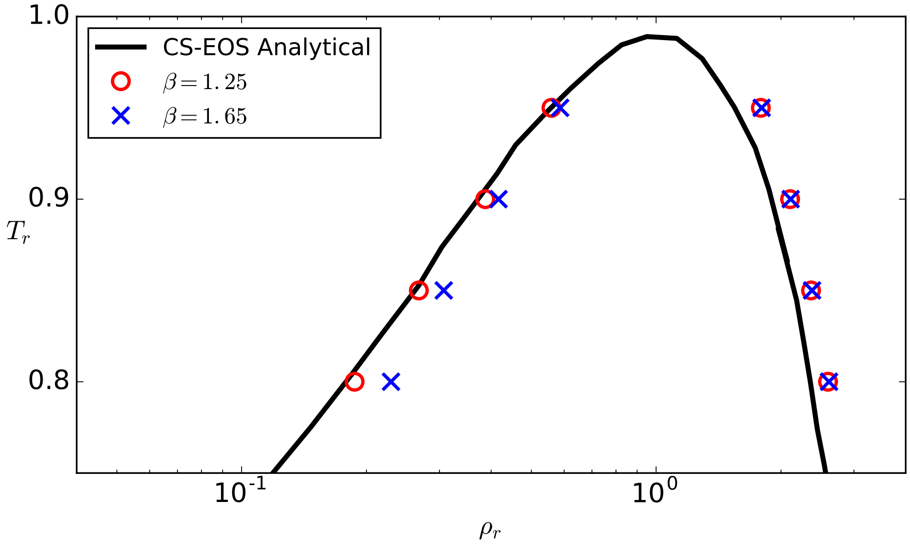


Figure B.16: Liquid and vapor reduced density variation with reduced temperature for C-S EOS. The analytical solution is obtained from the Maxwell-rule of equal areas.

However, as discussed in the text, the LB droplets at low density ratio can undergo unphysical expansion after initialization during a small transient period when the densities relax to equilibrium. In dynamic conditions, there can be further expansion or contraction, the reason for this behaviour is that the interaction force between the phases is weak and does not constrain the initial droplet volume as it begins to falls. Further, we found the amount of expansion also depends on the total mass in the system, which depends on the domain size. This has been shown in Fig. B.17, where a droplet is initialized with a diameter of 100 lattice units with increasing domain sizes ($N_x = N_y$), for the $\rho^\star \approx 3$ system considered in this study. The system is let to relax (up to 20000 iterations), after which an equivalent diameter is calculated as d^{eq}, which is seen to increase with N_x significantly. The expansion can be controlled to an extent by increasing the value of β such that the pseudopotential force is stronger.

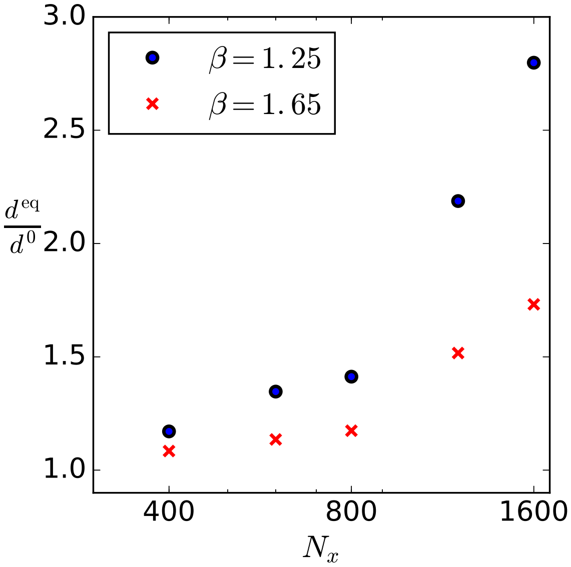


Figure B.17: Dependence of the droplet expansion on the domain size, shown as the ratio between an equivalent droplet diameter d^{eq} calculated after a stationary droplet is let to relax for 20000 iterations, beginning from an initial diameter $d^0 = 100$ lattice units, for increasing domain size in lattice units where $N_X = N_y$.

