Evaluation of two-phase flow solvers using Level Set and Volume of Fluid methods[☆]

Bilger C.[a,*], Aboukhedr M.[b], Vogiatzaki K.[c], Cant R. S.[a]

[a]*Department of Engineering, University of Cambridge, UK*
[b]*Department of Mechanical Engineering, City University of London, UK*
[c]*Advanced Engineering Centre, University of Brighton, UK*

Abstract

Two principal methods have been used to simulate the evolution of two-phase immiscible flows of liquid and gas separated by an interface. These are the Level-Set (LS) method and the Volume of Fluid (VoF) method. Both methods attempt to represent the very sharp interface between the phases and to deal with the large jumps in physical properties associated with it. Both methods have their own strengths and weaknesses. For example, the VoF method is known to be prone to excessive numerical diffusion, while the basic LS method has some difficulty in conserving mass. Major progress has been made in remedying these deficiencies, and both methods have now reached a high level of physical accuracy. Nevertheless, there remains an issue, in that each of these methods has been developed by different research groups, using different codes and most importantly the implementations have been fine tuned to tackle different applications. Thus, it remains unclear what are the remaining advantages and drawbacks of each method relative to the other, and what might be the optimal way to unify them. In this paper, we address this gap by performing a direct comparison of two current state-of-the-art variations of these methods (LS: *RCLSFoam* and VoF: *interPore*) and implemented in the same code (OpenFoam). We subject both methods to a pair of benchmark test cases while using the same numerical meshes to examine a) the accuracy of curvature representation, b) the effect of tuning parameters, c) the ability to minimise spurious velocities and d) the ability to tackle fluids with very different densities. For each method, one of the test cases is chosen to be fairly benign while the other test case is expected to present a greater challenge. The results indicate that both methods can be made to work well on both test cases, while displaying different sensitivity to the relevant parameters.

Keywords: Multiphase flows, Interface representation, Conservative Level-Set, Volume of Fluid

1. Introduction

The representation of complex interface movement in real multiphase flows requires robust but also efficient numerical techniques. This is particularly important for applications where hydrodynamic breakup is a dominant feature. For example when high speed diesel jet atomisation is of interest, then the accurate representation of the interface close to the nozzle is of paramount importance in order to predict primary breakup and the subsequent droplet size distribution [1]. Another example is low capillary oil/gas flows within porous media where interface capturing techniques can help to understand better the highly complex nature of oil mobilisation and extraction [2].

[☆]This document is a collaborative effort.
[*]Corresponding author
Email address: cb757@cam.ac.uk (Bilger C.)

Attempts to simulate various multiphase flows have resulted in a range of different numerical approaches. Among them, implicit interface-capturing methods such as Volume of Fluid (VoF) and Level-Set (LS) have attracted significant attention for simulating flows involving extensive topological changes [3, 4, 5, 6, 7, 8, 9]. A comprehensive review on the subject was given by Tryggvason et al. [10]. The VoF method has become a contemporary standard in commercial and open-source CFD software, e.g. [11, 12]. On the other hand, LS methods are still used mainly within more specialised in-house research codes. These interface-capturing methods, although successful in many problems, have a number of well-known but still inadequately addressed numerical deficiencies which limit their accuracy [13, 14, 15]. The most important of these are associated with the accurate representation of the sharp interface in terms of its location and its advection while guaranteeing mass conservation, as well as the representation of the effect of the surface tension forces. Amongst other things, this requires the accurate computation of the interface normal and curvature, using the underlying liquid volume fraction or level set field.

In the VoF method [16] a volume fraction α is used to discriminate between the fluids in the domain. When $\alpha = 1$ the cell is fully filled with fluid 1 and when $\alpha = 0$ the cell is filled with fluid 2. The interface is located in the cells where α lies between [0,1]. Numerical challenges related to the advection of the interface in the context of VoF are well documented in the literature [10]. Intrinsic to the method is numerical diffusion of the interface, at a rate that is highly dependent on the mesh size [17]. The numerical diffusion can be reduced by using a geometrical reconstruction coupled with a geometrical approximation of the VoF advection [18]. The position of the interface is determined by calculating the interface normal and the intersection points of the interface with the cell faces. The main drawback of geometrical methods is their complexity for 3D applications, in particular when used in conjunction with an unstructured mesh. Alternatively, using a compressive algorithm, the convective term of the VoF advection equation can be discretised using a compressive differencing scheme designed to preserve the interface sharpness: examples include CICSAM by Ubbink and Issa [17], HRIC by Muzaferija and Peric [19], or the compressive model available within OpenFoam [11]. Compression schemes do not require any geometrical reconstruction of the interface and extension to three dimensions and unstructured meshes is straightforward. However, compression schemes are not always sufficient to eliminate numerical diffusion completely and additional treatment is needed [20].

