

Open-source finite volume solvers for multiphase (n-phase) flows involving either Newtonian or non-Newtonian complex fluids

Chris Dritselis, George Karapetsas*

Department of Chemical Engineering, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece

ARTICLE INFO

Keywords:

Multiphase flow
Volume-of-fluid
Contact line
Non-newtonian
Floating lens
Drop levitation
multiFluidInterFoam
RheoMultiFluidInterFoam

ABSTRACT

Two new control volume solvers `multiFluidInterFoam` and `rheoMultiFluidInterFoam` are presented for the simulation of Newtonian and non-Newtonian n-phase flows ($n \geq 2$), respectively, fully accounting for interfacial tension and contact-angle effects for each phase. The `multiFluidInterFoam` solver modifies certain crucial aspects of the regular `multiPhaseInterFoam` solver provided by OpenFOAM for Newtonian flows, improving its efficiency and robustness, but most importantly improving considerably its accuracy for surface tension driven flows. The new solver uses the volume-of-fluid (VOF) method based on the MULES method and artificial interface compression for the interface capturing, in combination with a suitable continuum surface force model; i.e. the interfacial tension coefficient decomposition method is employed to treat pairings of interfacial tension between the phases. VOF smoothing is also implemented by applying a Laplacian filter on the distribution of volume fractions, and the SIMPLEC algorithm is employed for the velocity–pressure coupling. The above algorithm has also been extended with the development of the `rheoMultiFluidInterFoam` solver which is capable of fully taking into account complex non-Newtonian effects (e.g. yield stress, viscoelasticity, etc.) of the involved liquids. To this end, the developed solver incorporates the `RheoTool` toolbox (Pimenta and Alves, 2017), utilizing a wealth of constitutive equations suitable for modeling different types of fluids with complex rheology. Our solvers are tested for both the implementation of the surface tension and of the constitutive equations in a wide range of different flow setups, considering typical benchmark two-phase and three-phase flows, such as the cases of dam break with an obstacle, spreading of a floating lens, a levitating drop, and a bubble rising in an ambient fluid; flows involving Newtonian, viscoplastic and viscoelastic liquids have been considered. Comparisons against analytical solutions or previously published numerical data are performed clearly demonstrating the capability of the new solvers to efficiently and accurately simulate interfacial tension dominated three-phase flows (and n-phase flows in general) for both simple and complex liquids.

1. Introduction

Multiphase flows, where two or more fluids have interfacial surfaces, frequently arise in many industrial engineering and scientific applications. A typical example is oil recovery, which involves at least three-phase fluids including oil, water, and air. Other examples are nuclear reactors and microfluidics. The flow in all these applications exhibits a wide range of spatial and temporal scales, and it includes various complex interactions between the fluids, the motion of droplets and bubbles, as well as the formation of films and jets. Despite the fact that there is a sufficient number of numerical studies on two-phase flows, research on cases where a higher number of phases may be involved, e.g. three-phase flows, is still very limited, partly because of the variety and complexity of interactions between the phases and difficulties in the numerical treatment of special locations

where the three interfaces connect, i.e., triple junction points. With the rapid development of computer technology, numerical modeling provides reliable and accurate tools and it has a considerable role in the detailed investigation of multiphase flows. In the past, many efforts have been made to develop numerical approaches for the simulation of three-phase flows based on the numerical solution of Navier–Stokes equations together with various approaches devoted to the capturing or tracking of the interfaces between fluids, such as the level-set [1,2], front-tracking [3], volume-of-fluid [4,5], smoothed particle hydrodynamics [6], and phase field [7–9] methods. Usually, the fluids under consideration are Newtonian, with far less attention being paid to the use and development of such methods in the study of three-phase flows when one or more fluids exhibits a non-Newtonian behavior.

* Corresponding author.

E-mail addresses: cdritselis@gmail.com (C. Dritselis), gkarapetsas@auth.gr (G. Karapetsas).

<https://doi.org/10.1016/j.compfluid.2022.105590>

Received 29 September 2021; Received in revised form 25 May 2022; Accepted 5 July 2022

Available online 8 July 2022

0045-7930/© 2022 Elsevier Ltd. All rights reserved.

The present study is focused on the development of efficient and accurate numerical tools for the investigation of interfacial tension dominated flows involving more than two phases, such as for example, the three-phase flow of a droplet spreading between two stratified fluids. Earlier attempts to address the modeling of n -phase Newtonian flows under the framework of OpenFOAM [10,11] have culminated in the development of the `multiphaseInterFoam` solver [12].

As it will be discussed in detail below, the latter solver performs quite well in inertia dominated flows, but exhibits very poor performance in flows dominated by surface tension. To address this important issue, a new generic multiphase solver is presented, named `multiFluidInterFoam`, capable to accurately model Newtonian n -phase flows, $n \geq 2$, and fully taking into account interfacial phenomena and contact line effects. The new solver has been developed building upon and considerably improving, the regular `multiphaseInterFoam` solver provided by OpenFOAM [10,11]. The numerical framework of the developed solver consists of a finite volume method and the volume-of-fluid method (VOF) that employs a volumetric phase fraction for each fluid phase. Sharpening of the interface is achieved by introducing a properly adapted artificial compression term in the equation for each volume fraction. Smoothing of the volume fraction of each phase is also suitably incorporated in the calculation of curvature of interfaces for n -phase flows. The multiphase nature of the system is taken into account by a continuum surface force model. In contrast to the original multiphase solver, the interfacial tension coefficient decomposition method is used here to handle the pairings of interface tension between the different phases.

The `multiFluidInterFoam` solver has been developed for Newtonian fluids. However, a situation where one or more fluids exhibits a non-Newtonian behavior is not unusual. Such flows may arise in a number of different setups ranging from environmental applications (e.g. spreading of weathered oil spills exhibiting complex rheology due to evaporation of the most volatile components) to biomedical applications (e.g., inhaled drug in the form of aerosol droplets spreading along the mucus lining of the lung, which commonly exhibits viscoelastic properties and yield stress) [13,14].

To address the modeling of such flows, a second solver `rheoMultiFluidInterFoam` is developed as an extension of the aforementioned `multiFluidInterFoam`, able to handle multiphase flows, where any number of the involved phases can exhibit a rheologically complex behavior. To the best of our knowledge, the developed solver presented herein is the first fully capable solver of handling and accurately solving surface tension driven n -phase flows involving any number of rheologically complex fluids, including viscoelastic or elastoviscoplastic effects. This goal has been achieved by combining the VOF method, properly adapted for multiple fluids, with the `RheoTool` toolbox [15], in order to make use of several constitutive equations for the simulation of generalized Newtonian and viscoelastic fluids.

The new multiphase solvers are both validated through comparisons with existing analytical and numerical solutions in representative three-phase flows involving Newtonian fluids. The problems of the dam break, a floating lens, and the levitation of drop initially located within two fluids are used here for comparison purposes. Since there is no rigorous benchmark for non-Newtonian three-phase fluid flows, for the validation of the `rheoMultiFluidInterFoam` solver the following approach is adopted. First, we examine the two-phase problem of a bubble rising both in a viscoplastic (described by Herschel–Bulkley model) and a viscoelastic (described by Phan-Thien Tanner model) fluid, as a reference to validate the presently `RheoTool` based implementation of the model. It is pointed out that the main objective in the two-phase flows examined here is on assessing the performance of the constitutive equations for the non-Newtonian fluids provided by the `rheoTool` toolbox, and not that of the surface tension implementation. The latter has been the subject of previous works. Finally, consistency verification tests are performed for the floating lens and levitating drop problems,

which are re-examined by assuming that the bottom fluid layer exhibits a viscoplastic behavior.

The remainder of this article is organized as follows. Description of the models and numerical methods is given in Section 2. Results are shown and compared with analytical solutions or recently published numerical data in Section 3. Finally, a discussion is presented in Section 4 and some concluding remarks are summarized in Section 5. In addition to describe and provide access to the source codes of a robust open-source three-phase flow solvers (available for download on GitHub: <https://github.com/gkarapetsas/multiFluidInterFoam> and <https://github.com/gkarapetsas/rheoMultiFluidInterFoam>).

2. Governing equations and numerical methods

We begin by presenting in summary the main features of the standard `multiphaseInterFoam` solver provided by OpenFOAM and then proceed with a detailed presentation of the key features of the newly developed solvers.

2.1. Overview of the original `multiphaseInterFoam` solver provided by OpenFOAM

The standard multiphase solver available in OpenFOAM-v7 is `multiphaseInterFoam`. In contrast to the two-phase incompressible flow solver `interFoam`, it accounts for several incompressible fluids with interface capturing, surface-tension and contact-angle effects for each phase. Several non-Newtonian models are available at runtime, including a bounded bi-viscosity version of the Herschel–Bulkley model, which will also be used in the present work for comparison purposes.

In multi-fluids flows, several fluids are present in one or more phases. The volume fraction of fluid i is denoted as α_i , where $i = 1, 2, \dots, N_c$ and N_c is the number of fluids. In the present work, we shall focus on interfacial flows with three fluids and, thus, consider $N_c = 3$. The density and dynamic (kinematic) viscosity of fluid i are ρ_i^* and μ_i^* (ν_i^*), respectively; the asterisk indicates a dimensional quantity. The governing equations for multiphase flows are the equation of continuity, momentum and of transport of the volume fraction distributions. The continuity equation is

$$\nabla \cdot \mathbf{u}^* = 0. \quad (1)$$

The incompressible Navier–Stokes equation (NSE) is reformulated in order to take into account the multiphase nature of the system as

$$\frac{\partial \rho^* \mathbf{u}^*}{\partial t^*} + \nabla \cdot (\rho^* \mathbf{u}^* \mathbf{u}^*) = -\nabla p_d^* + \mathbf{g}^* \cdot \mathbf{x}^* \nabla \rho^* + \nabla \cdot \underline{\boldsymbol{\tau}}^* + \mathbf{f}_{st}^*. \quad (2)$$

In the above equations, \mathbf{u}^* is the velocity, t^* is the time, \mathbf{g}^* is the gravity acceleration, \mathbf{x}^* is the position vector, $\underline{\boldsymbol{\tau}}^*$ is the deviatoric stress tensor, ρ^* denotes the bulk fluid density and \mathbf{f}_{st}^* is the source of momentum due to the surface tension. The quantity p_d^* is a modified pressure containing the static pressure p^* and the hydrostatic component $p_d^* = p^* - \rho^* \mathbf{g}^* \cdot \mathbf{x}^*$.

In general, the fluids can be considered generalized Newtonian with

$$\underline{\boldsymbol{\tau}}^* = \rho^* \nu^* (\dot{\boldsymbol{\gamma}}^*) \dot{\boldsymbol{\gamma}}^*, \quad (3)$$

where $\dot{\boldsymbol{\gamma}}^* = \nabla \mathbf{u}^* + \nabla \mathbf{u}^{*T}$ is the rate of strain tensor, $\dot{\boldsymbol{\gamma}}^* = \sqrt{(\dot{\boldsymbol{\gamma}}^* : \dot{\boldsymbol{\gamma}}^*)/2}$ is its second invariant and ν^* is the bulk kinematic viscosity.

The capturing of interfaces is performed by the volume of fluid method (VOF) [16] utilizing the volumetric phase fraction α_i for each fluid i . The transport equation for the i th volume fraction α_i is

$$\frac{\partial \alpha_i}{\partial t^*} + \nabla \cdot (\alpha_i \mathbf{u}_i^*) = 0, \quad i = 1, 2, \dots, N_c, \quad (4)$$

where \mathbf{u}_i^* is the velocity field corresponding to the i th fluid. The volume fractions α_i are coupled with each other by the normalization:

$$\sum_{i=1}^{N_c} \alpha_i = 1, \quad \alpha_i \in [0, 1]. \quad (5)$$

The bulk density ρ^* and the bulk viscosities μ^* , ν^* are treated as volumetric mixture values and they are computed in each computational cell using the densities ρ_i^* and viscosities μ_i^* , ν_i^* of each fluid and the volumetric phase fractions as weighting factors as

$$\rho^* = \sum_{i=1}^{N_c} \alpha_i \rho_i^*, \quad (6)$$

$$\nu^*(\dot{\gamma}^*) = \frac{1}{\rho^*} \sum_{i=1}^{N_c} \alpha_i \rho_i^* \nu_i^*(\dot{\gamma}^*). \quad (7)$$

In the case of a purely Newtonian fluid, the viscosity is simply considered to be constant

$$\nu_i^*(\dot{\gamma}^*) = \nu_{i,Newtonian}^* = \text{constant}, \quad (8)$$

whereas other available constitutive equations such as the power-law, Carreau and Herschel–Bulkley models are available to the user. For the purposes of the present study we will consider the case of a viscoplastic material described by the Herschel–Bulkley model

$$\nu_i^*(\dot{\gamma}^*) = \min\left(\nu_{0,i}^*, \tau_{0,i}^*/\dot{\gamma}^* + k_i^*(\dot{\gamma}^*)^{n_i-1}\right), \quad (9)$$

which combines the effects of Bingham plastic and power-law behavior in a fluid. For low strain rates $\dot{\gamma}^*$, the material is modeled as a very viscous fluid with viscosity $\nu_{0,i}^*$. Beyond a threshold in strain-rate corresponding to a threshold stress $\tau_{0,i}^*$, the viscosity is described by a power law; where k_i^* is the flow consistency index and n_i is the power-law index of each phase.

The effect of surface tension is implemented using the Continuum-Surface-Force (CSF) model of Brackbill et al. [17] as a volumetric force near the interface as [18,19]

$$\mathbf{f}_{st}^* = \sum_i \sum_{j \neq i} \sigma_{ij}^* \kappa_{ij}^* \delta_{ij}^*, \quad (10)$$

with σ_{ij}^* denoting the constant interface tension between phases i and j . In this approach the interface is neither tracked explicitly nor shape or location are known and, thus, an exact boundary condition cannot be applied at the interfaces. The curvature of the interface ij is

$$\kappa_{ij}^* = -\nabla \cdot \frac{\alpha_j \nabla \alpha_i - \alpha_i \nabla \alpha_j}{|\alpha_j \nabla \alpha_i - \alpha_i \nabla \alpha_j| + sf}, \quad (11)$$

where sf is a stabilization factor that accounts for non-uniformity of the computational mesh $sf = \epsilon / (\sum_{i=1}^{ncv} V_i / ncv)^{(1/3)}$, ncv is the number of computational cells, V_i is the volume of cell i , and ϵ is a small positive number, typically $\epsilon = 10^{-8}$. The term $\delta_{ij}^* = \alpha_j \nabla \alpha_i - \alpha_i \nabla \alpha_j$ applies the volumetric force only near the interfaces, where variations of the indicator function δ_{ij}^* are present. Thus, the surface tension force becomes

$$\mathbf{f}_{st}^* = -\sum_i \sum_{j \neq i} \sigma_{ij}^* \nabla \cdot \left(\frac{\alpha_j \nabla \alpha_i - \alpha_i \nabla \alpha_j}{|\alpha_j \nabla \alpha_i - \alpha_i \nabla \alpha_j| + sf} \right) (\alpha_j \nabla \alpha_i - \alpha_i \nabla \alpha_j). \quad (12)$$

The fluid interfaces are sharpened by introducing artificial compression terms into Eq. (4), which it may be re-written as

$$\frac{\partial \alpha_i}{\partial t^*} + \nabla \cdot \left(\alpha_i \mathbf{u}^* + \sum_{k \neq i, k=1}^{N_c} \alpha_i \alpha_k \mathbf{u}_{r,ik}^* \right) = 0, \quad i = 1, 2, \dots, N_c, \quad (13)$$

where $\mathbf{u}_{r,ik}^*$ is the artificial compression velocity given by

$$(\mathbf{u}_{r,ik}^*)_{\mathbf{f}} = (\mathbf{n}_{ik})_{\mathbf{f}} \min \left[C_{\alpha_i} \frac{|\phi_{\mathbf{f}}^*|}{|\mathbf{S}_{\mathbf{f}}^*|}, \max \left(\frac{|\phi_{\mathbf{f}}^*|}{|\mathbf{S}_{\mathbf{f}}^*|} \right) \right], \quad (14)$$

$$i = 1, 2, \dots, N_c \text{ and } k \neq i,$$

where $(\mathbf{n}_{ik})_{\mathbf{f}}$ is the normal vector of the cell surface, $\phi_{\mathbf{f}}^*$ is the volume flux given as $\phi_{\mathbf{f}}^* = \mathbf{u}_{\mathbf{f}}^* \cdot \mathbf{S}_{\mathbf{f}}^*$, $\mathbf{S}_{\mathbf{f}}^*$ is the cell surface area, and C_{α_i} is an adjustable coefficient, the value of which can be set between 0 and 4. The derivation of Eq. (13) is shown in Appendix A. The quantity $(\mathbf{u}_{r,ik}^*)_{\mathbf{f}}$ can be interpreted as a relative velocity between fluids i and k , which arises from the changes in the density and viscosity within the

fluid interface. By taking the divergence of the compression velocity $\mathbf{u}_{r,ik}^*$, the conservation of the VOF function is guaranteed [20]. The term $\alpha_i \alpha_k$ in Eq. (13) ensures that the compression term is active only in the interfacial area where $0 < \alpha_i < 1$. The level of compression depends on the value of C_{α_i} . There is no compression for $C_{\alpha_i} = 0$, a moderate compression exists for $0 < C_{\alpha_i} \leq 1$, while an enhanced compression takes place for $1 < C_{\alpha_i} \leq 4$ [21,22]. The normal vector $(\mathbf{n}_{ik})_{\mathbf{f}}$ is calculated as

$$(\mathbf{n}_{ik})_{\mathbf{f}} = \frac{(\alpha_k)_{\mathbf{f}} (\nabla \alpha_i)_{\mathbf{f}} - (\alpha_i)_{\mathbf{f}} (\nabla \alpha_k)_{\mathbf{f}}}{|(\alpha_k)_{\mathbf{f}} (\nabla \alpha_i)_{\mathbf{f}} - (\alpha_i)_{\mathbf{f}} (\nabla \alpha_k)_{\mathbf{f}}| + sf}. \quad (15)$$

In the `multiphaseInterFoam` solver, the governing equations are solved sequentially in a segregated manner, where the transport equations for the volume fractions (Eqs. (4), (5)) are first solved, the properties of the mixture (Eqs. (6)–(7)) are then calculated based on the constitutive equations (Eqs. (8), (9)), followed by the solution of the momentum equation (Eq. (2)) and the continuity (pressure) equation (Eq. (1)). The velocity–pressure coupling is ensured using the PIMPLE algorithm that is a combination of PISO [23] (Pressure-Implicit Split Operator) and SIMPLE [24] (Semi-Implicit Method for the Pressure-Linked Equation) algorithms.

2.2. Overview of the `multiFluidInterFoam` solver

In the following sections, we describe the new `multiFluidInterFoam` solver and the main modifications introduced in the original multiphase solver, which are suitable for the accurate simulation of interfacial tension dominated, three-phase flows; the method should in principle be valid for any n -phase flows with $n \geq 2$. These changes can considerably increase the stability and computational speed of the new solver, while keeping its time and space accuracy.

2.2.1. Interfacial tension model

The interfacial tension force in two-phase flows is usually obtained by using the Continuum-Surface-Force (CSF) [17] method as $\mathbf{f}_{st}^* = \sigma_{12}^* \kappa_{12}^* \mathbf{n}_{12} \delta^*$, where σ_{12}^* is the interfacial tension coefficient between fluid 1 and 2, $\kappa_{12}^* = \nabla \cdot \mathbf{n}_{12}$ is the interfacial curvature, \mathbf{n}_{12} is the interface unit normal, and δ^* is the Dirac delta function. In three-phase flows, the fluid flow might be influenced simultaneously by more than two interfaces adjacent to triple junction points, and, thus, the above definition has to be further extended.