A comparison of the force magnitude is shown in Fig. B.18, which shows the interaction force along the centerline of a stationary droplet for simulations with $\beta = 1.25$ & 1.65. It can be seen that for $\beta = 1.65$, the interaction force increases slightly near the interface. Even this slight increase in interaction force influences droplet expansion significantly. It is also seen in from Fig. B.18 that the size of the stationary droplet is smaller for $\beta = 1.65$ as compared to the case with $\beta = 1.25$.

Lastly, the final droplet shapes obtained using $\beta = 1.25$ & 1.65 for Cases 1 and 5 are presented below in Fig. B.19. Here the shapes for both values of β are very similar, but when a similar comparison was performed on a smaller domain for these two cases, the $\beta = 1.25$ cases showed unphysical expansion and deformation. Further work is required to understand the exact behaviour depending on β and the domain size effect.

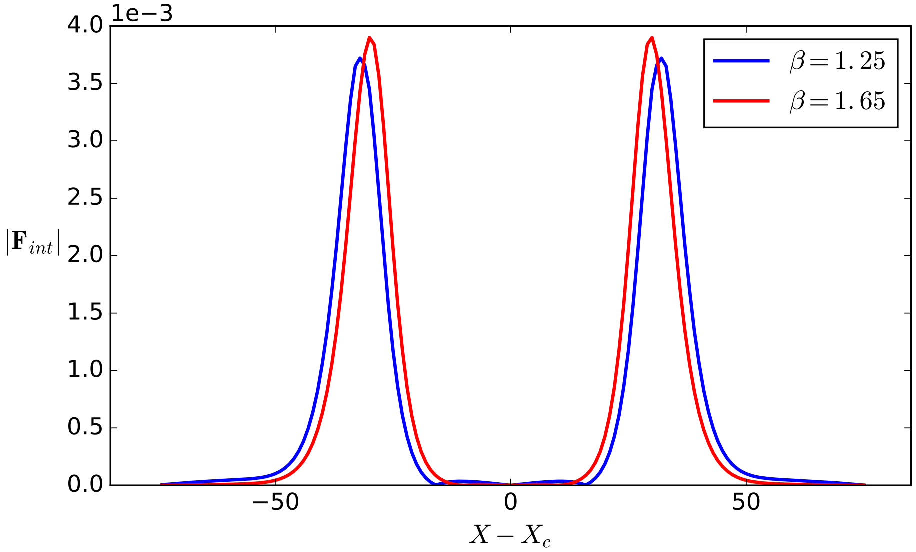
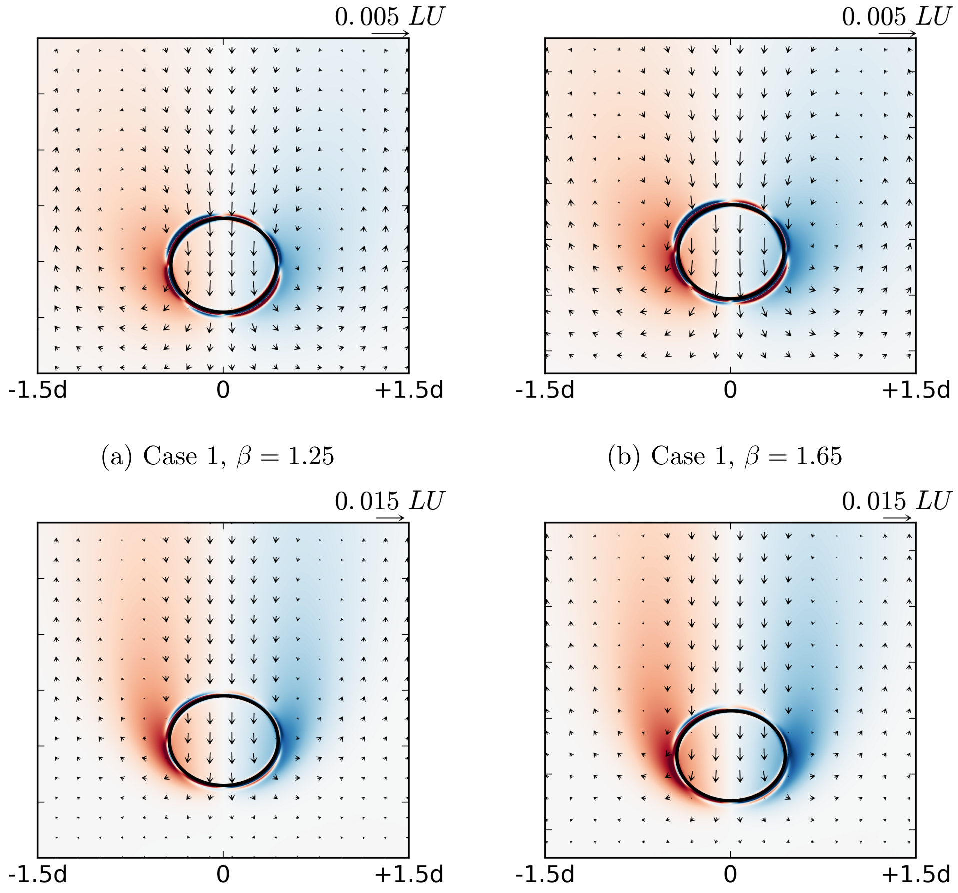