The LS method was first developed by Osher and Sethian [21]. After being initially used in the context of multiphase incompressible flows by Sussman et al. [22], LS methods have matured into a promising numerical technique for accurate simulation of multiphase flows as shown by Losasso et al. [23]. Instead of using a continuous volume fraction variable, LS relies on a signed distance function ϕ to distinguish between the two fluids in the mixture. The function has a positive value in one fluid and a negative value in the other, while it takes the value $\phi = 0$ at the interface. The interface is advected by solving a transport equation for ϕ that is re-initialised periodically to recover the distancing property [24]. The LS approach inherently offers a sharp representation of the interface and accurate representation of the interfacial quantities such as the interface normal and curvature, as needed for computing accurate surface tension forces. However, as opposed to VoF approaches, mass conservation is not embedded in the formulation. It has been shown in Sussman et al. [22] that mass is lost due to the re-distancing procedure that may involve artificial displacement of the interface [25]. Moreover, the re-initialisation procedure increases the computational cost.

The numerical challenge of representing the surface tension force at the phase interface is present in both methods. Surface tension is commonly represented as a source term in the momentum equation, calculated using

the Continuous Surface Force (CSF) model of Brackbill et al. [8]. The calculation involves an approximation of the interface curvature from the gradients of either the VoF or LS function, as well as the calculation of the normal to the interface. Since the interface is a discontinuous function the calculation of derivatives is problematic. For VoF methods that suffer from diffusion, it is difficult to achieve an accurate representation of interface curvature. For LS methods that provide a sharp interface, the effect of the volumetric surface force is confined to a narrow region around the interface and the calculation of the normal vector can be numerically unstable. These numerical effects are known to generate non-physical vortical flows at the interface, known as spurious currents [4, 26, 6]. Various methods have been developed to minimise spurious currents, such as (i) improvement of the curvature computation, (ii) achievement of discrete balance between surface tension and pressure gradient and (iii) use of an adaptive time integration scheme to tackle the stiffness induced by surface tension. One promising approach relies on minimal energy considerations and can eliminate spurious currents to machine precision. Alternatively, it is proposed to use consistent volume fluxes in the Navier–Stokes equations either in the VoF framework [27] or in the LS framework [28] using geometric arguments. It should be noted that the generation of spurious currents is not usually considered of great importance for inertia-dominated flows. However, it is detrimental in the computation of capillary flows and surface tension dominated flows, for example in hydrodynamic flows involving atomisation and pore-scale modelling. More recently it was shown that spurious currents also affect droplet collision with surfaces by preventing the transition to the recoil phase [29]. Moreover for flows with large density ratios, the interfacial force imbalance is large and thus generation of spurious currents is more likely [28, 10].

Hybrid methods have been proposed to combine the advantages of the VoF method in terms of mass conservation even on coarse meshes and the LS method in terms of accuracy of the interface contour and a smoothly differentiable field for the calculation of the surface tension forces. A fully coupled CLSVOF method was proposed by Sussman and Puckett [30] and has been implemented by a number of researchers since. The CLSVOF method has been applied on cartesian orthogonal meshes, successfully tested byMénard et al. [31, 32] on diesel jet atomisation, and by Yokoi [33] on the problem of droplet splashing and on the Rayleigh–Taylor instability. A recent development by Arienti and Sussman [34] has reformulated the CLSVOF method to adaptive cartesian meshes. In addition, some recent works have applied CLSVOF on unstructured meshes [35, 36].

Implementation of either the VoF or the LS method in a CFD code is very complex. Two-phase flow codes developed in-house by research groups have demonstrated remarkable results bypassing some of the shortcomings, but such codes are commonly tuned for specific sets of operating conditions that usually restrict the generality of the suggested remedies. Commercial codes offer much greater generality, but do not usually offer source-code access to the detailed numerical algorithms. The commercial open-source CFD toolkit OpenFoam incorporates a multiphase flow solver *interFoam* [37], and has been attracting an increasing amount of attention [38]. The advantage of OpenFoam is that it allows source-level modifications to the two-phase flow treatment within the same basic solver environment. The present study documents a detailed comparison in physical and computational performance of two novel in-house state-of-the-art two-phase flow solvers, i.e. the modified VoF solver (*interPore*) and the modified LS solver (*RCLSFoam*). Both are implemented within the same basic OpenFoam Navier-Stokes solver. These two solvers were initially tailored to tackle different classes of physical problems, namely surface-dominated microfluidics flows (*interPore*) and high-speed liquid atomisation (*RCLSFoam*). In this paper, the two solvers are compared using test cases designed to expose their relative

strengths and weaknesses. One test case is chosen to challenge the ability of each method to minimise the occurrence of spurious currents, while the other is designed to challenge the ability of each method to capture the dynamics of the phase interface.