As shown in Section 2.1, the original `multiphaseInterFoam` solver adopts Eq. (12) in order to account for the surface tension. In the present work, the interfacial tension coefficient decomposition method is employed to deal with tension pairings between different phases using a compositional approach [2,5,6,9]. The physical interfacial tension coefficients σ_{ij}^* between phase i and phase j are decomposed into phase-specific interfacial tension coefficients as $\sigma_{ij}^* = \Sigma_i^* + \Sigma_j^*$. In particular, for a three-phase flow ($N_c = 3$) once the physical interfacial tension coefficients σ_{ij}^* are known, the phase-specific interfacial tension coefficients can be obtained as

$$\Sigma_1^* = 0.5(\sigma_{12}^* + \sigma_{13}^* - \sigma_{23}^*), \quad (16)$$

$$\Sigma_2^* = 0.5(\sigma_{12}^* + \sigma_{23}^* - \sigma_{13}^*), \quad (17)$$

$$\Sigma_3^* = 0.5(\sigma_{13}^* + \sigma_{23}^* - \sigma_{12}^*). \quad (18)$$

The resulting interfacial tension force can be rewritten as the sum of each fluid i force

$$\mathbf{f}_{st}^* = \sum_{i=1}^{N_c} \Sigma_i^* \kappa_i^* \mathbf{n}_i \delta_i^*. \quad (19)$$

In accordance with the aforementioned works, we also use $\delta_i^* = |\nabla \alpha_i|$, $\mathbf{n}_i = \nabla \alpha_i / (|\nabla \alpha_i| + sf)$ and $\kappa_i^* = \nabla \cdot \mathbf{n}_i$, and the continuum surface force can be written based on the phase volume fraction α_i as:

$$\mathbf{f}_{st}^* = \sum_{i=1}^{N_c} \Sigma_i^* \kappa_i^* \nabla \alpha_i. \quad (20)$$

It is noted that according to this formulation the unit normal vector \mathbf{n}_i and the curvature κ_i^* have to be calculated separately for each phase i . The gradient terms are computed at the cell faces, resulting in a discrete balance between pressure and surface tension in a similar manner as discussed in [22].

2.2.2. VOF smoothing

In the VOF method, the fluid interface is implicitly represented by the volume fraction, with its value changing abruptly in a small but finite region. This creates errors in the evaluation of the normal vectors and the curvature of interfaces, which are used to calculate the interfacial body forces. In flows dominated by interfacial tension effects, these errors usually give rise to non-physical parasitic currents adjacent to the interface due to local variations in the continuum surface force [25–27]. These numerical artifacts may be efficiently suppressed by evaluating the curvature of interfaces κ_i^* from a smoothed VOF distribution $\tilde{\alpha}_i$. Here, the smoothed function is calculated from α_i by applying a filter over a finite region around the interface [26,28]. Thus, the curvature κ_i^* is computed as

$$\kappa_i^* = \nabla \cdot \left(\frac{\nabla \tilde{\alpha}_i}{|\nabla \tilde{\alpha}_i| + s_f} \right). \quad (21)$$

We applied as a smoother a Laplacian filter proposed by Lafaurie et al. [25] that transforms the original VOF function α_i into a smoothed function $\tilde{\alpha}_{i,p}$

$$\tilde{\alpha}_{i,p} = \frac{\sum_{f=1}^{n_f} (\alpha_i)_f |S_f^*|}{\sum_{f=1}^{n_f} |S_f^*|}, \quad (22)$$

where the subscript ‘p’ denotes the cell index and ‘f’ denotes the face index, n_f is the number of faces of a finite control volume, and S_f^* is the cell surface area. The interpolated value $(\alpha_i)_f$ at the face centre is calculated using linear interpolation. The application of this filter can be repeated n_{if} times in order to obtain a smoothed field. However, it is pointed out that the smoothing operation should be applied up to the level that is necessary to effectively suppress the parasitic currents, as it generally tends to level down regions of the interface having high curvature. It is noted that the smoothed VOF distribution $\tilde{\alpha}_i$ is only used in the evaluation of the curvature (see Eq. (21)) and the regular VOF function α_i is used in all other equations.

2.2.3. Interface sharpening

In the new multiFluidInterFoam solver, the fluid interfaces are also sharpened by introducing artificial compression terms into Eq. (4) in a similar manner as in the original multiphaseInterFoam solver. For consistency reasons, though, the artificial compression velocity $\mathbf{u}_{r,ik}^*$ is now calculated as

$$(\mathbf{u}_{r,ik}^*)_f = (\mathbf{n}_i)_f \min \left[C_{\alpha_i} \frac{|\phi_f^*|}{|S_f^*|}, \max \left(\frac{|\phi_f^*|}{|S_f^*|} \right) \right], \quad (23)$$

$i = 1, 2, \dots, N_c$ and $k \neq i$,

where the normal vector $(\mathbf{n}_i)_f$ is used instead of $(\mathbf{n}_{ik})_f$, and it is computed as

$$(\mathbf{n}_i)_f = \frac{(\nabla \alpha_i)_f}{|(\nabla \alpha_i)_f| + s_f}, \quad (24)$$

similarly to the evaluation of the normal vector in the calculation of the continuum surface force Eq. (20) in the new solver.

2.2.4. SIMPLEC velocity–pressure coupling

As it will be discussed below in detail, the implemented version of the PIMPLE algorithm for the velocity–pressure coupling in the multiPhaseInterFoam solver was not sufficiently stable for transient viscoplastic simulations in the floating lens, drop levitation and rising bubble problems. In order to avoid numerical instabilities or divergence, we found that it was necessary to (a) under-relax pressure and

velocity fields, (b) use a sufficiently large number of inner iterations performing the velocity–pressure coupling, or (c) use a very small time-step. The first option is not efficient for transient simulations, while the second and third ones lead to a burden on the computational cost.

The SIMPLEC (Semi-Implicit Method for Pressure-Linked Equations-Consistent), algorithm [29] in opposition to the SIMPLE, does not require under-relaxation of pressure and velocity, except for simulations with non-orthogonal grids. Additionally, its computational cost per iteration is lower than that in the PISO algorithm, because the pressure equation is only solved once per cycle. In SIMPLEC, the velocity correction at a given cell can be approximated by the coefficients-weighted average of the velocity corrections of its neighbor cells. This turns out to make the coupling scheme more stable, especially in the non-Newtonian flows, allowing a greater values of the time step (for more details, see Pimenta and Alves [30]). For these reasons, the SIMPLEC algorithm was adopted in the present work. Note that the SIMPLEC algorithm is already available in the OpenFOAM as a possible steady-state solver for (in)compressible, inelastic fluids. The SIMPLEC method is implemented in the new multiphase solver in a similar manner as in the solvers provided by the RheoTool [15] toolbox and, thus, the details of its implementation are not repeated here.

2.3. Overview of the rheoMultiFluidInterFoam solver

The rheoMultiFluidInterFoam solver is an extension of the multiFluidInterFoam solver, in which the library containing the non-Newtonian fluid models of RheoTool [15] has been fully incorporated enabling the modeling of a wide variety of complex fluids, e.g. exhibiting viscoelastic, viscoplastic or elasto-viscoplastic effects. It is recalled that for the purposes of the present work we mainly focus on the simulation of Newtonian, viscoplastic and viscoelastic fluids.

2.3.1. Viscoplastic material - Herschel–Bulkley model

A viscoplastic material deforms as non-Newtonian viscous fluid when the local stress exceeds some critical value, whereas it behaves as a solid at low values of the stress. The discontinuous Herschel–Bulkley constitutive equation is commonly used for modeling a viscoplastic fluid without accounting for the elasticity of the material and is given by

$$\underline{\boldsymbol{\tau}}^* = (\tau_0^* / \dot{\gamma}^* + k^* (\dot{\gamma}^*)^{n-1}) \dot{\boldsymbol{\gamma}}^*, \quad |\underline{\boldsymbol{\tau}}^*| \geq \tau_0^*, \quad (25)$$

$$\dot{\boldsymbol{\gamma}}^* = 0, \quad |\underline{\boldsymbol{\tau}}^*| < \tau_0^*, \quad (26)$$

where $|\underline{\boldsymbol{\tau}}^*| = \sqrt{(\boldsymbol{\tau}^* : \boldsymbol{\tau}^*)/2}$; the exponent n is the power law index and k^* is a consistency factor. In order to overcome the discontinuity introduced in the constitutive equation by the separation of yielded and unyielded regions, commonly viscosity regularization models and methods based on variational principles have been used [31,32].

The most famous variational approach for the solution of viscoplastic problems is the Augmented Lagrangian method [33–36], allowing one to cope with the exact discontinuous law. A popular alternative to the augmented Lagrangian approach are regularization models, where the discontinuous viscosity function is smoothed in order to asymptotically match the original model. Arguably the most commonly used approximation among them is the Papanastasiou’s regularization [37–39]. The main advantage of this approach is that it is applied easily in fluid flow solvers. Despite the simplicity of the Papanastasiou regularization model and the fact that it is generally found to be superior than bi-viscosity models [40], it is not currently available in OpenFOAM. It is provided, though, by the RheoTool, which is incorporated in the new Non-Newtonian multiphase solver, presented herein. The Papanastasiou regularization model can be written as

$$v_i^* (\dot{\gamma}^*) = \min \left(v_{0,i}^*, \frac{\tau_{0,i}^*}{\dot{\gamma}^*} [1 - \exp(-m_i^* \dot{\gamma}^*)] + k_i^* (\dot{\gamma}^*)^{n_i-1} \right), \quad (27)$$

where m_i^* is an additional parameter related to the exponential term. It is noted that the original Papanastasiou regularization does not include the artificial upper-bounding by $v_{0,i}^*$. However, this bounding is needed in order to avoid an infinite viscosity for $\dot{\gamma}^* \rightarrow 0$ (e.g. startup of flow) and $n_i < 1$. The original Papanastasiou regularization is recovered for $v_{i,0}^* \rightarrow \infty$. In practice, $v_{0,i}^*$ should be low enough to avoid an infinite viscosity in quiescent conditions and high enough to allow Papanastasiou regularization to take control in the remaining situations. It is noted that the model Eqs. (8) and (9) used in both the original multiPhaseInterFoam solver and the new multiFluidInterFoam solver are also available in the rheoMultiFluidInterFoam solver through the RheoTool toolbox.

We note that the rheoMultiFluidInterFoam solver adopts a different approach for the calculation of the deviatoric stress tensor $\underline{\tau}^*$ in Eq. (2) with respect to the multiFluidInterFoam and multiPhaseInterFoam solvers. In Eq. (3) used in the latter solvers, a bulk density and a bulk kinematic viscosity are required that are calculated as weighted averages of the density and viscosity of the involved generalized Newtonian fluids, respectively. Currently, the rheoMultiFluidInterFoam solver solves a constitutive equation for each phase and the extra-stress tensor contributing to the momentum equation is the weighted average of the extra-stress tensor for each phase $\underline{\tau}_i^*$ as follows

$$\underline{\tau}^* = \sum_i \alpha_i \underline{\tau}_i^*. \quad (28)$$

2.3.2. Viscoelastic material - Phan-Thien Tanner model

Several constitutive equations are available in RheoTool to model the viscoelastic properties of complex fluids. In this work, the linear Phan-Thien Tanner (PTT) viscoelastic model was chosen in order to assess the performance of the newly developed solver to predict complex fluid flows. This model reproduces both the elastic behavior and the dependence of the shear and extensional viscosity on local rate of deformation of several fluids. To simulate viscoelastic fluid flows, in particular, the total extra-stress tensor is usually split into a solvent contribution ($\underline{\tau}_s^*$) and a polymeric contribution ($\underline{\tau}_p^*$), so that Eq. (28) can be re-written as

$$\underline{\tau}^* = \sum_i \alpha_i (\underline{\tau}_{p,i}^* + \underline{\tau}_{s,i}^*). \quad (29)$$

A constitutive equation is required for each tensor contribution, which for the linear PTT model can be written as

$$\underline{\tau}_{s,i}^* = \mu_{s,i}^* (\dot{\gamma}^*) \underline{\dot{\gamma}}^*, \quad (30)$$

$$\left[1 + \frac{\epsilon_i \lambda_i^*}{\mu_{p,i}^*} \text{tr}(\underline{\tau}_{p,i}^*) \right] \underline{\tau}_{p,i}^* + \lambda_i^* (\dot{\gamma}^*) \underline{\dot{\tau}}_{p,i}^* = \mu_{p,i}^* (\dot{\gamma}^*) \underline{\dot{\gamma}}^*, \quad (31)$$

where $\mu_{s,i}^*$ is the solvent viscosity, $\mu_{p,i}^*$ is the polymeric viscosity coefficient, λ_i^* is the relaxation time, ϵ_i is the extensibility parameter controlling the degree of shear and extensional thinning, and

$$\underline{\dot{\tau}}_{p,i}^* = \underline{\dot{\tau}}_{p,i}^* + \frac{\zeta}{2} (\underline{\tau}_{p,i}^* \cdot \underline{\dot{\gamma}}^* + \underline{\dot{\gamma}}^* \cdot \underline{\tau}_{p,i}^*) \quad (32)$$

is the Gordon–Schowalter derivative, which is controlled by the parameter ζ in order to take non-affine deformation into account. The term

$$\underline{\dot{\tau}}_{p,i}^* = \frac{\partial \underline{\tau}_{p,i}^*}{\partial t^*} + \mathbf{u}^* \cdot \nabla \underline{\tau}_{p,i}^* - \underline{\tau}_{p,i}^* \cdot \nabla \mathbf{u}^* - \nabla \mathbf{u}^{*T} \cdot \underline{\tau}_{p,i}^* \quad (33)$$

represents the upper-convected time derivative, which renders the model frame-invariant. The log-conformation approach is utilized in order to solve numerically Eq. (31). For more details, please refer to Ref. [15].

2.4. Numerical methods and details

In the developed multiphase solvers, the time discretization is performed by using a second-order Crank–Nicolson scheme, except for the phase fraction equations in which a first-order Euler method is used. The space discretization is based on a second-order linear scheme. The advection of velocity is discretized by utilizing the *limitedLinearV 1* scheme, while the *vanLeer01* and *interfaceCompression* schemes are used for the phase fractions. The resulting linear system of equations is solved by using the Preconditioned bi-conjugate gradient solver with Diagonal incomplete-LU (PBiCG/DILU) for velocity, the Pre-conditioned Conjugate Gradient solver combined with Diagonal Incomplete Cholesky (PCG/DIC) for pressure and pressure correction, and a smoothSolver/Gauss–Seidel solver for the phase fractions. For all variables, the tolerance was set equal to 10^{-12} , whereas no under-relaxation was used. The time step was determined using a global and local Courant numbers of 0.5, while the maximum allowed time step was determined as the smallest obtained from the equations [18,22]

$$\Delta t_{c,i}^* = C_{c,i} \sqrt{\rho_i^* \Delta x^{*3} / |\Sigma_i^*|} \quad i = 1, 2, 3, \quad (34)$$

$$\Delta t_{v,i}^* = C_{v,i} \mu_i^* \Delta x^* / |\Sigma_i^*| \quad i = 1, 2, 3, \quad (35)$$

where $C_{c,i} = C_{v,i} = 0.5$. The above settings were used in this work unless otherwise stated.

2.5. Overview of the new solvers

The computational steps of the new multiphase solvers are as follows:

1. Initialize all the fields $\{p^*, \mathbf{u}^*, \alpha_i\}_0$ and time ($t^* = 0$)
2. Enter the time loop ($t^* = \Delta t^*$)

(a) Enter the inner iterations loop ($it = 0$)

- i. Calculate the Courant numbers and adjust the time step based on the imposed restraints if necessary
- ii. Solve the $\alpha_i^{(it)}$ equations by using the volumetric fluxes of the previous time level
- iii. For a generalized Newtonian fluid, use the appropriate constitutive equations together with the new values of $\alpha_i^{(it)}$ in order to update the estimates for the cell center viscosity $\mu^{*(it)}$ and density $\rho^{*(it)}$ and the face viscosity $\mu_f^{*(it)}$ and density $\rho_f^{*(it)}$ of the multiphase mixture. For a viscoelastic material, solve a differential constitutive equation for each phase and compute the extra-stress tensor contributing to the momentum equation as the weighted average of the extra-stress tensor for each phase by using the indicator VOF function $\alpha_i^{(it)}$; the latter approach can be used for every rheological model described by a differential constitutive equation, e.g. for elasto-viscoplastic materials
- iv. Based on the above values, solve the momentum equation to obtain a velocity prediction $\mathbf{u}^{*(it)}$
- v. Continue with the pressure-correction algorithm (SIMPLEC) and solve the pressure equation to obtain a continuity-compliant pressure field, $p^{*(it)}$
- vi. Correct both the face $\hat{\mathbf{u}}_f^{*(it)}$ and cell-centered $\hat{\mathbf{u}}^{*(it)}$ velocities using the correct pressure field
- vii. Increment the inner iteration index ($it = it + 1$) and return to step *i* until the pre-defined number of inner iterations is reached
- viii. Set $\{p^*, \mathbf{u}^*, \alpha_i\}_t = \{p^{*(it)}, \mathbf{u}^{*(it)}, \alpha_i^{(it)}\}$

(b) If the final time has not been reached, advance to the next time level $t = t + \Delta t$ and return to step 2(a)

3. Stop the simulation and exit

The only difference between the two new solvers is in step 2(a)iii, in which the `multiFluidInterFoam` solver uses constitutive relations like Eqs. (8) and (9) already available in `OpenFOAM`, whereas the `rheoMultiFluidInterFoam` solver employs constitutive relations given by the `RheoTool`. In the above numerical algorithm, there is a main loop advancing in time, step 2, and an inner loop, step 2(a). The purpose of the inner iterations is to reduce the explicitness of the numerical method, which is crucial to enhance both the stability and the time accuracy of the solver. The lastly computed value of a variable is used in the inner loop, either from previous iteration of the inner loop or from the previous time step.

The main changes introduced relative to the original `multiPhaseInterFoam` solver can be summarized as follows: (a) the interfacial tension coefficient decomposition method is employed to deal with tension pairings between different faces using a compositional approach, and the resulting interfacial force is reformulated properly as the sum of each phase body force, (b) VOF smoothing is implemented to estimate accurately the curvature of interfaces, (c) the velocity–pressure coupling is utilized based on the SIMPLEC algorithm, and (d) the `RheoTool` toolbox has been implemented in the new solver to make use of several constitutive models for complex fluids.

3. Results

We begin our study in Section 3.1 with the investigation of the behavior of our new solver `multiFluidInterFoam`. To this end, the results obtained by the new solver are compared against those yielded by the original `multiPhaseInterFoam` solver for three benchmark flow problems: (a) the three-phase flow in a dam break with an obstacle, (b) the spreading of a droplet between two stratified fluids, and (c) the levitation of a droplet between two stratified fluids. Case (a) is a slightly modified case with respect to the standard `damBreak4Phase` tutorial (i.e. a flow case with four fluid phases) available in `OpenFOAM`.

Next, we proceed in Section 3.2 with the validation of the `rheoMultiFluidInterFoam` solver. Given that there are no available benchmarks in the literature for Non-Newtonian three-phase flows and in order to validate the code and the proper implementation of `RheoTool`, we consider the limiting case of the two-phase flow of a Newtonian bubble rising in (a) a viscoplastic fluid described by the Herschel–Bulkley model and (b) a viscoelastic fluid described by the Phan-Thien Tanner model. The former case is useful since it is possible to perform a comparison between the two new multiphase solvers when using the bi-viscosity version of the Herschel–Bulkley model, while it is also possible to assess the numerical implementation of the Herschel–Bulkley model based on the `RheoTool`. On the other hand, the latter case provides a representative test for the implementation of differential constitutive equations provided by the `RheoTool`.