Figure B.18: Force distribution over a diametric axis across a droplet centered at X_c for $\beta = 1.25$ and $\beta = 1.65$.



(a) Case 1, $\beta = 1.25$

(b) Case 1, $\beta = 1.65$

(c) Case 5, $\beta = 1.25$

(d) Case 5, $\beta = 1.65$

Figure B.19: Comparison of final droplet shapes for Case 1 and Case 5 from LB2D simulations for $\beta = 1.25$ (lower surface tension and relatively higher inter-phase miscibility) and $\beta = 1.65$ (higher surface tension and low inter-phase miscibility).

[1] A. Prosperetti, G. Tryggvason, Computational methods for multiphase flow, Cambridge university press, 2009.

[2] G. Tryggvason, R. Scardovelli, S. Zaleski, Direct numerical simulations of gas–liquid multiphase flows, Cambridge University Press, 2011.

[3] M. Loewenberg, E. Hinch, Numerical simulation of a concentrated emulsion in shear flow, Journal of Fluid Mechanics 321 (1996) 395–419.

[4] V. Cristini, Y.-C. Tan, Theory and numerical simulation of droplet dynamics in complex flowsa review, Lab on a Chip 4 (2004) 257–264.

[5] S. F. Roudsari, G. Turcotte, R. Dhib, F. Ein-Mozaffari, Cfd modeling of the mixing of water in oil emulsions, Computers & Chemical Engineering 45 (2012) 124–136.

[6] S. Kokal, et al., Crude oil emulsions: A state-of-the-art review, in: SPE Annual Technical Conference and Exhibition, Society of Petroleum Engineers.

[7] L. W. Lake, Enhanced oil recovery (1989).

[8] R. Arshady, Suspension, emulsion, and dispersion polymerization: A methodological survey, Colloid & Polymer Science 270 (1992) 717–732.

[9] N. Takada, M. Misawa, A. Tomiyama, S. Fujiwara, Numerical simulation of two-and three-dimensional two-phase fluid motion by lattice boltzmann method, Computer Physics Communications 129 (2000) 233–246.

[10] K. Sankaranarayanan, I. Kevrekidis, S. Sundaresan, J. Lu, G. Tryggvason, A comparative study of lattice boltzmann and front-tracking

finite-difference methods for bubble simulations, International Journal of Multiphase Flow 29 (2003) 109–116.