3.1. Consistency and verification tests for the `multiFluidInterFoam` solver

3.1.1. Dam break with obstacle

The newly developed solver is capable of predicting the multiphase flow of three or more immiscible fluids by capturing moving interfaces on an Eulerian mesh. Merging or breakup of the interfaces is also handled in a natural way. A modified and refined version of the standard dam break tutorial of `OpenFOAM` is used here as an initial investigation of the behavior of our new solver; for our comparison we consider the case of a three-phase system.

The considered computational domain is a square box having width and height of 0.584 m, as shown in Fig. 1. A small rectangular obstacle of size of 0.024 m × 0.048 m is located on the bottom wall at a distance of 0.292 m from the left wall. Phases 1 (water, blue color) and 2 (oil, green color) are initially placed in square columns at the left side of

the domain, while the rest of the box is filled with phase 3 (air, red color). The dynamic viscosity and density of phase 1 are 10^{-3} Pa s and 1000 kg m^{-3} , respectively, while those of phases 2 and 3 are 5×10^{-4} Pa s, 500 kg m^{-3} and 1.48×10^{-5} Pa s, 1 kg m^{-3} , respectively. The surface tension coefficients for all interfaces are set to 0.07 N m^{-1} . The magnitude of gravity acceleration is $g^* = 9.81 \text{ m s}^{-2}$.

No-slip boundary condition is applied to all boundaries except for the top surface, which is assumed to be an open boundary. Zero gradient is applied at the walls for each volume fraction, while at the open boundary all fractions are assigned to be zero except for that of phase 3 which is set to unity. The resolution of the computational mesh is 92×92 control volumes. All the simulations use the same computational mesh and discretizations schemes summarized in Section 2.4. The time step is determined using a global and local Courant numbers of 0.5, while the maximum allowed time step is set equal to $\Delta t^* = 10^{-4}$ s. The interface compression parameter C_{α_i} is set to unity. No smoothing in the calculation of curvature is applied in these simulations to facilitate the comparison with the original multiphase solver.

The initial condition is given in Fig. 1(a), (b), (c). The simulations were performed based on the new `multiFluidInterFoam` solver by using either the SIMPLEC (see Fig. 1(a), (d), (g), (j)) or the PIMPLE (see Fig. 1(b), (e), (h), (k)) pressure–velocity coupling algorithms, and the original `multiPhaseInterFoam` solver (see Fig. 1(c), (f), (i), (l)) for a time period of 1 s. It can be seen that all solvers give almost the same prediction for the distribution of the phases, as shown in Fig. 1(d)–(i). As time proceeds, the three phases change their shape and position because of density differences. At time $t^* = 0.25$ s, phase 1 has just stumbled over the obstacle on the floor, an air bubble is trapped adjacent to the obstacle, whereas phase 2 has hit the right wall and moves upward. At time $t^* = 0.5$ s, the air bubble is located within a larger drop of phase 2, while phase 1 has reached the right wall. Overall, we observe that the qualitative characteristics of the distribution of phases are similar in all three simulations, with small differences arising only at the very late stages of the simulation, at $t^* = 1$ s. This kind of multiphase flow depends mostly on the density differences between the phases, which generate sufficient momentum so that any surface tension effects are rather small. By using the height of the fluid columns as a length scale $H^* = 0.292$ m, a proper characteristic velocity scale can be determined as $(\rho_1^* - \rho_2^*)/\rho_2^*(g^*H^*)^{1/2}$ and, thus, the Weber number can be estimated as $We = \rho_1^*U^{*2}H^*/\sigma_{12}^* \approx 1.2 \times 10^4$, where ρ_1^* , ρ_2^* are the density of phases 1 and 2, respectively, and σ_{12}^* is the interface tension coefficient of the interface between fluids 1 and 2. The Weber number measures the relative magnitude of the inertia and surface tension forces, and the presently high We value indeed suggests that any surface tension effects are negligible and the phenomenon is driven mostly by the inertia forces. Clearly in this example, the contact point, that is observed at the early times of the simulation and moves toward the left wall, do not control the dynamics of the multiphase flow, which exhibits interfaces mainly between two phases.

3.1.2. Spreading of a droplet between two stratified Newtonian fluids

The spreading of a liquid lens is a very classic benchmark problem and it has been widely used previously to assess the numerical methods for three-phase flows [2,5,7–9,41]. Here, we examine this benchmark problem in detail and consider the cases of partial and full spreading of the droplet in order to investigate whether the newly developed numerical solver is capable of accurately predicting the dynamics of spreading and also producing the correct equilibrium contact angles with triple junction points.

Problem description

The geometry and relevant variables for the spreading of a droplet between two stratified fluids are shown in Fig. 2. We consider the case of a viscous fluid 2 of density ρ_2^* and viscosity μ_2^* spreading between two fluid layers of density ρ_1^* , ρ_3^* and viscosity μ_1^* , μ_3^* . The bottom and upper layers are bounded from below and above as well as from left and

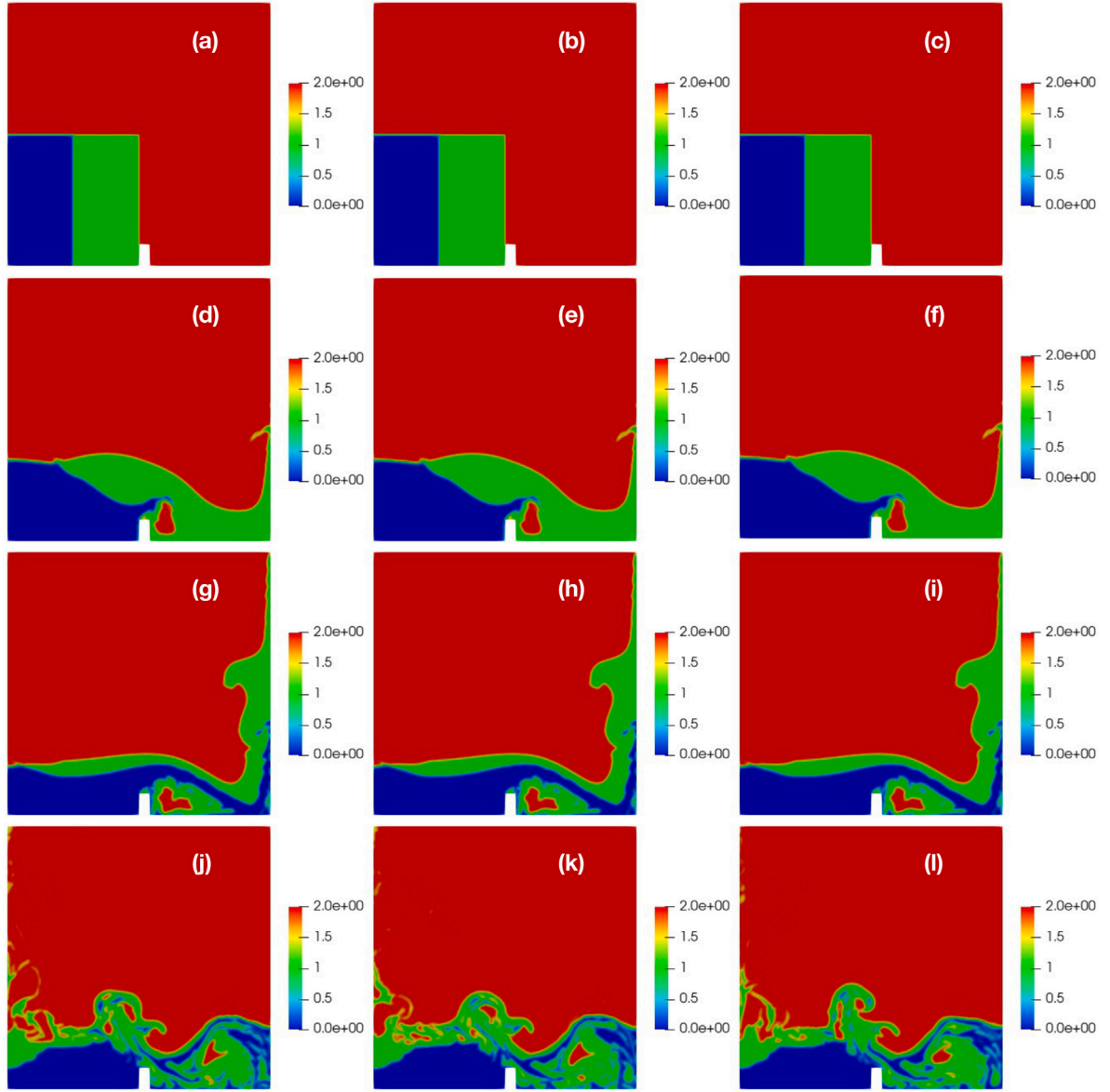


Fig. 1. Distribution of phase 1 (water, blue color), phase 2 (oil, green color) and phase 3 (air, red color) for the problem of dam break with an obstacle at times $t^* = 0$ s [(a), (b), (c)], $t^* = 0.25$ s [(d), (e), (f)], $t^* = 0.5$ s [(g), (h), (i)], and $t^* = 1$ s [(j), (k), (l)]. The results are obtained by using the new multiFluidInterFoam solver with the SIMPLEC method (left column) or the PIMPLE method (middle column) for velocity–pressure coupling, and the original multiphaseInterFoam solver (right column).

right by impermeable, rigid, and solid substrates. The surface tension of the interface between fluids i and j is denoted with σ_{ij}^* ($i = 1, 2, 3$ and $j = 1, 2, 3$). All fluids are assumed to be incompressible, Newtonian and immiscible.

Initially, a circular droplet (fluid 2) with radius R_d^* and area A^* is placed between the interface of the lower fluid 1 and upper fluid 3 with zero velocity. The center of the droplet is located at the point (X_c^*, Y_c^*) . No-slip boundary condition is applied along the walls, whereas a symmetry condition is applied at the left boundary; therefore, the computational domain consists of half of the square box shown in Fig. 2. The characteristic scales of length, velocity, time and pressure are denoted as L^* , U^* , U^*/L^* and ρ^*U^{*2} , respectively. The total height of the fluid layers is chosen as a proper length scale L^* , while the characteristic velocity is defined as $U^* = \mu_1^*/(\rho_1^*L^*)$. Here, the gravity effects are not taken into account. The problem is characterized by the non-dimensional density, viscosity, and surface tension coefficients given as $\rho_i = \rho_i^*/\rho_1^*$, $\mu_i = \mu_i^*/\mu_1^*$, and $\sigma_{ij} = \sigma_{ij}^*/\sigma_{23}^*$, $\Sigma_i = \Sigma_i^*/\Sigma_2^*$ ($i, j = 1, 2, 3$), respectively. The Reynolds number is defined as $Re = \rho_1^*U^*L^*/\mu_1^*$, and the Reynolds number for each i phase as $Re_i = \rho_i/\mu_i Re$. The Weber

number is given as $We = \rho_1^*L^*U^{*2}/\sigma_{23}^*$, and the Weber number for each ij interface as $We_{ij} = \rho_i/\sigma_{ij}We$, while the phasic Weber number is given as $We_p = \rho_1^*L^*U^{*2}/\Sigma_2^*$ and the phasic Weber number based on fluid i as $We_{p,i} = \rho_i/\Sigma_iWe_p$.

Two different spreading phenomena can be observed depending on the values of the interfacial tensions, i.e., partial spreading and full spreading. This can be categorized based on the spreading parameter of the phase i at the interface between phases j and k [42–44]

$$S_i^* = \sigma_{jk}^* - \sigma_{ij}^* - \sigma_{ik}^*. \quad (36)$$

If S_i^* is negative ($S_i^* < 0$), the spreading is said to be partial. In this case, each of the three interfacial tension coefficients is less than the sum of the other two coefficients and, thus, stable triple points can be observed in regions where the three fluids meet. On the other hand, if S_i^* is positive ($S_i^* > 0$), the spreading is said to be full or complete. In this case, one interfacial tension is so large that the sum of the other two cannot balance it. The triple points become unstable and one fluid will spread in a such a way that the triple points disappear. Then, the system tends to an equilibrium state in which the interfaces are finally

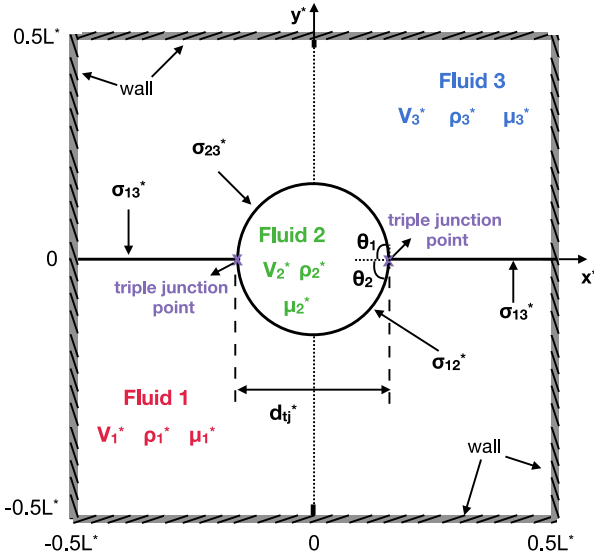


Fig. 2. Geometry and initial setup of the spreading of a droplet between two stratified fluids problem.

either plane or spherical, depending on the combination of values of the interfacial tension coefficients.

Partial spreading

We first consider a case resulting in partial spreading, for which the three interfacial tensions yield a Neumann's triangle. In partial spreading, the droplet can eventually reach a steady lens shape having geometrical characteristics that can be analytically calculated [45–47]. For the cases presented here, we consider a droplet with initial radius of $R_d^* = 150 \mu\text{m}$ and it is placed at $(X_c^*, Y_c^*) = (0, 0)$, while the characteristic length scale is $L^* = 1 \text{ mm}$. All fluids have the same density and dynamic viscosity and therefore $\rho_i = \mu_i = 1$ ($\rho_1^* = 1000 \text{ kg m}^{-3}$, $\mu_1^* = 1 \text{ Pa s}$) for $i = 1, 2, 3$. The Reynolds number is $Re = Re_1 = Re_2 = Re_3 = 1$. The phasic Weber number corresponding to the phase 2 is kept constant to $We_{p,2} = 60$ while we consider three different cases of increasing interfacial tension with $We_{p,1} = We_{p,3} = 108$ ($\sigma_{23}^* = 0.026 \text{ N m}^{-1}$), 60 ($\sigma_{23}^* = 0.033 \text{ N m}^{-1}$), 36 ($\sigma_{23}^* = 0.044 \text{ N m}^{-1}$). We also vary the parameters C_{α_i} and n_{lfi} to examine the effect of interface sharpening and smoothing operation in the calculation of curvature, respectively. All the simulations use the same computational mesh, discretizations schemes and numerical parameters. The computational mesh consists of 100×200 control volumes, unless otherwise stated.

In Fig. 3, we show the equilibrium droplet shapes for three different cases (i.e. A1, A2 and A3) as predicted by our new multiFluidInterFoam solver for various combinations of the interface compression parameter C_{α_i} and the Laplacian filter parameter n_{lfi} .

Our solutions are compared with those obtained by using the standard multiphase solver multiPhaseInterFoam as well as with the numerical results of Xie et al. [5] and Kim and Lowengrub [48] produced by codes other than the OpenFOAM. The methods used in the aforementioned works are different from the present work. For instance, Xie et al. [5] have used a control volume/finite element VOF method, while Kim and Lowengrub [48] have utilized a phase field method. Moreover, the results adopted from these studies in the comparisons are grid converged. Case A1 corresponds to $We_{p,1} = We_{p,2} = We_{p,3} = 60$ and is shown in Fig. 3(a), (d); Similarly, in Fig. 3(b), (e) we depict the predictions for Case A2 with $We_{p,2} = 60$ and $We_{p,1} = We_{p,3} = 36$, and we show the results for Case A3 with $We_{p,2} = 60$ and $We_{p,1} = We_{p,3} = 108$ in Fig. 3(c), (f). In the panels (a)–(c), the numerical parameters associated with our solver are $C_{\alpha_i} = 0$

and $n_{lfi} = 0, 1$, whereas the panels (d)–(f) correspond to $C_{\alpha_i} = 1$ and $n_{lfi} = 0, 1$.

The initially circular droplet spreads and its final shape, depicted in Fig. 3, depends on the values of the surface tension coefficients of each system. It can be seen that the newly developed solver is capable of reproducing the correct trends in the droplet shape at the steady-state regime with respect to the overall surface tension effect, as indicated by the close agreement of our predictions with other published numerical results [5,48]. This is valid for all combinations of the parameters C_{α_i} and n_{lfi} and it is also noted that our results appear to be in better agreement with the predictions of the more accurate scheme of Xie et al. [5]. Turning our attention to the predictions of the original multiPhaseInterFoam solver, also shown in Fig. 3 for $C_{\alpha_i} = 0$ (see Fig. 3(a–c)) and $C_{\alpha_i} = 1$ (see Fig. 3(d–f)), we find large discrepancies relative to the both previously published results and our predictions, clearly pointing out a much lower accuracy of the original multiphase solver. These discrepancies appear to be unaffected by the inclusion of interface compression. A closer examination of Fig. 3 also reveals that the latter predictions exhibit a very weak dependency on the surface tension coefficients, in clear disagreement with the rest numerical schemes.

The interface compression parameter C_{α_i} and the Laplacian filter parameter n_{lfi} have a small but not negligible effect on the predicted droplet shape by our numerical scheme, as shown in Fig. 3. In particular, the application of filtering with $n_{lfi} = 1$ leads to an increase of the drop's length accompanied by a reduction to the maximum height of the drop due to the conservation of its mass and area. This turns out to be more evident in the cases with $C_{\alpha_i} = 1$. On the other hand, the interface compression parameter seems to have exactly the opposite effect to that of the filtering operation. A reduction of the drop's length is apparent in Fig. 3 when comparing cases without filtering $n_{lfi} = 0$ having $C_{\alpha_i} = 0$ (upper row) with $C_{\alpha_i} = 1$ (lower row). The effect of these parameters on the solution accuracy will also be discussed in more detail below.

The better performance of the new multiFluidInterFoam solver is also illustrated in Fig. 4, which shows the time evolution $t (= t^* U^* / L^*)$ of the non-dimensional lens' length $d_{lj} (= d_{lj}^* / L^*)$ for the same three cases (A1, A2 and A3). In this figure, we depict the prediction of our solver (solid lines) along with the results of the regular multiPhaseInterFoam solver (solid lines with points). In addition, the dashed line depicts the analytical solution for the equilibrium lens' length; note that the derivation of an analytical solution for the equilibrium lens area, its length and the equilibrium contact angle is possible when the values of σ_{ij}^* are specified. When a full drop is considered, the distance between the two triple junction points (denoted by d_{lj}^* in Fig. 2) can be calculated using the following expression [45–47]:

$$d_{lj}^* = \sqrt{\frac{4A^*}{\sum_{l=1}^2 \frac{1}{\sin \theta_l} \left(\frac{\theta_l}{\sin \theta_l} - \cos \theta_l \right)}}, \quad (37)$$

where A^* is the area of the droplet and θ_l ($l = 1, 2$) is the contact angle as illustrated in Fig. 2, determined as

$$\cos((\pi - \theta_2) - \theta_1) = \frac{\sigma_{13}^{*2} + \sigma_{23}^{*2} - \sigma_{12}^{*2}}{2\sigma_{23}^* \sigma_{13}^*}, \quad \cos \theta_1 = \frac{\sigma_{13}^{*2} + \sigma_{12}^{*2} - \sigma_{23}^{*2}}{2\sigma_{12}^* \sigma_{13}^*}. \quad (38)$$

After a transient time period, all numerical solutions provided by our solver approach the expected analytical solutions. Evidently, our solver predicts very well the correct long-term solution, for all cases (A1, A2, and A3) which correspond to different values of the interfacial tension coefficients. On the other hand, the original multiphase solver of OpenFOAM seems to be insensitive to the variation of the interfacial tension coefficients, showing almost the same transient behavior with varying $We_{p,1}$ and $We_{p,3}$ and providing an obviously wrong prediction for the long-term solution.

Turning back our attention to the predictions of the present numerical scheme, we observe in Fig. 4 that the predicted dynamics can be

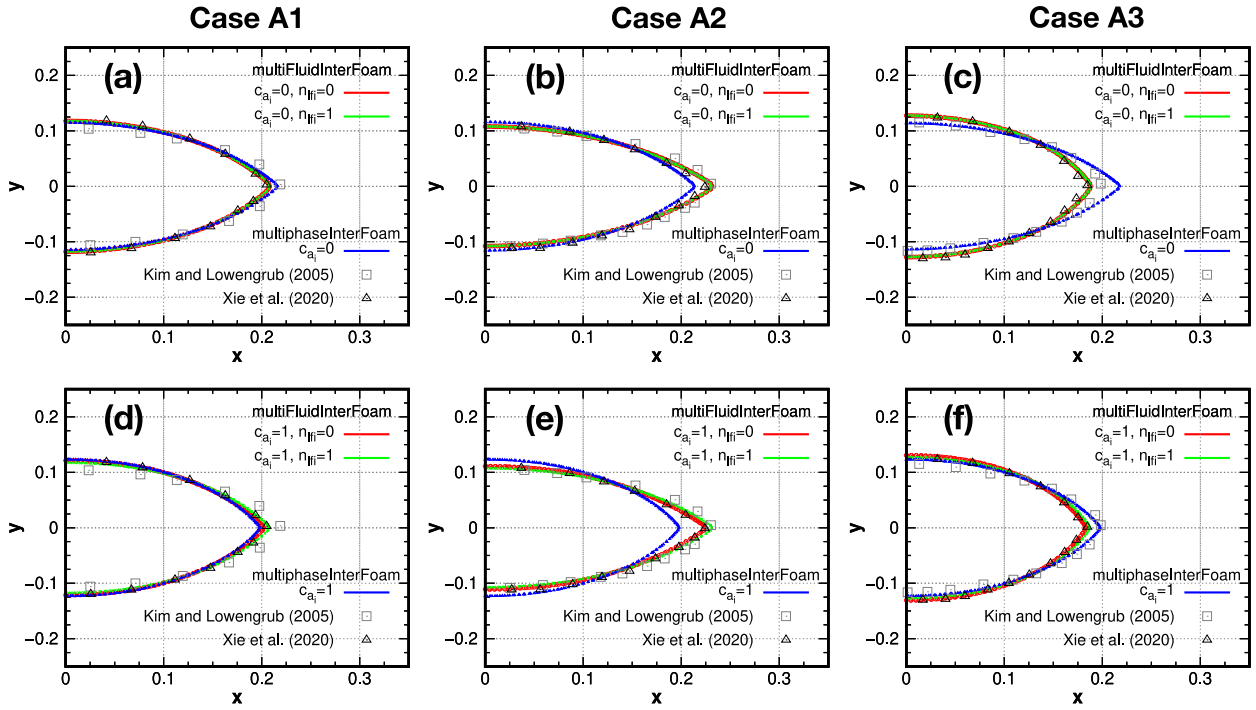


Fig. 3. Equilibrium shape of the droplet (corresponding to $\alpha_2 = 0.5$ isoline) predicted by the new `multiFluidInterFoam` solver for Case A1 with $We_{p,1} = We_{p,2} = We_{p,3} = 60$ (left column), Case A2 with $We_{p,2} = 60$ and $We_{p,1} = We_{p,3} = 36$ (middle column), and Case A3 with $We_{p,2} = 60$ and $We_{p,1} = We_{p,3} = 108$ (right column). Simulations are performed with various combinations of the interface compression parameter and the filtering parameter $C_{\alpha_i} = 0$, $n_{lfi} = 0.1$ [(a), (b), (c)] and $C_{\alpha_i} = 1$, $n_{lfi} = 0.1$ [(d), (e), (f)]. The results are compared with the numerical results of Xie et al. [5] and Kim and Lowengrub [48] and those obtained by the regular `multiphaseInterFoam` solver. The density and viscosity ratios are $\rho_i = 1$ and $\mu_i = 1$, respectively.

affected to some extent by the values of the C_{α_i} and n_{lfi} parameters and, therefore, their role and effect in the accuracy of the proposed solver should be discussed in more detail. In particular, we see in this figure that for $C_{\alpha_i} = 1$ and $n_{lfi} = 0$ (i.e. with interface compression enabled and filtering disabled) the numerical scheme fails to predict the correct long term solution. This result provides an indication that the filtering of the volume fractions is essential in order to improve the quality of the predictions when interface compression is taken into account. Moreover, the time evolution of d_{ij} when $C_{\alpha_i} = 0$ differs significantly from that when surface compression is enabled (with $C_{\alpha_i} = 1$), and the long-time solutions obtained in these cases do not coincide with each other.

To obtain a clearer picture regarding the effect of the parameters C_{α_i} and n_{lfi} on the numerical prediction of the drop shape, the relative error of the drop's length was computed as $e_{d_{ij}} = (d_{ij,n} - d_{ij,a})/d_{ij,a}$ and it is shown in Fig. 5, where $d_{ij,n}$ and $d_{ij,a}$ are the values calculated numerically and analytically based on Eq. (37) [45–47], respectively. Results for three computational meshes of 20×40 , 50×100 , and 100×200 are presented. A number of observations can be made; at first, the relative error is reduced as the mesh is refined, with the exception of case ($C_{\alpha_i} = 1$, $n_{lfi} = 0$), in which $|e_{d_{ij}}|$ seems to saturate at approximately 2% as the grid spacing is reduced. In all the other cases, the value of $|e_{d_{ij}}|$ is below 1%. Secondly, it turns out that the adopted grid resolution of 100×200 is so fine and the time step is also small enough, so that the effect of spurious currents are rather insignificant and the results obtained with ($C_{\alpha_i} = 0$, $n_{lfi} = 0$) are adequate. Thirdly, the improvement of the predictions when applying only filtering or when combining the sharpening and filtering operations is more evident for the smaller mesh of 20×40 control volumes. The absolute error difference $\Delta(e_{d_{ij}})^{C_{\alpha_i}, n_{lfi}} = (e_{d_{ij}})^{C_{\alpha_i}, n_{lfi}} - (e_{d_{ij}})^{0,0}$ can be used for quantification purposes. For example, it is $\Delta(e_{d_{ij}})^{0,1} = 0.919$, 0.551 , and 0.056 for the meshes 20×40 , 50×100 , and 100×200 , respectively, while it holds that $\Delta(e_{d_{ij}})^{1,1} = 1.579$ (20×40), 0.77 (50×100), and 0.324 (100×200). It is worth mentioning that the opposite is observed

for the effect of the C_{α_i} parameter, as it is $\Delta(e_{d_{ij}})^{1,0} = 0.188$, 1.114 , and 2 for non-dimensional grid spacings of $\Delta x = 1/40$, $1/100$, and $1/200$, respectively. This points out that the optimum C_{α_i} depends on the grid resolution, revealing that there is no need for application of excessive levels of artificial sharpening of interfaces since the interfaces are well captured as the grid is refined. The above discussion suggests a stronger impact of the filtering and combined filtering/sharpening operations in coarser simulations. Finally, the worst and best performances are found for the parameter combination ($C_{\alpha_i} = 1$, $n_{lfi} = 0$) and ($C_{\alpha_i} = 1$, $n_{lfi} = 1$), respectively. A better agreement of the present numerical solutions with the analytical solutions and previous data of Xie et al. [5] and Kim and Lowengrub [48] is generally observed when using the new solver with $C_{\alpha_i} = 1$ and $n_{lfi} = 1$.

In Table 1, we compare the length d_{ij} predicted by the numerical simulations and compare it against both the analytical solutions and two previous works [5,48]. The results of the new solver compares favorably with both the analytical solution and the results of Xie et al. [5] and Kim and Lowengrub [48]. A quantitative comparison with the analytical solution reveals that the relative error of the `multiFluidInterFoam` solver is 0.18%, 1%, and 0.28% for Cases A1, A2, and A3 respectively, providing further confirmation of the excellent accuracy of the present solver. On the other hand, using the exact same grid of 100×200 control volumes, the relative error for the original `multiphaseInterFoam` solver is 4.7% (Case A1), 13.54% (Case A2), and 5.62% (Case A3). Our solver achieves the same or better level of accuracy with respect to the methods of Xie et al. [5] and Kim and Lowengrub [48], as also indicated in Table 1. It is pointed out that the present results are obtained with the use of a slightly coarser non-dimensional grid spacing of $\Delta x = 1/200$ as compared with $\Delta x = 1/256$ adopted in the aforementioned works.

Finally, in order to assess how the predictions of the new solver are affected by the selected grid, a convergence study is presented below. Beside the standard adaptive mesh refinement algorithm provided by OpenFOAM, the present solver also employs a proper adaptation of this

Table 1

A comparison of the non-dimensional equilibrium distance between triple junctions d_{ij} in three representative cases obtained from analytical solutions, the present solver ($C_{a_i} = 1, n_{l_{f_i}} = 1, \Delta x = 1/200$), the original OpenFOAM solver ($C_{a_i} = 1, \Delta x = 1/200$), and past works with $\Delta x = 1/256$ [5,48]. The relative error of each numerical prediction against the analytical solution is shown in the parentheses.

Case	$W_{e_{p,1}}$	$W_{e_{p,2}}$	$W_{e_{p,3}}$	Analytical	Present solver	OpenFOAM solver	Xie et al. [5]	Kim and Lowengrub [48]
A1	60	60	60	0.4138	0.4145 (0.17%)	0.3957 (4.37%)	0.4069 (1.67%)	0.4368 (5.56%)
A2	36	60	36	0.4578	0.4626 (1.05%)	0.3958 (13.54%)	0.4485 (2.03%)	0.4622 (0.96%)
A3	108	60	108	0.3746	0.3735 (0.29%)	0.3957 (5.63%)	0.3701 (1.2%)	0.3982 (6.3%)

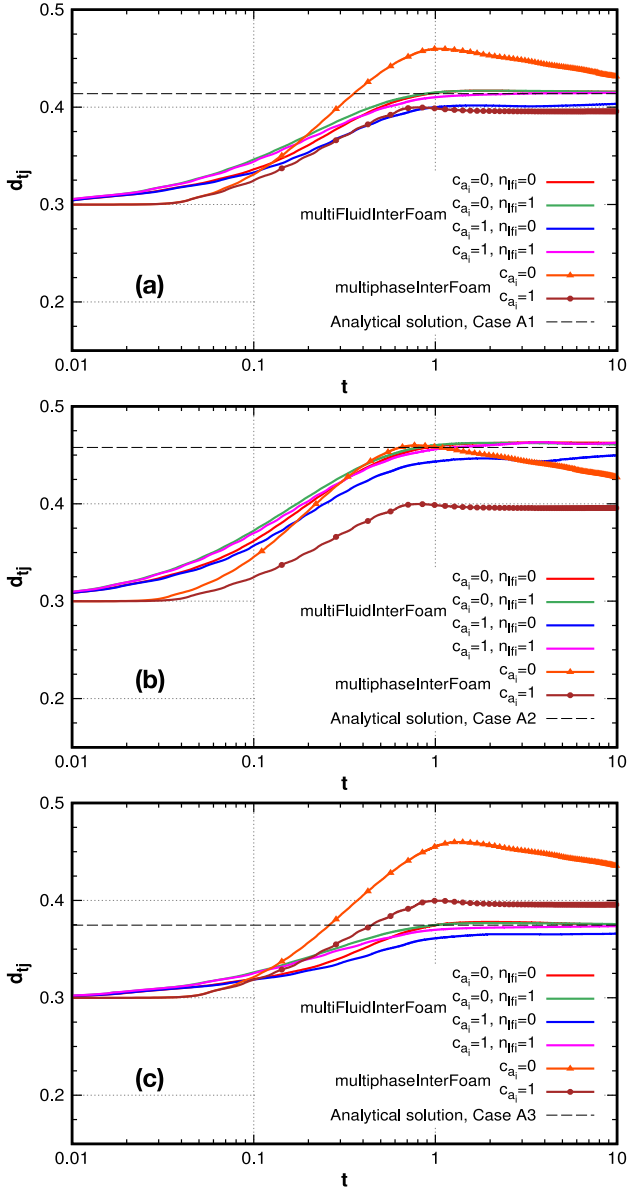


Fig. 4. Time evolution of the interfaces of a floating lens for Case A1 with $W_{e_{p,1}} = W_{e_{p,2}} = W_{e_{p,3}} = 60$ (a), Case A2 with $W_{e_{p,2}} = 60$ and $W_{e_{p,1}} = W_{e_{p,3}} = 36$ (b), and Case A3 with $W_{e_{p,2}} = 60$ and $W_{e_{p,1}} = W_{e_{p,3}} = 108$ (c). The results are obtained for various values of C_{a_i} and $n_{l_{f_i}}$ based on the new and the regular multiphaseInterFoam solvers. The dashed line depicts the equilibrium lens length (see Eq. (37)). The density and viscosity ratios are $\rho_i = 1$ and $\mu_i = 1$, respectively.

mesh tool for two dimensional problems [49]. Therefore, we performed simulations using four static meshes and one simulation which was carried out with an adaptive mesh for the representative Case A1 (i.e. for $W_{e_{p,1}} = W_{e_{p,2}} = W_{e_{p,3}} = 60$). Fig. 6 shows the comparison between the numerical simulations with the different grids for the evolution of the droplet length, d_{ij} , while the analytical solution at

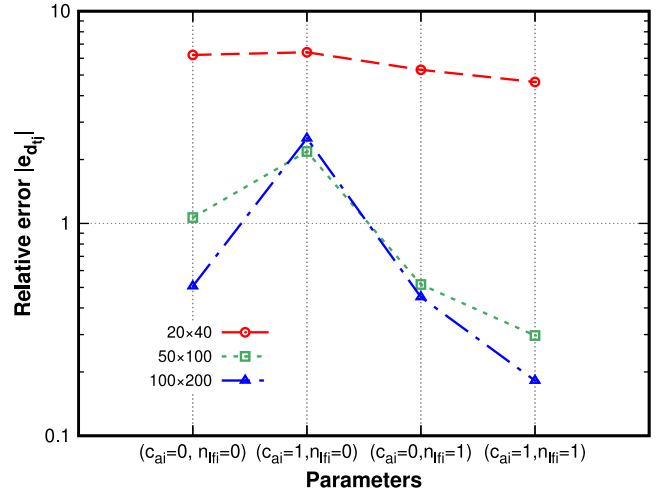


Fig. 5. The relative error of the drop's length at equilibrium $e_{d_{ij}}$ between theory and numerical predictions for various combinations of the parameters C_{a_i} and $n_{l_{f_i}}$ and three computational meshes consisting of 20×40 , 50×100 , and 100×200 control volumes. $W_{e_{p,1}} = W_{e_{p,2}} = W_{e_{p,3}} = 60$, and the density and viscosity ratios are $\rho_i = 1$ and $\mu_i = 1$, respectively.

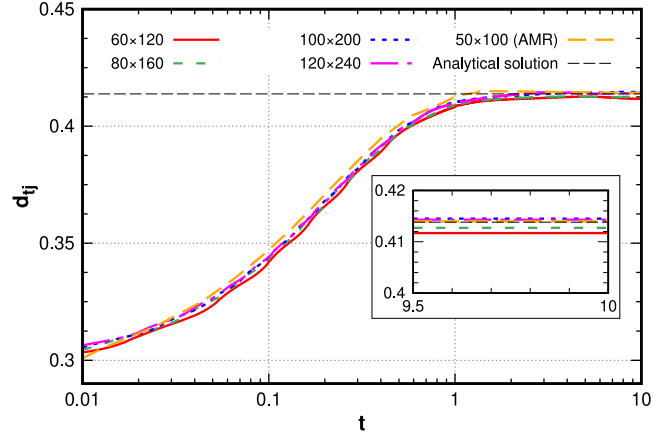


Fig. 6. Time evolution of the interfaces of a floating lens for $W_{e_{p,1}} = W_{e_{p,2}} = W_{e_{p,3}} = 60$ for various fixed uniform computational meshes 60×120 , 80×160 , 100×200 , 120×240 , and an adaptive mesh of 50×100 with a minimum non-dimensional grid spacing of $1/200$. The density and viscosity ratios are $\rho_i = 1$ and $\mu_i = 1$, respectively.

equilibrium is depicted with the dashed line. Sharpening of interfaces is activated with $C_{a_i} = 1$ and smoothing operation in the calculation of curvature is applied with $n_{l_{f_i}} = 1$. The time step in these simulations is set to $\Delta t^* = 10^{-7}$ s. It can be easily concluded that the results shown here are converged. More specifically, the long-term solution of d_{ij} is 0.4117, 0.4127, 0.4145, 0.4143 for the fixed meshes of 60×120 (1), 80×160 (2), 100×200 (3), and 120×240 (4), respectively, and 0.414 for the adaptive mesh of 50×100 with a minimum non-dimensional grid spacing of $1/200$. The numerical errors for the different meshes are 0.5%, 0.26%, 0.18%, 0.12% for the fixed mesh 1, 2, 3, and 4, respectively, and 0.05% for the adaptive mesh. It is noted that, for

the adaptive mesh, the total number of control volumes needed to achieve the same level of accuracy is considerably smaller as the mesh is only refined near the interfaces. The grid is locally refined in regions in which $0.01 \leq \sum_i \alpha_i \leq 0.99$ every 10 time steps (*refineInterval*). Both the maximum refinement level (*maxRefinement*) and the buffer layers (*nBufferLayers*) are set to unity. Converged results and similar findings are observed by varying the values of $We_{p,1}$, $We_{p,3}$, C_{α_i} and n_{lfi} , and they are not shown here.

Full spreading

We then consider the case of complete spreading, where the three interfacial tensions cannot yield a Neumann's triangle. The parameters for the simulated case of complete spreading are $\sigma_{13} = 10$, and $\sigma_{12} = \sigma_{23} = 1$. Clearly, it holds that $S_2 = \sigma_{13} - \sigma_{12} - \sigma_{23} > 0$, which corresponds to a supercritical state. All the other numerical details and parameters are those given in Section 3.1.2.

A complete spreading of phase 2 (green) is anticipated, which is closely reproduced by the new multiphase solver with $C_{\alpha_i} = 1$, $n_{lfi} = 1$ and $C_{\alpha_i} = 0$, $n_{lfi} = 0$ as seen in Fig. 7(a) and (b), respectively. The initially circular droplet evolves finally into a plane fluid layer located within the upper and lower fluids. An obvious blurring occurs near the interfaces when the artificial compression is deactivated, whereas sharper interfaces can be observed for $C_{\alpha_i} = 1$. The size of the computational domain is small and the right wall is close to the droplet. When fluid 2 hits the right wall, it has sufficient momentum to bounce back, forming eventually a wave. It is found that this wave takes a long time period to be suppressed. An even longer time is required when interface compression and smoothing of the volume fractions are not considered.

Fig. 7 also shows the time evolution of the droplet shape and interfaces predicted by using the original *multiPhaseInterFoam* solver with $C_{\alpha_i} = 1$ and 0. In both cases, the green droplet keeps adhering to the interface between the upper and lower fluids and three-phase regions are clearly seen at long times. Since the three interfacial tensions do not satisfy a Neumann's triangle, a stable three-phase region cannot be formed and the results shown in Fig. 7(c) and (d) are clearly not correct. The interface sharpening, as implemented in *multiPhaseInterFoam*, introduces also instabilities in the numerical solutions as indicated by the up and down movements of the drop at times $t = 23$ to 25, which affect the shape of the drop.