[11] L. Scarbolo, D. Molin, P. Perlekar, M. Sbragaglia, A. Soldati, F. Toschi, Unified framework for a side-by-side comparison of different multicomponent algorithms: Lattice boltzmann vs. phase field model, Journal of Computational Physics 234 (2013) 263–279.

[12] C. W. Hirt, B. D. Nichols, Volume of fluid (vof) method for the dynamics of free boundaries, Journal of computational physics 39 (1981) 201–225.

[13] V. R. Gopala, B. G. van Wachem, Volume of fluid methods for immiscible-fluid and free-surface flows, Chemical Engineering Journal 141 (2008) 204–221.

[14] L. Chen, Q. Kang, Y. Mu, Y.-L. He, W.-Q. Tao, A critical review of the pseudopotential multiphase lattice boltzmann model: Methods and applications, International Journal of Heat and Mass Transfer 76 (2014) 210–236.

[15] X. Shan, H. Chen, Lattice boltzmann model for simulating flows with multiple phases and components, Physical Review E 47 (1993) 1815.

[16] X. Shan, H. Chen, Simulation of nonideal gases and liquid-gas phase transitions by the lattice boltzmann equation, Physical Review E 49 (1994) 2941.

[17] S. Chen, S. Dawson, G. Doolen, D. Janecky, A. Lawniczak, Lattice methods and their applications to reacting systems, Computers & chemical engineering 19 (1995) 617–646.

[18] S. Chen, G. D. Doolen, Lattice boltzmann method for fluid flows, Annual review of fluid mechanics 30 (1998) 329–364.

[19] M. R. Swift, W. Osborn, J. Yeomans, Lattice boltzmann simulation of nonideal fluids, Physical Review Letters 75 (1995) 830.

[20] P. Yuan, L. Schaefer, Equations of state in a lattice boltzmann model, Physics of Fluids 18 (2006) 042101.

[21] S. Mukherjee, A. Zarghami, C. Haringa, S. Kenjereš, H. E. Van den Akker, A comparative assessment of lattice boltzmann and volume of fluid (vof) approaches for generic multiphase problems, in: Proceedings of ICMF 2016 - 9th International Conference on Multiphase Flow.

[22] R. Clift, J. R. Grace, M. E. Weber, Bubbles, drops, and particles, Courier Corporation, 2005.

[23] S. Succi, The lattice Boltzmann equation: for fluid dynamics and beyond, Oxford university press, 2001.

[24] S. Gong, P. Cheng, Numerical investigation of droplet motion and coalescence by an improved lattice boltzmann model for phase transitions and multiphase flows, Computers & Fluids 53 (2012) 93–104.

[25] A. Zarghami, N. Looije, H. Van den Akker, Assessment of interaction potential in simulating nonisothermal multiphase systems by means of lattice boltzmann modeling, Physical Review E 92 (2015) 023307.

[26] A. Zarghami, H. E. Van den Akker, Thermohydrodynamics of an evaporating droplet studied using a multiphase lattice boltzmann method, Physical Review E 95 (2017) 043310.

[27] A. Kupershtokh, D. Medvedev, D. Karpov, On equations of state in a lattice boltzmann method, Computers & Mathematics with Applications 58 (2009) 965–974.

[28] L. Amaya-Bower, T. Lee, Single bubble rising dynamics for moderate reynolds number using lattice boltzmann method, Computers & Fluids 39 (2010) 1191–1207.

[29] K. Sankaranarayanan, X. Shan, I. Kevrekidis, S. Sundaresan, Bubble flow simulations with the lattice boltzmann method, Chemical Engineering Science 54 (1999) 4817–4823.

[30] K. Sankaranarayanan, X. Shan, I. Kevrekidis, S. Sundaresan, Analysis of drag and virtual mass forces in bubbly suspensions using an implicit formulation of the lattice boltzmann method, Journal of Fluid Mechanics 452 (2002) 61–96.