3.1.3. Drop levitation

In this Section, we consider a droplet leaving an interface under surface tension forces [2,5,6]. Based on the aforementioned studies, we use the same computational setup as in the floating lens problem in Section 3.1.2, where a circular droplet (fluid 2, green) is initialized between the interface of fluid 1 (blue) and fluid 3 (red) with zero velocity, as shown in Figs. 2 and 8. Here, we consider a droplet levitation problem by setting the interfacial tension between top and bottom fluids σ_{13} and the interfacial tension between droplet and top fluid σ_{23} equal to unity, whereas the interfacial tension between droplet and bottom fluid σ_{12} is set to 10. The spreading parameter is positive $S_3 = \sigma_{12} - \sigma_{13} - \sigma_{23} > 0$, indicating a complete spreading of the top fluid along the droplet in such a way that the top fluid becomes entrained between the droplet and its subphase and the triple points disappear.

Fig. 8(a) shows the predicted results for the evolution of the droplet due to the flow caused by the balance of the surface tension forces. The new multiphase solver with $C_{\alpha_i} = 1$ and $n_{lfi} = 1$ is able to deal with large deformation of the interface, yielding results similar to the those of Smith et al. [2], Xie et al. [5] and Tofighi et al. [6]. At large times, the whole droplet resides completely inside the upper fluid. Consistently with what is expected from theory, the system tends to an equilibrium state in which the interfaces are finally plane or spherical. The total area for the droplet for fluid 2 is kept constant during the simulation pointing out the conservative nature of the new multiphase VOF based solver.

In contrast, the original *multiPhaseInterFoam* solver with $C_{\alpha_i} = 1$ cannot reproduce the levitation of the droplet and it predicts a steady shape of the lens at long simulation times. Similar results are obtained when the surface compression is deactivated. It is recalled that the combination of surface tension coefficients in this Section do not satisfy a Neumann's triangle and, thus, stable three-phase regions cannot be formed. Consequently, the results produced by the original multiphase solver of *OpenFOAM* contradict theoretical anticipations and they are obviously incorrect.

3.2. Consistency and verification tests for the *rheoMultiFluidInterFoam* solver

We now proceed with the verification of the second multiphase solver *rheoMultiFluidInterFoam*. Since there is no rigorous benchmark for non-Newtonian three-phase fluid flows, for the validation of our solver we will examine the two-phase problem of a bubble rising both in a viscoplastic (described by the Herschel–Bulkley model) and a viscoelastic (described by the Phan-Thien Tanner model) fluid. The latter serves as a reference to validate the presently *RheoTool* based implementation of the model and check the consistency of our solver in the limiting case of two-phase flows. Finally, we proceed with the simulation of typical three-phase flows, such as the floating lens and levitating drop problems, which are re-examined by assuming that the bottom fluid layer exhibits a viscoplastic behavior (described by the Herschel–Bulkley model), while the predictions of our generic Non-Newtonian multiphase solver are compared against the previously presented *multiFluidInterFoam* solver, which provides a different implementation of the Herschel–Bulkley model.

3.2.1. Rising bubble in a viscoplastic fluid

The buoyancy-driven rise of an axisymmetric bubble in a viscoplastic material has been studied in detail [33,38,50] and we have chosen this problem to validate the second multiphase solver *rheoMultiFluidInterFoam* in the limiting case of two-phase flow involving a non-Newtonian liquid described by a Generalized Newtonian model, such as the regularized Herschel–Bulkley model.

Initially, a spherical bubble with a non-dimensional radius R_b is placed at a distance of $4.5R_b$ from the bottom rigid wall. The bubble moves upward opposed to the action of gravity in a viscoplastic fluid. Fig. 9 shows the geometry and the initial configuration of the rising bubble problem. Detailed simulations of a bubble rising in Newtonian or viscoplastic materials following the Papanastasiou regularization [37] were undertaken by Tsamopoulos et al. [38], who have used a completely different numerical approach to the present *OpenFOAM* method. In particular, the flow equations are solved numerically using a mixed finite-element/Galerkin method. The grid converged simulations in Tsamopoulos et al. [38]. These are used as a reference for comparison against our numerical predictions. Both the bounded and the Papanastasiou regularization versions of the Herschel–Bulkley model are utilized here. It is recalled that the former implementation is the only one available in the *multiFluidInterFoam* solver, whereas both regularization methods are implemented in the *rheoMultiFluidInterFoam* solver based on the *RheoTool* toolbox.

The non-dimensional parameters characterizing this system are: the Bond number $Bo = \rho_p^* g^* R_b^{*2} / \sigma_{bp}^*$, the Archimedes number $Ar = \rho_p^{*2} g^* R_b^{*3} / \mu_p^{*2}$, and the Bingham number $Bn = \tau_0^* / (\rho_p^* g^* R_b^*)$. Here, R_b^* denotes the radius of the bubble, ρ_b^* , μ_b^* are the density and dynamic viscosity of the bubble, respectively, and σ_{bp}^* is the surface tension coefficient between the two phases. The density of the pseudo-plastic fluid and its dynamic viscosity are ρ_p^* and μ_p^* , respectively, τ_0^* is the yield stress of the material, and g^* is the gravity acceleration. Proper scales for the length, velocity, and stresses are R_b^* , $\rho^* g^* R_b^{*2} / \mu_p^*$, and $\rho^* g^* R_b^*$, respectively. The parameters used in the study are $Ar = 50$, $Bo = 10$, and $Bn = 0$ and 0.14; these dimensionless parameters

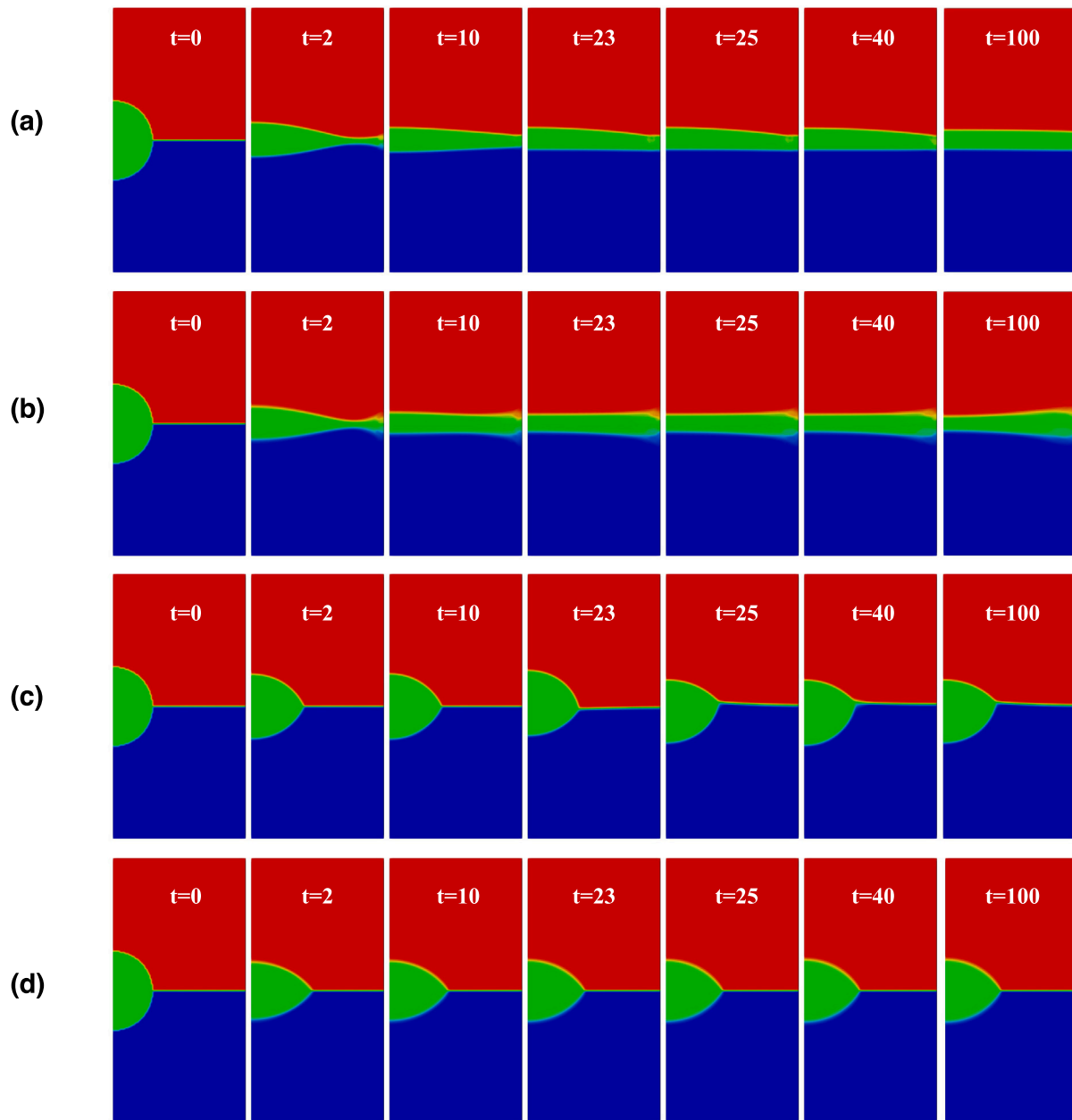


Fig. 7. Complete spreading of an initially spherical droplet under surface tension forces at different time instants. Results are obtained based on the new multiFluidInterFoam solver with $C_{a_i} = 1$, $n_{f_i} = 1$ (a) and $C_{a_i} = 0$, $n_{f_i} = 0$ (b), and by using the original multiPhaseInterFoam solver with $C_{a_i} = 1$ (c) and $C_{a_i} = 0$ (d). The density and viscosity ratios are $\rho_i = 1$, $\mu_i = 1$, respectively, while the surface tension coefficient ratios are $\sigma_{13} = 10$, $\sigma_{12} = \sigma_{23} = 1$.

correspond to a system with $\rho_p^* = 1000 \text{ kg m}^{-3}$, $\rho_b^* = 1 \text{ kg m}^{-3}$, $\mu_p^* = 1.414 \times 10^{-2} \text{ kg m}^{-1} \text{ s}^{-1}$, $\mu_b^* = 10^{-5} \text{ kg m}^{-1} \text{ s}^{-1}$, $\tau_0^* = 1.4 \text{ Pa}$, $R_b^* = 0.001 \text{ m}$, and $g^* = 10 \text{ m s}^{-2}$.

No-slip boundary condition is applied to the walls, while the top surface is assumed to be an open boundary. Zero gradient was enforced at the walls for the bubble volume fraction, while at the open boundary it was set to zero. Axial symmetry is applied for all variables at the left boundary. The size of the computational domain is $7.5R_b \times 20R_b$ in the r , z directions, respectively. The resolution of the computation mesh is 150×400 control volumes. The time step is determined by using a global and local Courant numbers of 0.05, while the maximum time step allowed is set equal to $\Delta t^* = 10^{-5} \text{ s}$. Interface compression is taken into account by setting the parameter C_a to unity, while no filtering is applied. All the results shown in this Section are obtained based on the rheoMultiFluidInterFoam solver. The same case could be easily examined by using the rheoInterFoam solver which

is the RheoTool counterpart of the standard interFoam solver and it can handle two-phase flows. Both two-phase solvers treat the surface tension in exactly the same manner. It was verified that both solvers produce the same results in the case under investigation (not presented herein for conciseness), without considerable differences in the execution times.

Fig. 10(a) shows results regarding the shape of the bubble for a representative Newtonian case ($Bn = 0$, $n = 1$) with $Ar = 50$ and $Bo = 10$ for which a steady-state solution is generally expected. It can be seen that the gas bubble changes from its initially spherical to a steady oblate-spheroid shape. We find excellent agreement when comparing our bubble shape with that predicted by Tsamopoulos et al. [38], shown in left half of Fig. 10(a).

The case of a viscoplastic surrounding fluid is examined in Fig. 10(b) where we depict the bubble shape obtained in steady-state for a characteristic case with $Ar = 10$, $Bo = 50$, along with a color map of the

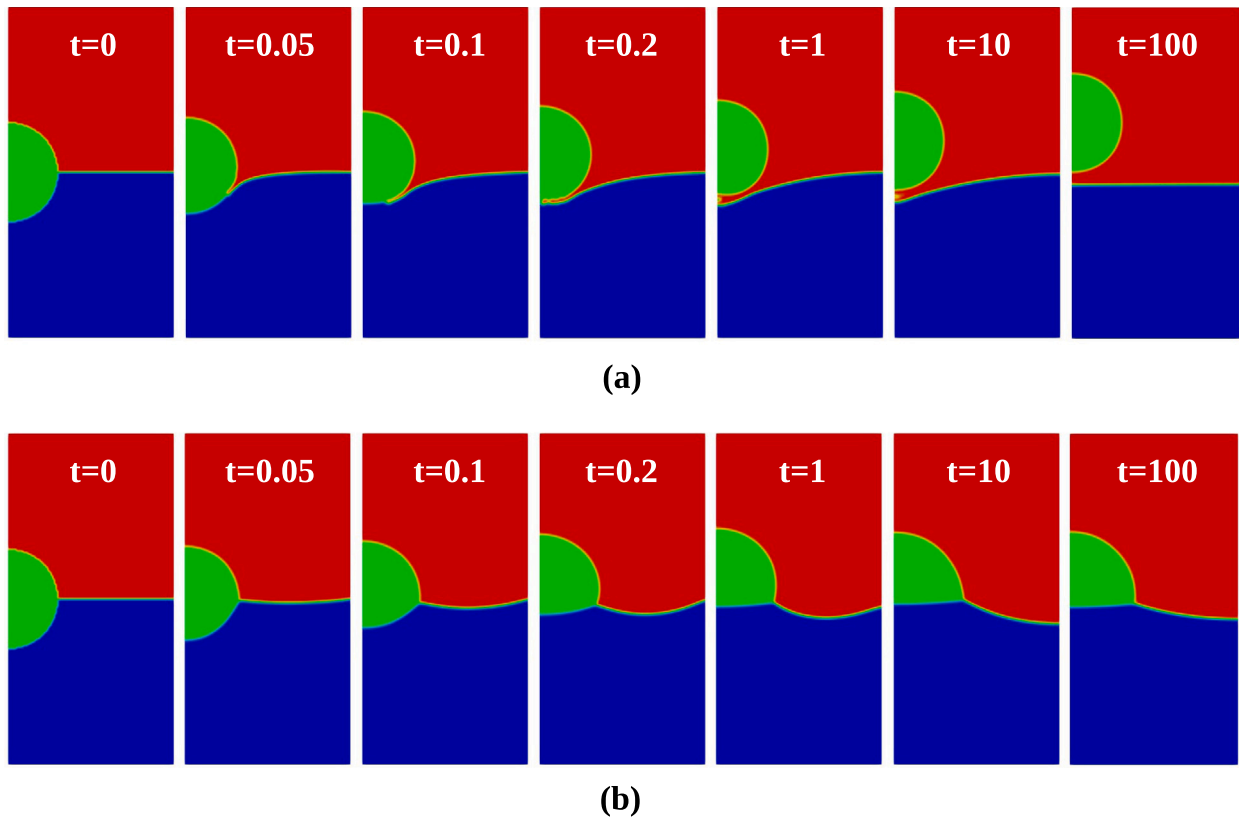


Fig. 8. Evolution of a droplet leaving an interface under the action of surface tension forces a different time instants. Here, the density and viscosity ratios of the phases are $\rho_i = 1$ and $\mu_i = 1$, respectively, and $\sigma_{12} = 10$, $\sigma_{13} = \sigma_{23} = 1$. Results are obtained by the new multiFluidInterFoam solver with $C_{a_i} = 1$, $n_{fji} = 1$ (upper row) and the regular multiphaseInterFoam solver with $C_{a_i} = 1$ (lower row).

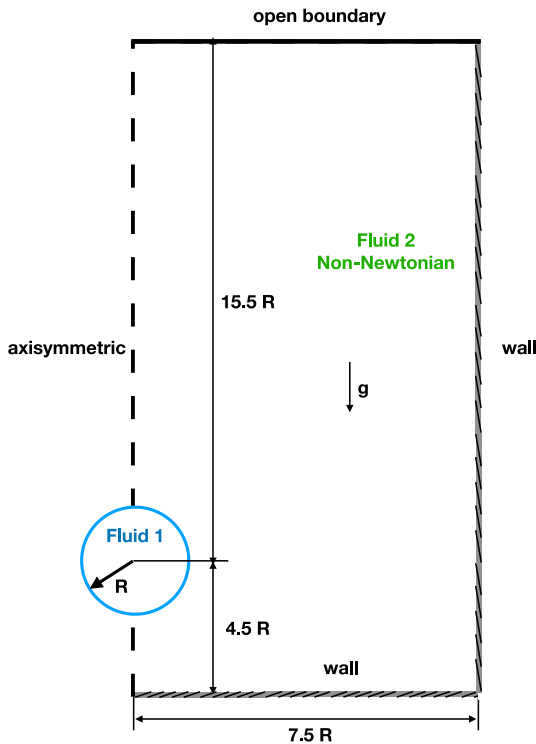


Fig. 9. Geometry and configuration of a bubble rising in a viscoplastic fluid.

dynamic viscosity. Here we have considered the case for $Bn = 0.14$ and $n = 1$ while employing the Papanastasiou regularization model for $N = m^* \rho_p^* g^* R_b^* / \mu_p^* = 10^4$. As seen in Fig. 10(b), the bubble takes a bullet-like shape due to the fact that the effective viscosity around the equatorial plane is higher than that at the poles. To determine the size and shape of the yield/unyielded regions, there are two criteria that have been employed by several researchers in the past. The location of the yield surface can be defined as the locus where either: (a) $\dot{\gamma}^* = 0$, or (b) $|\underline{\tau}^*| = \tau_0^*$. Although these criteria are equivalent in principle according to the discontinuous Bingham model, they are not equivalent when a regularized model such as the Papanastasiou or the bi-viscosity model are used [38]. Only the second criterion is appropriate for regularized models, i.e. that the material flows when the second invariant of the extra stress tensor exceeds the yield stress, and has been used for the results presented herein. This criterion in its dimensionless form becomes $|\underline{\tau}| > Bn$ for yielded material and $|\underline{\tau}| \leq Bn$ for unyielded material. In Fig. 10(b), the yield surface (isoline of $|\underline{\tau}| = Bn$ shown with white line) is also plotted revealing that two regions of unyielded material appear, where the rate of strain is low enough around so that the experienced stresses are below the yield stress of the material: one far from the bubble and another around the equatorial plane, in excellent agreement with the results of Tsamopoulos et al. [38].

In Fig. 10(c) we provide a third consistency test by comparing the predictions between the multiFluidInterFoam and rheoMultiFluidInterFoam when the bounded bi-viscosity model is used which is the only available regularization method in the former solver. To provide as accurate predictions as possible with the bi-viscosity model we have set the upper bounding value of the viscosity, $\mu_{0,p}^*$, to an appropriate value to maintain the same level of accuracy with the Papanastasiou model presented above; the latter is achieved by setting $\mu_{0,p}^* = (1 + Bn N)\mu_p^*$. As seen in Fig. 10(c) the two solvers provide nearly identical predictions.

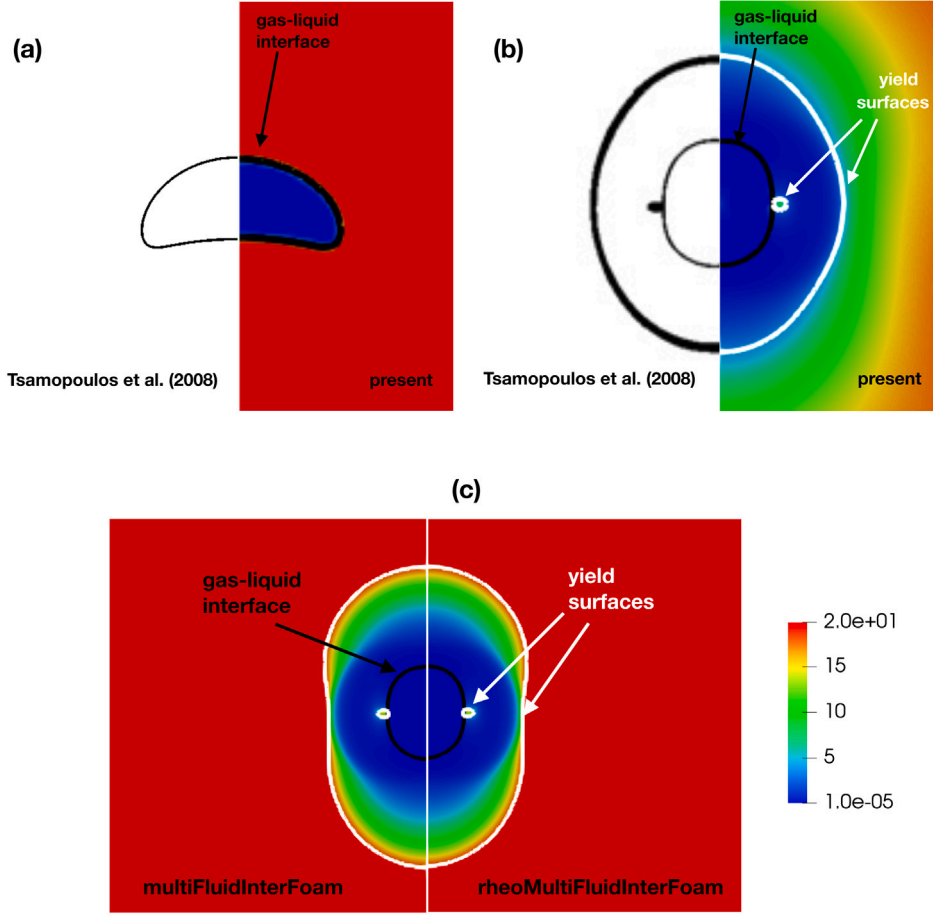


Fig. 10. (a) Bubble shape (black line) in a Newtonian fluid for $Ar = 50$ and $Bo = 10$ and (b) bubble shape (black line) and $|\tau| = Bn$ isoline (white line) in a viscoplastic fluid for $Ar = 50$ and $Bo = 10$, $Bn = 0.14$, and $N = 10^4$, predicted in the present work by the Herschel–Bulkley model with the Papanastasiou regularization (right half) and in Tsamopoulos et al. [38] (left half). (c) Comparison of bubble shape (black line), $|\tau| = Bn$ isoline (white line) and dynamic viscosity obtained by the multiFluidInterFoam (left half) and rheoMultiFluidInterFoam (right half) solvers together with the bounded bi-viscosity Herschel–Bulkley model. The parameters are the same as in (b).

Finally, at this point it also is appropriate to make a comment on the use of the PIMPLE algorithm for the velocity–pressure coupling which is regularly used in the original multiPhaseInterFoam solver. The transient simulations presented in this Section using rheoMultiFluidInterFoam solver employ the SIMPLEC algorithm, as mentioned in Section 2.2.4. It was found that when using the standard PIMPLE algorithm, a sufficiently smaller time step Δt^* was required or a greater number of cycles were necessary in order to obtain meaningful and correct solutions. The same behavior was also found when using the multiPhaseInterFoam solver which employs the PIMPLE method. To assess the effect of the velocity–pressure coupling we summarize in Table 2, five different numerical experiments for $Ar = 50$, $Bo = 10$, $n = 1$, $Bn = 0.14$; the Papanastasiou model has been employed with $N = 10^4$. In case B1, the SIMPLEC algorithm exhibiting by far the best behavior in terms of computational time. On the other hand, when the PIMPLE algorithm is used, e.g. see cases B2 and B3, where a higher number of inner iterations n_{ii} of the coupling method was used while maintaining the same time step, an unsteady flow was predicted due to numerical instabilities; the high viscosity of the viscoplastic material in the unyielded regions generated numerical instabilities near the walls that traveled toward the inner region. A careful inspection, though, of the local viscosity revealed that this behavior was merely a numerical artifact, which significantly deteriorated the accuracy of the solver. By either significantly increasing the number of inner iterations (case B4) or decreasing the time step, Δt^* , (case B5) the correct steady flow regime was predicted, albeit with considerably increased computational cost.

Table 2

Summary of the execution time and the resulting flow regime obtained by the rheoMultiFluidInterFoam solver when using the SIMPLEC and the PIMPLE algorithms for various time steps Δt^* and inner iterations n_{ii} of the velocity–pressure coupling methods. These simulations were performed in parallel using 4 cores for a total simulation time of $T_{sim}^* = 0.001$ s. The parameters are $Ar = 50$, $Bo = 10$, $n = 1$, $Bn = 0.14$; the Papanastasiou model has been employed with $N = 10^4$.

Case	Velocity–pressure coupling	Δt^* [s]	n_{ii}	Flow regime	Execution time [s]
B1	SIMPLEC	10^{-5}	1	steady	231.2
B2	PIMPLE	10^{-5}	3	unsteady	323.8
B3	PIMPLE	10^{-5}	10	unsteady	614.3
B4	PIMPLE	10^{-5}	50	steady	1714.1
B5	PIMPLE	10^{-6}	3	steady	2566.3

3.2.2. Rising droplet in a viscoelastic fluid

In this Section, the rheoMultiFluidInterFoam solver is validated for a Newtonian droplet rising in a viscoelastic fluid described by the Phan-Thien Tanner model. We consider a two-dimensional (2D) rectangular domain having width L_x^* and height L_y^* . Initially, a spherical droplet with a radius $R_{d,0}^*$ is located at $(L_x^*/2, L_y^*/4)$. The no-slip condition is applied at the horizontal boundaries, whereas the free-slip condition is applied at the vertical boundaries. The viscosity μ_2^* and density ρ_2^* of droplet fluid 2 are smaller than those μ_1^* , ρ_1^* of the surrounding fluid 1; $\mu_1^* = \mu_{s,1}^* + \mu_{p,1}^*$, where $\mu_{s,1}^*$ and $\mu_{p,1}^*$ denote the solvent and the polymeric viscosities of the viscoelastic fluid. The gravity vector \mathbf{g}^* points toward the bottom of the domain. Proper reference scales are $L_{ch}^* = 2R_{d,0}^*$, $U_{ch}^* = \sqrt{g^* L_{ch}^*}$, L_{ch}^*/U_{ch}^* for the length,

velocity, and time, respectively. The same case has been examined by Prieto [51]. The relevant non-dimensional parameters are: the Reynolds number $Re = \rho_1^* U_{ch}^* L_{ch}^* / \mu_1^*$, the Weber number $We = \rho_1^* U_{ch}^{*2} L_{ch}^* / \sigma_{12}^*$, the Weissenberg number $Wi = \lambda^* U_{ch}^* / L_{ch}^*$, the viscosity ratio $r_\mu = \mu_2^* / \mu_1^*$, the density ratio $r_\rho = \rho_2^* / \rho_1^*$, and the solvent viscosity ratio $\beta_1 = \mu_{s,1}^* / \mu_1^*$. The quantity σ_{12}^* is the surface tension coefficient of the interface between the two fluids. The linear PTT model with $\epsilon_1 = \zeta_1 = 0$ is used here to facilitate a direct comparison with the results of Prieto [51], who utilized the microscopic Hooke model.

Fig. 11(a, b) show a comparison of the results based on our new multiphase solver ($C_{\alpha_i} = 1$, $n_{lfi} = 1$) with those of Hysing et al. [52] and Prieto [51] for a Newtonian (Case NN: $Wi = 0$, $\beta_1 = 1$) and a viscoelastic surrounding fluid (Case NV: $Wi = 1$, $\beta_1 = 0.5238$), respectively; for both cases the other parameters are $Re = 35$, $We = 10$, $r_\rho = 0.1$, and $r_\mu = 0.1$. In particular, the steady-state drop shape is shown in Fig. 11(a) and the time evolution of the rise velocity is depicted in Fig. 11(b). A uniform computational mesh consisting of 160×320 control volumes was adopted as in the study of Prieto [51], who indicated that convergence of the results was achieved for that mesh. Prieto [51] have used a particle level set method implemented using the Finite Element Method (FEM) and semi-Lagrangian schemes. The results of Hysing et al. [52] correspond to the TP2D method (finite element/level set/straight line approximation) based on a mesh consisting of 320×640 cells for which they reported that the results were grid converged. The same grid was utilized also here. A good agreement of the present predictions with the previously published results obtained by different non-OpenFOAM can be seen for both cases.

To further validate our code, we consider a second test case with $Re = 1.419$, $We = 35.28$, $Wi = 8.083$, $\beta_1 = 0.07$, $r_\rho = 0.1$, and $r_\mu = 0.1$ and compare the results of our new solver against the converged results of Zainali et al. [53] and Vahabi and Sadeghy [54]. Following these works, a spherical droplet is initially placed in a 2D rectangle computational domain of width $L_x^* / L_{ch}^* = 3.333$ and height $L_y^* / L_{ch}^* = 6.666$, while its center is located at $(1/2 L_x^* / L_{ch}^*, 1/4 L_y^* / L_{ch}^*)$. No-slip conditions are enforced at all boundaries. For a droplet with radius $R_{d,0}^* = 0.3$ cm the dimensionless parameters correspond to a system with the following physical properties: $\rho_1^* = 1000 \text{ kg m}^{-3}$, $\rho_2^* = 100 \text{ kg m}^{-3}$, $\mu_{s,1}^* = 0.0717 \text{ Pa s}$, $\mu_{p,1}^* = 0.9533 \text{ Pa s}$, $\mu_2^* = 0.1025 \text{ Pa s}$, $\lambda_1^* = 0.2 \text{ s}$, and $g^* = 9.8 \text{ m s}^{-2}$. The grid resolution is 120×240 control volumes, while interface sharpening and filtering operations are considered with $C_{\alpha_i} = 1$, $n_{lfi} = 1$. The computations of Zainali et al. [53] were performed by using an incompressible smoothed particle hydrodynamics method, while Vahabi et al. [54] utilized a weakly-compressible smoothed particle hydrodynamics approach. The Oldroyd-B constitutive model was used in both works, and therefore for our computations we set $\epsilon_1 = \zeta_1 = 0$.

Fig. 11(c) shows a comparison for the droplet shape at $t^* = 0.13 \text{ s}$. The droplet shape predicted by the `rheoMultiFluidInterFoam` solver exhibits a cusped trailing edge which is a common feature of a Newtonian droplet in viscoelastic medium at high polymer concentrations [56,57]. We note that, for the same dimensional time instant, our prediction is somewhere in-between the sharper shape of Vahabi and Sadeghy [54] and the smoother shape of Zainali et al. [53]. As shown in Fig. 11(c), the cusp at the trailing edge becomes sharper at a later time ($t^* = 0.17 \text{ s}$) based on our multiphase solver. Even at that time instant, an equilibrium shape has not been established (a longer time is required $t^* > 0.3 \text{ s}$ leading to an elongated cusp edge). The exact reasons for this discrepancy are not clear. To check whether this might be an effect of the grid resolution we have performed a mesh refinement study, which showed a negligible effect without influencing the above-mentioned observations. For instance, the maximum (minimum) y^* position of the interface denoting the droplet shape (depicted by the $\alpha_2 = 0.5$ isoline) varied from 1.885 cm (1.034 cm) to 1.89 cm (1.04 cm) as the mesh was refined from 120×240 up to 320×640 control volumes. We have additionally performed simulations using the `rheoInterFoam` solver of the `RheoTool` toolbox [15] to simulate this two-phase

flow case, which produced almost identical results to those of our `rheoMultiFluidInterFoam` solver (see dashed lines in Fig. 11(c)). Therefore, it may be reasonable to assume that these discrepancies could be attributed to the different numerical scheme employed by these authors. The latter conclusion is also supported by the comparison presented in Fig. 11(d) where we provide a comparison with the predictions of von Danwitz [55] for the exact same case. We note that von Danwitz [55] employed a deforming-spatial-domain/stabilized space-time (DSD/SST) procedure that allows computations on moving meshes using the finite element method and reported similar differences with the results of Vahabi and Sadeghy [54] and Zainali et al. [53]. He also provided results about the evolution of the rise velocity of the droplet and the direct comparison between our predictions and those by von Danwitz [55] illustrates a good agreement as shown in Fig. 11(d).

3.2.3. Spreading of a droplet over a viscoplastic fluid

Now that we have verified the consistency of our solver for two-phase flows, we turn our attention to typical three-phase flows. Thus, in this Section we re-examine the floating lens problem, considering that the lower fluid layer is now a Herschel–Bulkley fluid. The interest is mostly focused on verifying whether the `rheoMultiFluidInterFoam` solver coupled with the bi-viscosity Herschel–Bulkley model available in the `RheoTool` toolbox produce the same results with the `multiFluidInterFoam` solver. This consistency check is verified in Fig. 12, which shows negligible differences in the time evolution of d_{ij} and the droplet shape obtained by the two new multiphase solvers. The Reynolds number is $Re = 1$ and the phasic Weber numbers are $We_{p,1} = We_{p,2} = We_{p,3} = 60$, whereas all the numerical details and parameters are exactly the same as those described in . The parameters related to the Herschel–Bulkley model are $Bn = \tau_{0,1}^* R_d^* / (k_1^* U^*) = 0.15$, $\mu_{0,1}^* / k_1^* = 100$, and $n_1 = 1$.

Fig. 12 indicates that a longer time is required in order to reach asymptotically the steady-state solution in the presence of the lower non-Newtonian fluid. Two regions are evident; the first region can be identified at early times, where there is a rapid increase of the droplet's length d_{ij} with time and the second one where a slow increase of d_{ij} takes place. As the droplet is initially away from a stable regime, it is suddenly put in motion. The stress in the region adjacent to the droplet at the non-Newtonian fluid is greater than the yield stress and it eventually starts to flow. In this region, because of the intense shear stresses the effective viscosity of the subphase decreases considerably and becomes much smaller than that of the fluid away from the contact line. Consequently, the material yields only adjacent to the droplet, whereas the rest remains unyielded with a large effective viscosity, $\mu_{0,1}^*$. As a result, we find that the spreading of the droplet resembles that of the pure Newtonian case, as suggested by the similar slopes in the d_{ij} curves at early times. The lower fluid and the droplet are decelerated and the magnitude of the velocities is reduced, delaying the spreading process. As the spreading proceeds and the droplet approaches its equilibrium state, the level of stresses experienced in the subphase decreases considerably and the size of the yielded region adjacent of the droplet decreases with time. Eventually, at very late stages it is expected that when the experienced stress in the bottom fluid becomes smaller than the yield stress and the whole lower fluid may become unyielded, impeding droplet spreading. However, we have to note that the use of a regularized model does not allow to accurately predict the droplet entrapment by unyield material, since the material below yield stress is effectively described by a fluid with very large viscosity. Therefore, the regularized model allows for creeping flow motion of the viscoplastic material and the droplet eventually reaches its equilibrium shape. Since the steady-state solution is independent of the viscosities of the phases, and the predicted geometrical features of the droplet at equilibrium shown in Fig. 13 are found to be the same as in the Newtonian case described in Section 3.1.2.

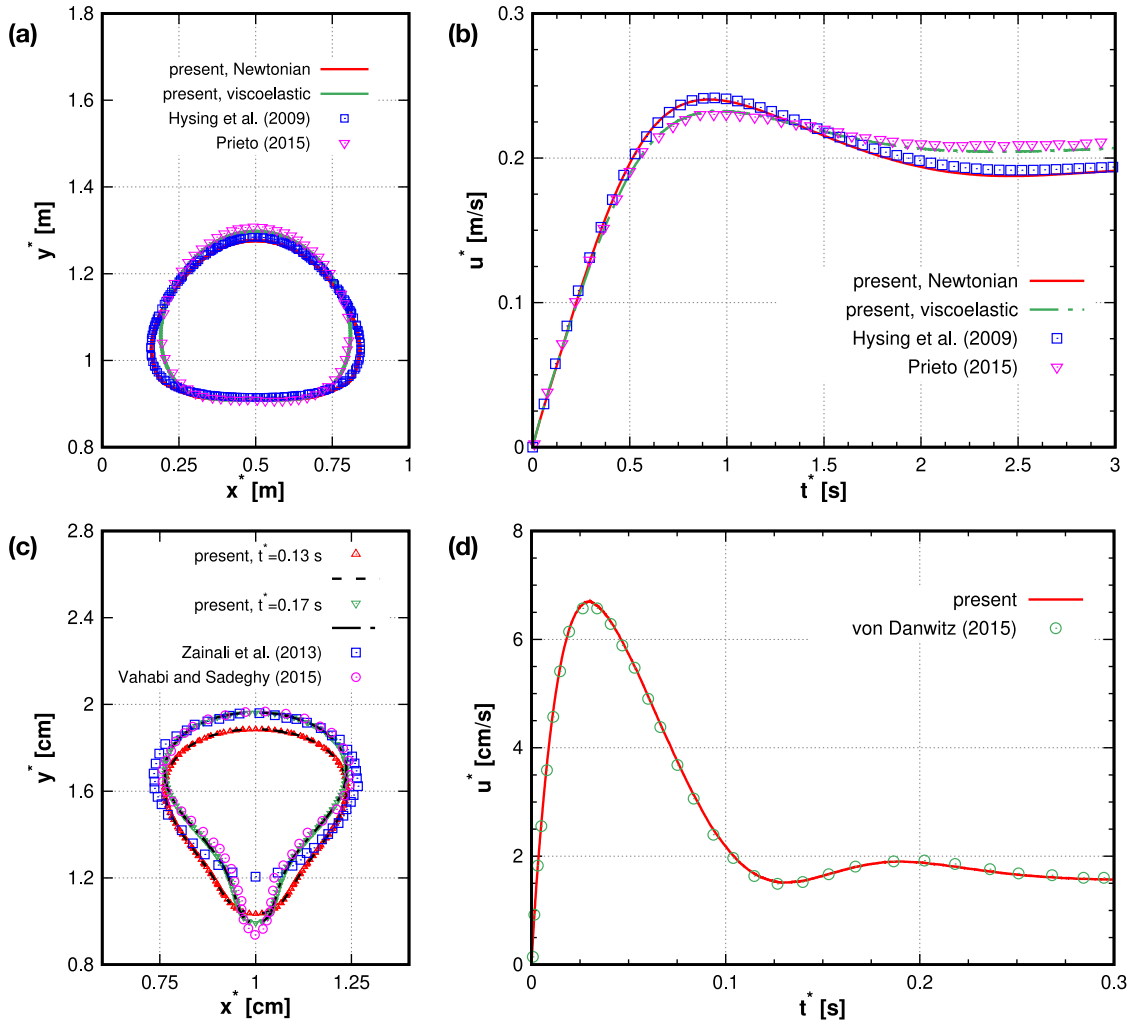


Fig. 11. (a) Steady-state shape and (b) time evolution of the rise velocity for a Newtonian droplet in Newtonian (Case NN) and viscoelastic (Case NV1) mediums. The lines denote results obtained by our solver, while symbols correspond to previous data of Hysing et al. [52] and Prieto. [51]. (c) Droplet shape at time instances $t^* = 0.13$ s and $t^* = 0.17$ s for Case NV2. Symbols correspond to the results obtained by `rheoMultiFluidInterFoam` solver or previous studies of Zainali et al. [53] and Vahabi and Sadeghy [54]; Lines denote results yielded by the `rheoInterFoam` solver. (d) Time evolution of the droplet rise velocity for Case NV2. Lines: present work; Symbols: von Danwitz [55]. Case NN: $Re = 35$, $We = 10$, $r_p = 0.1$, and $r_\mu = 0.1$. Case NV1: $Re = 1.419$, $We = 35$, $Wi = 1$, $\beta = 0.5238$, $r_p = 0.1$, and $r_\mu = 0.1$. Case NV2: $Re = 1.419$, $We = 35.28$, $Wi = 8.083$, $\beta_1 = 0.07$, $r_p = 0.1$, and $r_\mu = 0.1$.