[31] H. Huang, M. Krafczyk, X. Lu, Forcing term in single-phase and shan-chen-type multiphase lattice boltzmann models, Physical Review E 84 (2011) 046710.

[32] H. Jasak, A. Jemcov, Z. Tukovic, et al., Openfoam: A c++ library for complex physics simulations, in: International workshop on coupled methods in numerical dynamics, volume 1000, IUC Dubrovnik, Croatia, pp. 1–20.

[33] H. Jasak, Openfoam: open source cfd in research and industry, International Journal of Naval Architecture and Ocean Engineering 1 (2009) 89–94.

[34] S. S. Deshpande, L. Anumolu, M. F. Trujillo, Evaluating the performance of the two-phase flow solver interfoam, Computational science & discovery 5 (2012) 014016.

[35] H. Marschall, K. Hinterberger, C. Schueler, F. Habla, O. Hinrichsen, Numerical simulation of species transfer across fluid interfaces in free-surface flows using openfoam, Chemical engineering science 78 (2012) 111–127.

[36] A. Q. Raeini, M. J. Blunt, B. Bijeljic, Modelling two-phase flow in porous media at the pore scale using the volume-of-fluid method, Journal of Computational Physics 231 (2012) 5653–5668.

[37] D. A. Hoang, V. van Steijn, L. M. Portela, M. T. Kreutzer, C. R. Kleijn, Benchmark numerical simulations of segmented two-phase flows in microchannels using the volume of fluid method, Computers & Fluids 86 (2013) 28–36.

[38] C. Bilger, M. Aboukhedr, K. Vogiatzaki, R. Cant, Evaluation of two-phase flow solvers using level set and volume of fluid methods, Journal of Computational Physics (2017).

[39] E. Berberovic, Investigation of free-surface flow associated with drop impact: numerical simulations and theoretical modeling, Ph.D. thesis, TU Darmstadt/FG Strömungslehre und Aerodynamik, 2010.

[40] H. Rusche, Computational fluid dynamics of dispersed two-phase flows at high phase fractions, Ph.D. thesis, Imperial College London (University of London), 2003.

[41] H. Weller, A new approach to vof-based interface capturing methods for incompressible and compressible flow, OpenCFD Ltd., Report TR/HGW/04 (2008).

[42] P. Bohórquez, Study and numerical simulation of sediment transport in free-surface flow, Technical Report, University of Malaga, Spain, 2008.

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

[44] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows with surfer, Journal of Computational Physics 113 (1994) 134–147.

[45] M. J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, Journal of computational Physics 53 (1984) 484–512.

[46] R. I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, Journal of computational physics 62 (1986) 40–65.

[47] S. Patankar, Numerical heat transfer and fluid flow, CRC press, 1980.

[48] D. L. Youngs, Time-dependent multi-material flow with large fluid distortion, Numerical methods for fluid dynamics 24 (1982) 273–285.

[49] O. Ubbink, Numerical prediction of two fluid systems with sharp interfaces, Ph.D. thesis, University of London PhD Thesis, 1997.

[50] M. K. Tripathi, K. C. Sahu, R. Govindarajan, Why a falling drop does not in general behave like a rising bubble, Scientific reports 4 (2014).

[51] M. K. Tripathi, K. C. Sahu, R. Govindarajan, Dynamics of an initially spherical bubble rising in quiescent liquid, Nature communications 6 (2015).

[52] J. Hadamard, Mouvement permanent lent dune sphere liquide et visqueuse dans un liquide visqueux, CR Acad. Sci 152 (1911) 1735–1738.

[53] W. Rybczynski, On the translatory motion of a fluid sphere in a viscous medium, Bull. Acad. Sci., Cracow, Series A 40 (1911).

[54] H. D. Mendelson, The prediction of bubble terminal velocities from wave theory, AIChE Journal 13 (1967) 250–253.