3.2.4. Levitation of a drop within Newtonian/viscoplastic fluids

In this Section, the simulations for the drop levitation are repeated with the lower fluid exhibiting a viscoplastic behavior. Again, our interest is mostly focused on verifying whether the two new multiphase solvers coupled with a bi-viscosity Herschel–Bulkley model produce the same results. This consistency check is verified in Fig. 14, which shows negligible differences in the time evolution of the droplet shape and interfaces obtained by the new solvers. This is expected since the only difference between the two solvers is that the constitutive models of `RheoTool` toolbox are used the `rheoMultiFluidInterFoam` solver, and comparisons are performed for the same viscosity model. All the numerical details and other parameters are exactly the same as those described above for the Newtonian case. The parameters related to the Herschel–Bulkley model are $Bn = \tau_{0,1}^* R_d^* / (k_1^* U^*) = 0.15$, $\mu_{0,1}^* / k_1^* = 100$, and $n_1 = 1$.

4. Discussion

The most significant difference between `multiFluidInterFoam`, `rheoMultiFluidInterFoam` solvers and the original `multiPhaseInterFoam` solver is the treatment of the continuum surface tension force. It can be shown that both Eq. (12) of the original solver

and Eq. (20) of the new solvers produce the correct expression for the surface tension force in case of two-phase flows. However, even though the two CSF models are identical theoretically, their numerical behavior differs significantly. The CSF model used in the original solver involves the calculation of more gradients of the phase fraction, which may introduce additional numerical errors, especially for highly discontinuous α_i fields. These errors, though, can be mitigated by smoothing the distribution of α_i in the calculation of curvature.

In the dambreak problem, the flow is mainly controlled by differences in the density of the phases. The interfacial tension effects are secondary and they do not contribute considerably to the dynamics of the multiphase flow. Any three-phase regions, such as contact points, disappear during the early stages of the flow evolution and mostly interfaces between two phases are observed. Therefore, for this problem, the particular treatment of the continuum surface tension force does not play a significant role; similar results are obtained between the standard `multiPhaseInterFoam` and both our newly developed solvers with only some small differences at the very late stages of the simulation, which can clearly be attributed to the different numerical implementation of the CSF model.

In the case of interfacial tension dominated multiphase flows, though, it is demonstrated that the original `multiPhaseInterFoam`

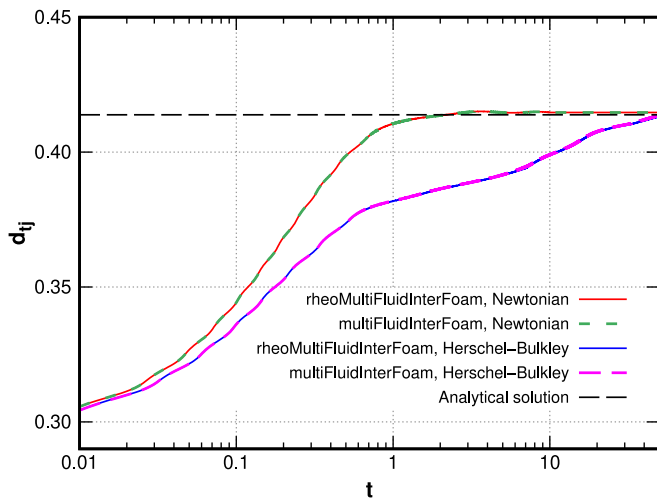


Fig. 12. Time evolution of the interfaces of a floating lens with a non-Newtonian lower fluid for $Re = 1$ and $We_{p,1} = We_{p,2} = We_{p,3} = 60$. The results are obtained by using the `multiFluidInterFoam` and `rheoMultiFluidInterFoam` solvers together with the bounded bi-viscosity Herschel–Bulkley model, $Bn = \tau_{0,1}^* R_d^* / (k_1^* U^*) = 0.15$, $\mu_{0,1}^* / k_1^* = 100$, and $n_1 = 1$.

solver is not able to produce the correct behavior, providing inaccurate predictions. The obtained results contradict both theory and previously published results for both problems of a floating lens and a levitating drop. On the other hand, we show that the newly developed `multiFluidInterFoam` solver and its non-Newtonian counterpart (`multiFluidInterFoam`) improve significantly the quality of the predictions, exhibiting robust behavior and providing very accurate predictions, as indicated by the good agreement against analytical and previous numerical results.

Finally, we have shown that the use of the PIMPLE velocity–pressure algorithm turned out to be insufficient in cases involving a non-Newtonian fluid. For example, in the floating lens and the rising bubble problems with a Herschel–Bulkley fluid, although the numerical solutions do not necessarily diverge, the results were found to be quite sensitive to the choice of the time step. In some case, the interactions between a very viscous fluid and the wall induced numerical instabilities that diffused in the inner flow region. In this way, we often found that the developed shear stress was greater than the yield stress and the whole lower fluid layer was put in motion eliminating the presence any regions with unyielded material. Moreover, we noticed several regions of local maximum values of viscosity corresponding to the unyielded region appear randomly in the non-Newtonian fluid, with the droplet or the bubble mostly located adjacent to a fluid having a lower viscosity. This resulted in nonphysical unsteady solutions for the rising bubble problem. In addition, the transient dynamics of the droplet spreading were influenced, indicating an unexpected and nonphysical fast evolution of the spreading process. In order to obtain a reasonable and accurate solution, we found that the time step must be reduced or the inner iterations of the PIMPLE algorithm must be increased. However, both choices increase considerably the overall computational cost. The adoption of the SIMPLEC algorithm allows to use higher time steps without sacrificing the accuracy of the solutions, obtaining also reliable results. This is particularly important, since complex fluid flows are typically more computationally demanding and e.g. in the case of droplet spreading, the non-Newtonian case generally requires greater times to reach a steady-state regime.

5. Conclusions

In this article, a robust and accurate finite-volume solver `multiFluidInterFoam` for the simulation of three-phase flows

of Newtonian/non-Newtonian fluids with significant surface tension effects is developed in the open source `OpenFOAM`. The volume of fluid method (VOF) is utilized, which employs a volumetric phase fraction for each fluid phase. Sharpening of the interface is achieved by introducing an artificial compression term in the equation for each volume fraction. Starting from the original `multiphaseInterFoam` solver, we implemented the interfacial tension coefficient decomposition method to deal with tension pairings between different phases using a compositional approach. The resulting interfacial force is reformulated properly as the sum of each phase body force. VOF smoothing is also performed to estimate accurately the curvature of each interface. The SIMPLEC algorithm was adopted to utilize the velocity–pressure coupling.

Finally, a second solver `rheoMultiFluidInterFoam` was developed by implementing the `RheoTool` toolbox in order to make use of several constitutive models for complex fluids. This allows the investigation of more physically and rheologically complex situations in which several non-Newtonian properties such as viscoelasticity, shear-thinning, and yield stress are taken into account.

The numerical framework has been validated against the three-phase dam break problem and several benchmark problems for a floating lens and a levitating droplet for Newtonian fluids. The performance of the `rheoMultiFluidInterFoam` solver was assessed in the case of two-phase rising bubble in a viscoplastic or a viscoelastic fluid and several consistency tests were performed for the floating lens and levitating drop, in which the lower fluid exhibited a viscoplastic behavior. The newly developed multiphase solvers can represent better the underlying physics of surface tension dominated flow problems and decrease the overall computational cost without sacrificing accuracy.

CRediT authorship contribution statement

Chris Dritselis: Writing – original draft, Software, Investigation, Validation, Visualization. **George Karapetsas:** Conceptualization, Methodology, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project has received funding from the Hellenic Foundation for Research and Innovation, Greece (HFRI) and the General Secretariat for Research and Technology (GSRT), Greece, under grant agreement No 792. Results presented in this work have been produced using the Aristotle University of Thessaloniki (AUTH) High Performance Computing Infrastructure and Resources.

Appendix A. Derivation of VOF sharpening for multiphase flows

A mixed velocity \mathbf{u}^* can be defined as a weighted average of the fluid velocities \mathbf{u}_k^* as follows

$$\mathbf{u}^* = \sum_k \alpha_k \mathbf{u}_k^*. \quad (39)$$

The relative velocity between fluids i and k can be defined as

$$\mathbf{u}_{r,ik}^* = \mathbf{u}_i^* - \mathbf{u}_k^*. \quad (40)$$

The mixed velocity in Eq. (39) can be re-written as

$$\mathbf{u}^* = \sum_{k \neq i} \alpha_k \mathbf{u}_k^* + \alpha_i \mathbf{u}_i^* \Rightarrow \alpha_i \mathbf{u}_i^* = \mathbf{u}^* - \sum_{k \neq i} \alpha_k \mathbf{u}_k^*. \quad (41)$$

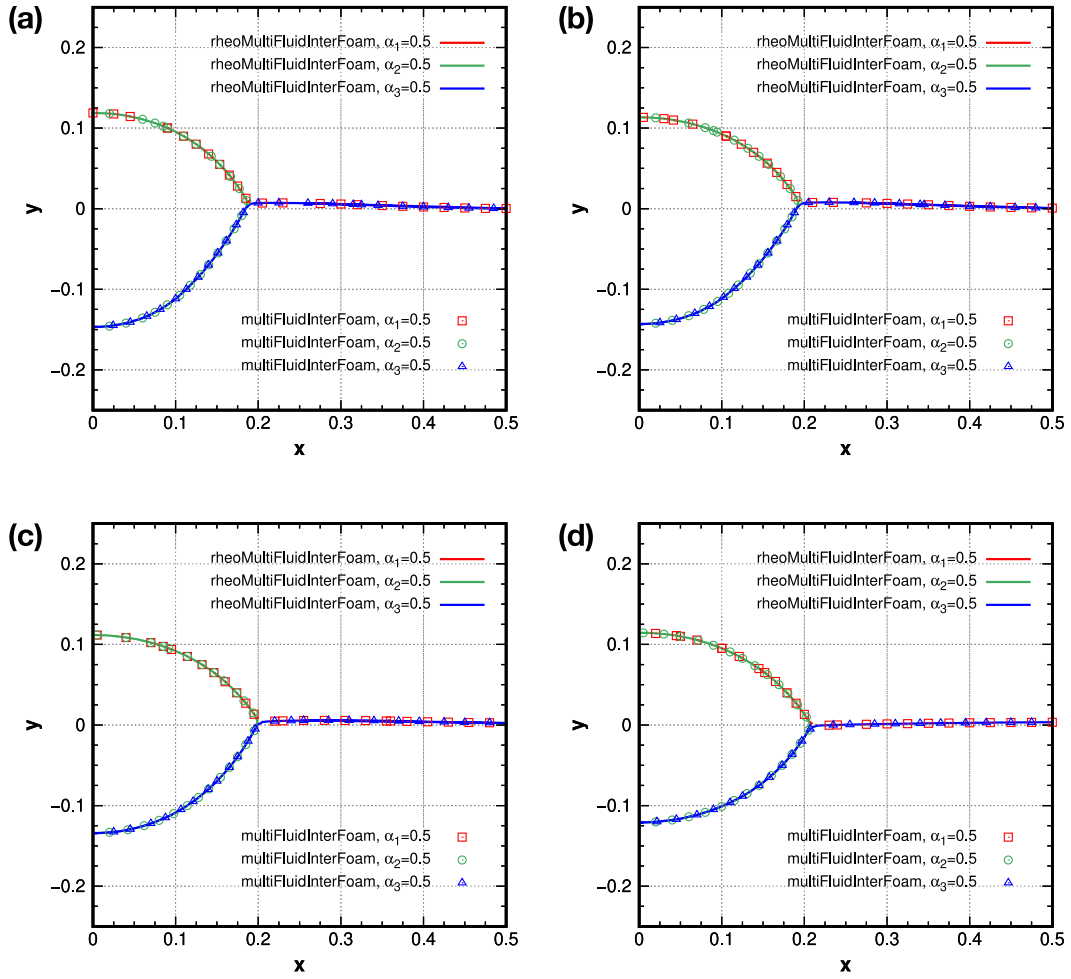


Fig. 13. Shape of a floating lens with a non-Newtonian lower fluid for $Re = 1$ and $We_{p,1} = We_{p,2} = We_{p,3} = 60$ at non-dimensional times $t = 1$ (a), $t = 10$ (b), $t = 25$ (c), and $t = 50$ (d). The results are obtained by using the multiFluidInterFoam and rheoMultiFluidInterFoam solvers together with the bounded bi-viscosity Herschel–Bulkley model, $Bn = \tau_{0,1}^* R_d^* / (k_1^* U^*) = 0.15$, $\mu_{0,1}^* / k_1^* = 100$, and $n_1 = 1$.

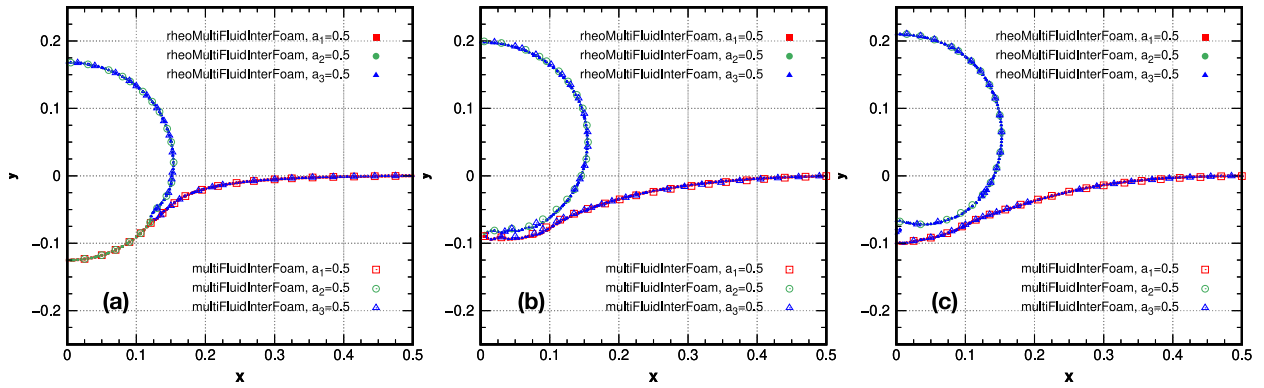


Fig. 14. Comparison of the $\alpha_1 = \alpha_2 = \alpha_3 = 0.5$ level contour for all phases at non-dimensional times $t = 0.01$ (a), $t = 0.1$, (b) and $t = 1$ (c) in the case of levitating drop with a non-Newtonian lower fluid. Results are obtained based on the multiFluidInterFoam and rheoMultiFluidInterFoam solvers with the bounded bi-viscosity Herschel–Bulkley model, $Bn = \tau_{0,1}^* R_d^* / (k_1^* U^*) = 0.15$, $\mu_{0,1}^* / k_1^* = 100$, and $n_1 = 1$. The relevant parameters are the same to those for the case of Fig. 8.

Multiplying Eq. (41) by α_i , it yields

$$\alpha_i \alpha_i \mathbf{u}_i^* = \alpha_i \mathbf{u}^* - \alpha_i \sum_{k \neq i} \alpha_k \mathbf{u}_k^*. \quad (42)$$

The term $\alpha_k \mathbf{u}_k^*$ in the above equation can be rearranged further, by solving Eq. (40) to \mathbf{u}_k and multiplying the result by α_k , and Eq. (42)

can be written as

$$\alpha_i \alpha_i \mathbf{u}_i^* = \alpha_i \mathbf{u}^* - \alpha_i \sum_{k \neq i} (\alpha_k \mathbf{u}_i^* - \alpha_k \mathbf{u}_{r,ik}). \quad (43)$$

After some maths, the following is obtained

$$\alpha_i \mathbf{u}_i^* = \alpha_i \mathbf{u}^* + \alpha_i \sum_{k \neq i} \alpha_k \mathbf{u}_{r,ik}. \quad (44)$$

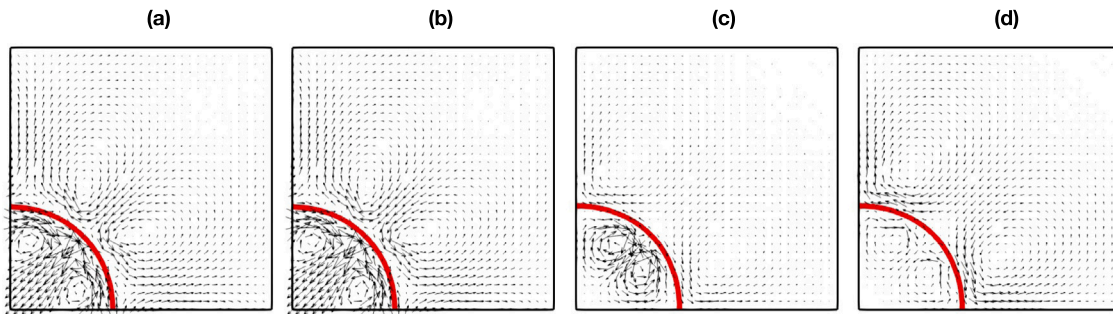


Fig. 15. Visualization of spurious current for the static drop case for $La = 120$ at time $t = 30$ obtained by the `interFoam` (a) and the `rheoMultiFluidInterFoam` with $n_{ifi} = 0$ (b), $n_{ifi} = 1$ (c), and $n_{ifi} = 2$ (d). The sharpening parameter is $C_a = 1$. The red line denotes the isoline $\alpha = 0.5$.

Finally, Eq. (13) that is numerically solved in the multiphase solvers can be derived from the transport equation (Eq. (4)) for the i th volume fraction α_i by replacing the term $\alpha_i \mathbf{u}_i^*$ with the above relation (44).

Appendix B. Spurious currents quantification

In order to quantify the effect of spurious currents of the newly developed multiphase solvers the elementary two-phase test case of a drop in hydrostatic equilibrium under zero gravity is firstly examined here (see Popinet [58]). A cylindrical interface of fluid 2 is initialized inside a fluid 1. Both fluids have the same viscosity μ^* and density ρ^* . The surface tension between the two fluids is denoted as σ^* . Initially, the drop has a diameter $D_0^* = 2R_0^* = 0.8\text{m}$. Only a quarter of the 2D geometry is simulated in a computational domain of size $1\text{m} \times 1\text{m}$. The drop is located in the bottom left corner of the domain. Symmetry conditions are applied to all boundaries. A uniform computational mesh is used consisting of 32×32 control volumes. The problem is characterized by the Laplace number $La = \rho^* D_0^* \sigma^* / \mu^{*2}$, which is set to $La = 120$ here.

At first, it was verified that the newly developed multiphase solvers are well-balanced, in accordance to the standard two-phase solver `interFoam`. For this reason, the exact curvature was imposed in the solvers instead of the computed one. It was found that the maximum velocities in the domain obtained very small values of the order of the numerical precision. Moreover, the pressure jump across the 2D drop was equal to the theoretical one $\Delta P_{th}^* = 2.5\text{Pa}$ for the presently theoretical curvature of $k_{th}^* = 2.5\text{m}^{-1}$.

Fig. 15 shows a qualitative view of the spurious currents generated by the two-phase solver `interFoam` and our multiphase solver `rheoMultiFluidInterFoam`. Similar results are also obtained by using the `multiFluidInterFoam` solver.

For the quantification of the spurious velocities, the capillary number $Ca_{max} = \mu^* U_{max}^* / \sigma^*$ was calculated, where U_{max}^* is the maximum velocity magnitude over the whole computational domain. Fig. 16 shows the time evolution of Ca_{max} . As discussed in the paper, the new multiphase solvers and the standard two-phase flow solver `interFoam` produce approximately the same results in two-phase flows. This is also valid here, as shown in Fig. 16, where Ca_{max} is exactly the same for the two solvers. The noticeable point is that our multiphase solver improves its performance with respect to the spurious velocities due to the filtering procedure. The spurious currents are reduced about one order of magnitude and they are comparable to the results when using the geometrical VOF `isoAdvector` method combined with the reconstructed distance function (RDF) method `plicRDF-5` (Gamet et al. [59]). All the results do not converge to the machine precision as in case of the Basilisk code [58] and the spurious velocity remain constant.

Next, we focus on the quantification of the spurious currents in a representative three-phase test case of a lens placed between two immiscible and stationary fluids. It is pointed out that for this test

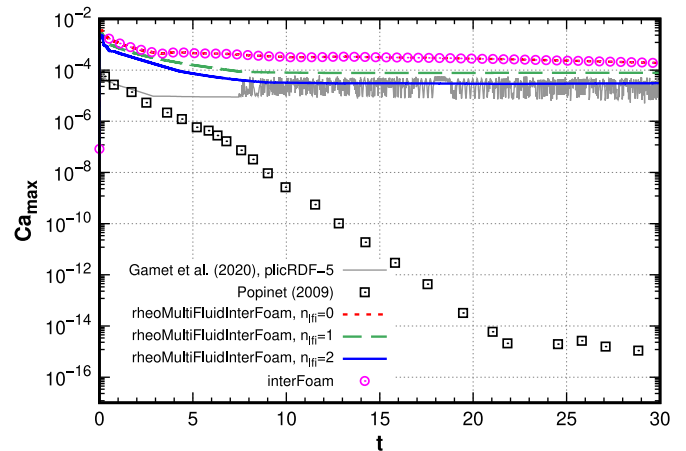


Fig. 16. Maximum capillary number over the computational domain for the static drop case for $La = 120$.

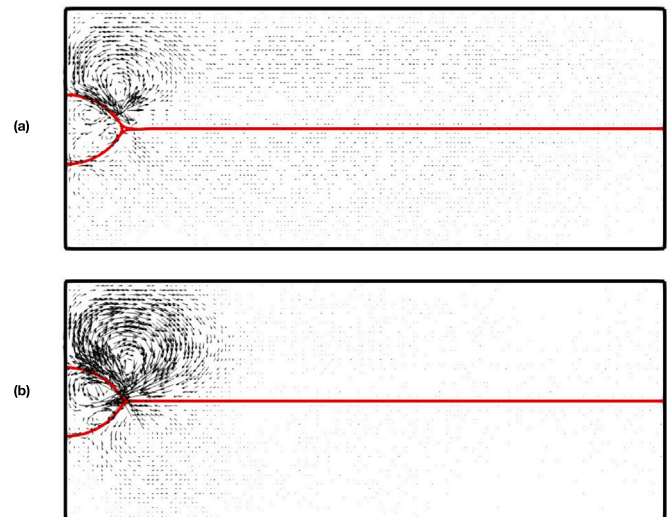


Fig. 17. Visualization of spurious current for the three-phase case of a static drop between two fluids at time $t = 1200$ obtained by the `rheoMultiFluidInterFoam` (a) and the `multiphaseInterFoam` (b). The red lines denote the isolines $\alpha_i = 0.5$ for $i = 1, 2, 3$.

case there are no other reference numerical solutions for comparison. Therefore, a comparison is presented here between the `rheoMultiFluidInterFoam` and the `multiphaseInterFoam` solvers, and the analytical solutions as well.

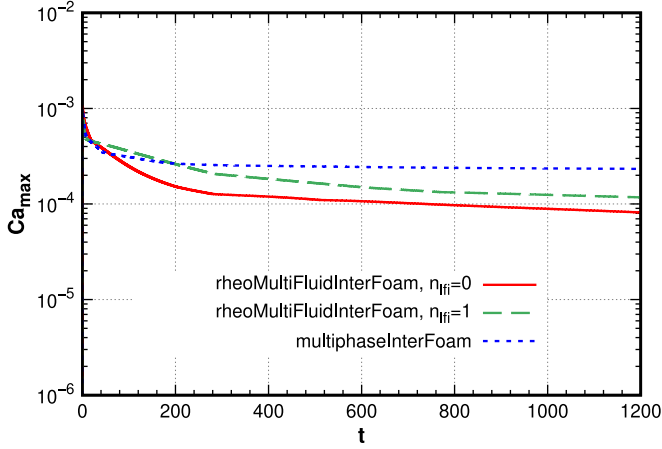


Fig. 18. Maximum capillary number over the computational domain for the three-phase case of a static drop between two fluids.

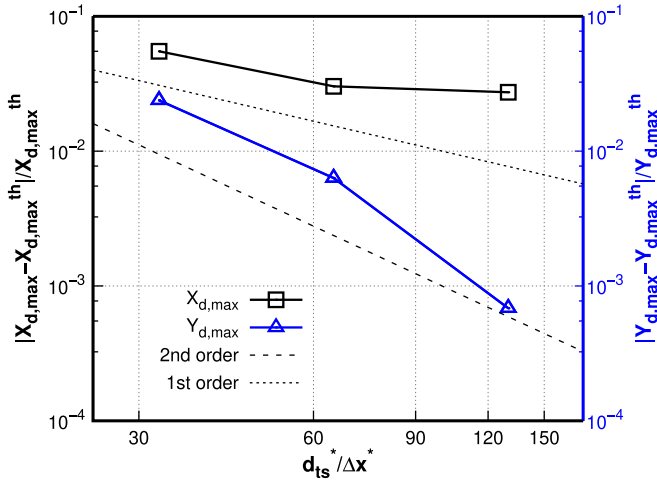


Fig. 19. Relative error for the maximum size of the lens $X_{d,max}^{*th}$ and $Y_{d,max}^{*th}$ in the x and y directions, respectively, for the three-phase case of a static droplet between two fluids.

A lens (fluid 2) is initially imposed in the interface between fluid 1 (bottom) and fluid 3 (top) based on the analytical solutions given by Leclaire et al. [41]. Similarly to the problem examined in Section 3.1.2, only the half domain is simulated by setting symmetry and zero gradient conditions at the left and right boundaries. No-slip conditions are enforced at the upper and lower boundaries. The lens is placed at $(X, Y) = (0, 0)$. The size of the computational domain is $[0 \text{ m}, 4 \text{ m}] \times [-0.8 \text{ m}, 0.8 \text{ m}]$. Three meshes are used 156×52 , 312×104 , and 624×208 . The parameters of the problem are $\rho_1^* = 100 \text{ kg m}^{-3}$, $\rho_2^* = 10 \text{ kg m}^{-3}$, and $\rho_3^* = 1 \text{ kg m}^{-3}$, $\mu_1^* = 5 \text{ Pa s}$, $\mu_2^* = 0.5 \text{ Pa s}$, and $\mu_3^* = 0.05 \text{ Pa s}$. The surface tensions coefficients are $\sigma_{ij}^* = 0.05 \text{ N m}^{-1}$ for $ij = 12, 13$, and 23 . The area of the lens is $A_c^* = 0.26205 \text{ m}^2$, the length of the lens is $d_{lj}^* = 0.8 \text{ m}$, and the maximum position of the lens in the x , y directions is $X_{d,max}^* = 0.4 \text{ m}$ and $Y_{d,max}^* = 0.2309 \text{ m}$, respectively.

The different implementation of the surface tension force results in different predictions of the spurious velocities as seen in Fig. 17. It is obvious that the new multiphase solver behaves better than the original solver for this stationary problem, in total agreement with the findings for the moving interface problem examined in Section 3.1.2.

The maximum capillary number Ca_{max} is calculated and it is also used here to quantify the spurious velocities. Fig. 18. indicates that similar results to those of the two-phase test case are obtained. Once

again, the performance of the new solver regarding the spurious currents is better as compared with the `multiphaseInterFoam` solver. However, the filtering procedure seems to improve the results in the stationary case, in contrast to what is found in the moving interface problem.

The numerical predictions of the `rheoMultiFluidInterFoam` solver are compared against the analytical solutions for three computational meshes. For this reason, the errors for the maximum size of the lens $X_{d,max}^*$ and $Y_{d,max}^*$ in the x and y directions, respectively, were calculated and plotted in Fig. 19. The relative errors are defined as $|\phi - \phi^{th}|/|\phi^{th}|$, where ϕ can be either $X_{d,max}^*$ or $Y_{d,max}^*$, and th indicates values based on the analytical solution. It can be seen that both $X_{d,max}^*$ and $Y_{d,max}^*$ are reduced with decreasing Δx^* . The range of the relative error varies between 2.73–5.5 percent for $X_{d,max}^*$ and 0.68–2.38 percent for $Y_{d,max}^*$.

Therefore, the performance of the newly developed multiphase solver with respect to the spurious currents is better than that of the standard `multiphaseInterFoam` solver. Even though the spurious velocities do not decrease down to the machine accuracy, a good agreement with the analytical solutions is found. In accordance with the moving interface problem in Section 3.1.2, the numerical results here are also converged with grid refinement and errors below 5.5 percent are generally obtained.

References

- [1] Zhao HK, Chan T, Merriman B, Osher S. A variational level set approach to multiphase motion. *J comput phys* 1996;127:179–95.
- [2] Smith K, Solis F, Chopp D. A projection method for motion of triple junctions by levels sets. *Interfaces Free Bound* 2002;4:263–76.
- [3] Muradoglu M, Tasoglu S. A front-tracking method for computational modeling of impact and spreading of viscous droplets on solid walls. *Comput Fluids* 2010;39:615–25.
- [4] Bonhomme R, Magnaudet J, Duval F, Piar B. Inertial dynamics of air bubbles crossing a horizontal fluid-fluid interface. *J Fluid Mech* 2012;707:405–43.
- [5] Xie Z, Pavlidis D, Salinas P, Pain CC, Matar OK. A control volume finite element method for three-dimensional three-phase flows. *Int J numer Meth Fluids* 2020;1–20.
- [6] Tofighi N, Yildiz M. Numerical simulation of single droplet dynamics in three-phase flows using ISPH. *Comput Math Appl* 2013;66:525–36.
- [7] Garcke H, Nestler B, Stoth B. A multiphase field concept: Numerical simulations of moving phase boundaries and multiple junctions. *SIAM J Appl Math* 1999;60:295–315.
- [8] Kim J, Kang K, Lowengrub J. Conservative multigrid methods for Cahn-Hilliard fluids. *Commun Math Sci* 2004;2:53–77.
- [9] Kim J. Phase field computations for ternary fluid flows. *Comput Methods Appl Mech Eng* 2007;196:4779–88.
- [10] Weller HG, Tabor G, Jasak H, Fureby C. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Comput Phys* 1998;12:620–31.
- [11] The OpenFOAM Foundation. *OpenFOAM v7 user guide*. 2020.
- [12] Andersson P. *Tutorial multiphaseInterFoam*. 2020, http://www.tfd.chalmers.se/hani/kurser/OS_CFD_2010/patrikAndersson/patrikAnderssonReport.pdf.
- [13] Khanal A, Sharma R, Corcoran TE, Garoff S, Przybycien TM, Tilton RD. Surfactant driven post-deposition spreading of aerosols on complex aqueous subphases. 1: High deposition flux representative of aerosol delivery to large airways. *J Aerosol Med Pulmon Drug Deliv* 2015;28:382–93.
- [14] Sharma R, Khanal A, Corcoran TE, Garoff S, Przybycien TM, Tilton RD. Surfactant driven post-deposition spreading of aerosols on complex aqueous subphases. 2: Low deposition flux representative of aerosol delivery to small airways. *J Aerosol Med and Pulmon Drug Deliv* 2015;28:394–405.
- [15] Pimenta F, Alves MA. *RheoTool*. 2016, <https://github.com/fppimenta/rheoTool>.
- [16] Hirt CW, Nichols BD. Volume-of-fluid method for the dynamics of free boundaries. *J Comput Phys* 1981;39:201–25.
- [17] Brackbill JU, Kothe DB, Zemach C. A continuum method for modeling surface tension. *J Comput Phys* 1992;100:335–54.
- [18] Personnettaz P, Beckstein P, Landgraf S, Köllner T, Nimitz M, Weber N, et al. Thermally driven convection in Li||Bi liquid metal batteries. *J. Power Sources* 2018;401:362–74.
- [19] Bublik S, Einarsrud KE. Inverse modelling of interfacial tension between ferroalloy and slag using openfoam. In: 14th international conference on CFD in Oil & Gas, metallurgical and process industries SINTEF. 2020, 28–38.
- [20] Weller HG. A new approach to VOF-based interface capturing methods for incompressible and compressible flow. Technical report, OpenCFD Ltd; 2008.

- [21] Berberovic E, van Hinsberg NP, Jakirlić S, Roisman IV, Tropea C. Drop impact onto a liquid layer of finite thickness: dynamics of the cavity evolution. *Phys Rev E* 2009;79:036306.
- [22] Deshpande SS, Anumolu L, Trujillo MF. Evaluating the performance of the two-phase flow solver interFoam. *Comput Sci Disc* 2012;5:014016.
- [23] Issa RI. Solution of implicitly discretized fluid flow equations by operator-splitting. *J. Comput Phys* 1986;62:45–65.
- [24] Patankar SV, Spalding DB. A calculation procedure for heat mass and momentum transfer in three dimensional parabolic flows. *Int J Heat Mass Transf* 1972;15:1787–806.
- [25] Lafaurie B, Nardone C, Scardovelli R, Zaleski S, Zanetti G. Modelling merging and fragmentation in multiphase flows with SURFER. *J Comput Phys* 1994;113:134–47.
- [26] Scardovelli R, Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annu Rev Fluid Mech* 1999;31:567–603.
- [27] Harvie DJE, Davidson MR, Rudman M. An analysis of parasitic current generation in volume of fluid simulations. *Appl Math Model* 2006;30:1056–66.
- [28] Gerlach D, Tomar G, Biswas G, Durst F. Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows. *Int J Heat Mass Transf* 2006;49:740–54.
- [29] van Doormaal JP, Raithby GD. Enhancements of the SIMPLE method for predicting incompressible fluid flows. *Numer Heat Transf* 1984;7:147–63.
- [30] Pimenta F, Alves MA. Stabilization of an open-source finite-volume solver for viscoelastic fluid flows. *J Non-Newtonian Fluid Mech* 2017;239:85–104.
- [31] Balmforth NJ, Frigaard IA, Ovarlez G. Yielding to stress: Recent developments in viscoplastic fluid mechanics. *Annu Rev Fluid Mech* 2014;46:121–46.
- [32] Mitsoulis E, Tsamopoulos J. Numerical simulations of complex yield-stress fluid flows. *Rheol Acta* 2017;56:231–58.
- [33] Dimakopoulos Y, Pavlidis M, Tsamopoulos J. Steady bubble rise in Herschel–Bulkley fluids and comparison of predictions via the augmented Lagrangian method with those via the Papanastasiou model. *J Non-Newtonian Fluid Mech* 2013;200:34–51.
- [34] Dimakopoulos Y, Makrigiorgos G, Georgiou GC, Tsamopoulos J. The PAL (Penalized augmented Lagrangian) method for computing viscoplastic flows: A new fast converging scheme. *J Non-Newtonian Fluid Mech* 2018;256:23–41.
- [35] Glowinski R. Numerical methods for nonlinear variational problems. Springer series in computational physics, New York: Springer-Verlag; 1984.
- [36] Vinay G, Wachs A, Agassant JF. Numerical simulation of weakly compressible bingham flows: Application to the restart of waxy crude oils. *J Non-Newtonian Fluid Mech* 2006;136:93–105.
- [37] Papanastasiou TC. Flows of materials with yield. *J Rheol* 1987;31:385–404.
- [38] Tsamopoulos J, Dimakopoulos Y, Chatzidai N, Karapetsas G, Pavlidis M. Steady bubble rise and deformation in Newtonian and viscoplastic fluids and conditions for bubble entrapment. *J Fluid Mech* 2008;601:123–64.
- [39] Tripathi MK, Sahu KC, Karapetsas G, Matar OK. Bubble rise dynamics in a viscoplastic material. *J Non-Newt Fluid Mech* 2015;222:217–26.
- [40] Frigaard IA, Nouar C. On the usage of viscosity regularization methods for visco-plastic fluid flow computation. *J Non-Newtonian Fluid Mech* 2005;127:1–26.
- [41] Leclaire S, Reggio M, Trépanier J. Progress and investigation on lattice Boltzmann modeling of multiple immiscible fluids or components with variable density and viscosity ratios. *J Comput Phys* 2013;246:318–42.
- [42] Karapetsas G, Craster RV, Matar OK. Surfactant-driven dynamics of liquid lenses. *Phys Fluids* 2011;23:122106.
- [43] Craster RV, Matar OK. On the dynamics of liquid lenses. *J Colloid Interface Sci* 2006;303:503–16.
- [44] Boyer F, Lapuerta C, Minjeaud S, Piar B, Quintard M. Cahn–Hilliard/Navier–Stokes model for the simulation of three-phase flows. *Transp Porous Med* 2010;82:463–83.
- [45] Rowlinson JS, Widom B. Molecular theory of capillarity. Courier Corporation; 2013.
- [46] Liang H, Shi BC, Chai ZH. Lattice Boltzmann modeling of three-phase incompressible flows. *Phys Rev E* 2016;93:013308.
- [47] Yu Y, Liu H, Liang D, Zhang Y. A versatile lattice Boltzmann model for immiscible ternary fluid flows. *Phys Fluids* 2019;31:012108.
- [48] Kim J, Lowengrub J. Phase field modeling and simulation of three-phase flows. *Interfaces Free Bound* 2005;7:43–66.
- [49] Kumar A. Adaptive mesh refinement in OpenFoam-v1812 for 2-dimensional problems. 2019, <https://github.com/krajit/dynamicRefine2DFvMesh>.
- [50] Karapetsas G, Photeinos D, Dimakopoulos Y, Tsamopoulos J. Dynamics and motion of a gas bubble in a viscoplastic medium under acoustic excitation. *J Fluid Mech* 2019;865:381–413.
- [51] Prieto JL. Stochastic particle level set simulations of buoyancy-driven droplets in non-Newtonian fluids. *J Non-Newtonian Fluid Mech* 2015;226:16–31.
- [52] Hysing S, Turek S, Kuzmin D, Parolini N, Burman E, Ganesan S, et al. Quantitative benchmark computations of two-dimensional bubble dynamics. *Int J Num Method Fluids* 2009;60:1259–88.
- [53] Zainali A, Tofghi N, Shadloo MS, Yildiz M. Numerical investigation of Newtonian and non-Newtonian multiphase flows using ISPH method. *Comput Methods Appl Mech Eng* 2013;254:99–113.
- [54] Vahabi M, Sadeghy K. On the use of SPH method for simulating gas bubbles rising in viscoelastic liquids. *Nihon Reoraji Gakkaishi* 2015;42:309–19.
- [55] von Danwitz MHW. Simulation of rising bubbles in viscoelastic fluids [Master Thesis], Aachen, Germany: RWTH Aachen University; 2015.
- [56] Amirmia S, de Bruyn JR, Bergougnou MA, Margaritis A. Continuous rise velocity of air bubbles in non-Newtonian biopolymer solutions. *Chem Eng Sci* 2013;94:60–8.
- [57] Ohta M, Kobayashi N, Shigekane Y, Yoshida Y, Iwata S. The dynamic motion of single bubbles with unique shapes rising freely in hydrophobically modified alkali-soluble emulsion polymer solutions. *J Rheol* 2015;59:303–16.
- [58] Popinet S. An accurate adaptive solver for surface-tension-driven interfacial flows. *J Comput Phys* 2009;228(16):5838–66.
- [59] Gamet L, Scala M, Roenby J, Scheufler H, Pierson J-L. Validation of volume-of-fluid openfoam isoAdvector solvers using bubble benchmarks. *Comput & Fluids* 2020;213:104722.