[55] Z. Zapryanov, S. Tabakova, Dynamics of bubbles, drops and rigid particles, volume 50, Springer Science & Business Media, 2013.

[56] J. Han, G. Tryggvason, Secondary breakup of axisymmetric liquid drops. i. acceleration by a constant body force, Physics of Fluids (1994-present) 11 (1999) 3650–3667.

[57] W. J. Dahm, C. M. Scheil, G. Tryggvason, Dynamics of vortex interaction with a density interface, Journal of Fluid Mechanics 205 (1989) 1–43.

[58] D. J. Harvie, M. Davidson, M. Rudman, An analysis of parasitic current generation in volume of fluid simulations, Applied mathematical modelling 30 (2006) 1056–1066.

[59] X. Shan, Analysis and reduction of the spurious current in a class of multiphase lattice boltzmann models, Physical Review E 73 (2006) 047701.

[60] J. Klostermann, K. Schaake, R. Schwarze, Numerical simulation of a single rising bubble by vof with surface compression, International Journal for Numerical Methods in Fluids 71 (2013) 960–982.

[61] Y. Renardy, M. Renardy, Prost: a parabolic reconstruction of surface tension for the volume-of-fluid method, Journal of Computational Physics 183 (2002) 400–421.

[62] M. Sbragaglia, R. Benzi, L. Biferale, S. Succi, K. Sugiyama, F. Toschi, Generalized lattice boltzmann method with multirange pseudopotential, Physical Review E 75 (2007) 026702.

[63] Q. Li, K. Luo, Thermodynamic consistency of the pseudopotential lattice boltzmann model for simulating liquid–vapor flows, Applied Thermal Engineering 72 (2014) 56–61.

[64] J. Hua, J. Lou, Numerical simulation of bubble rising in viscous liquid, Journal of Computational Physics 222 (2007) 769–795.

[65] L. Janssen, M. Warmoeskerken, Transport phenomena data companion, VSSD Delft, The Netherlands, 2006.

[66] K. Connington, T. Lee, A review of spurious currents in the lattice boltzmann method for multiphase flows, Journal of mechanical science and technology 26 (2012) 3857.

[67] M. Magnini, B. Pulvirenti, J. Thome, Characterization of the velocity fields generated by flow initialization in the cfd simulation of multiphase flows, Applied Mathematical Modelling 40 (2016) 6811–6830.

[68] R. Verberg, A. Ladd, Simulation of low-reynolds-number flow via a

time-independent lattice-boltzmann method, Physical Review E 60 (1999) 3366.

[69] M. Bernaschi, S. Succi, H. Chen, Accelerated lattice boltzmann schemes for steady-state flow simulations, Journal of scientific computing 16 (2001) 135–144.

[70] T. Krüger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, E. M. Viggen, The Lattice Boltzmann Method: Principles and Practice, Springer, 2016.

[71] Z. Guo, T. Zhao, Y. Shi, Preconditioned lattice-boltzmann method for steady flows, Physical Review E 70 (2004) 066706.

[72] S. Izquierdo, N. Fueyo, Optimal preconditioning of lattice boltzmann methods, Journal of Computational Physics 228 (2009) 6479–6495.

[73] D. d'Humières, Multiple–relaxation–time lattice boltzmann models in three dimensions, Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 360 (2002) 437–451.

[74] M. E. McCracken, J. Abraham, Multiple-relaxation-time lattice-boltzmann model for multiphase flow, Physical Review E 71 (2005) 036701.

[75] D. O. Martinez, W. H. Matthaeus, S. Chen, D. Montgomery, Comparison of spectral method and lattice boltzmann simulations of two-dimensional hydrodynamics, Physics of Fluids 6 (1994) 1285–1298.

[76] M. Sbragaglia, S. Succi, A note on the lattice boltzmann method beyond the chapman-enskog limits, EPL (Europhysics Letters) 73 (2005) 370.

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