The paper is organised as follows. The numerical description of both modelling algorithms is first given in Section 2, with special attention to the different treatment of the phase interface in each approach. The first test case, consisting of a three-dimensional stationary spherical droplet in equilibrium, is described in Section 3.1. Results are compared in terms of the development of spurious currents at steady state. The second test case involves a developing Rayleigh-Taylor instability and is shown in Section 3.2. The evolution of the instability is tracked and the solvers are compared in terms of their numerical and physical performance in capturing the interface dynamics including the production of small droplets.

2. Mathematical formulation

2.1. Solution of the incompressible Navier–Stokes equations and surface tension forces

Both numerical methods presented in this paper are implemented within the open-source CFD toolkit OpenFoam [11] that solves for incompressible two-phase flows with constant densities ρ_g and ρ_l and viscosities μ_g and μ_l. A single set of equations is used to describe mass and momentum conservation as follows:

$$\frac{D}{Dt}(\rho \boldsymbol{u}) = \nabla \cdot \mathbf{T} - \nabla p + \boldsymbol{f} \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2}$$

where $\mathbf{u}$ is the fluid velocity, p is the pressure and ρ is the density. The pressure-velocity coupling is handled using the Pressure-Implicit with Splitting Operators (PISO) method of Issa [39]. The term $\nabla \cdot \mathbf{T} = \nabla \cdot (\mu \nabla \boldsymbol{u}) + \nabla \boldsymbol{u} \cdot \nabla \mu$ is the viscous stress tensor. The term $\boldsymbol{f} = \boldsymbol{f_g} + \boldsymbol{f_c}$ corresponds to all the external forces, i.e. $\boldsymbol{f_g} = \rho \boldsymbol{g}$ is the gravitational force and $\boldsymbol{f_c}$ represents the capillary forces which for the case of constant surface tension coefficient σ are defined as:

$$\boldsymbol{f_c} = \sigma \kappa \delta_\Gamma \tag{3}$$

where δ_Γ denotes a distribution concentrated on the interface. The sharp interface Γ represents a discontinuous change of the properties of the two fluids. The dynamics of the interface are determined by the Young-Laplace balance condition:

$$\Delta \mathrm{P}_{\mathrm{exact}} = \sigma \kappa \tag{4}$$

where $\Delta \mathrm{P}_{\mathrm{exact}}$ is the pressure difference across the fluid interface. The CSF description of Brackbill et al. [8] is used to represent the surface tension forces in the cell volume. The quantity $\kappa = -\nabla \cdot (\boldsymbol{\eta_s})$ in Eq. 3 is the interface curvature and $\boldsymbol{\eta_s}$ the normal to the interface given by:

$$\boldsymbol{\eta_s} = \frac{\nabla \gamma}{|\nabla \gamma|} \tag{5}$$

Although the two methods use the same set of Navier–Stokes equations for mass and momentum conservation and are also based on the CSF description of the surface tension, their main difference lies in the calculation of κ in Eq. 4 as well as the normal $\boldsymbol{\eta_s}$ in Eq. 5. The indicator function γ is defined according to each solver's own numerical approach, either as a volume fraction α or as a conservative level set function ψ (see Section 2.2 and Section 2.3).

2.2. Computational framework for the RCLS method

The RCLS method is based on the conservative level-set method (CLS) of Olsson and Kreiss [40] and Olsson et al. [41] that transports the interface accurately while conserving mass. The indicator function γ, introduced in Section 2.1, here takes the form of a hyperbolic tangent function ψ ($\gamma = \psi$), defined as follows:

$$\psi = \frac{1}{2}\left(\tanh(\frac{\phi}{2\epsilon}) + 1\right). \tag{6}$$

where ϕ is the signed distance to the interface satisfying $|\nabla\phi| = 1$. The conservative level-set field (or CLS field) ψ goes smoothly from zero (in the gas phase) to unity (in the liquid phase). It takes values between zero and unity in the transition region in a similar way to α in the VoF approach (see Section 2.3). In this manner, the level-set ψ is identified as a smeared out liquid volume fraction. The location of the interface is given by the level-set value $\psi_\Gamma = 0.5$. This equation has been shown to guarantee mass conservation [40, 41, 42]. The main parameter of the CLS method is the coefficient ϵ that controls the width of the hyperbolic tangent profile in Eq. 6. Therefore, ϵ effectively controls the thickness of the numerical representation of the phase transition. Stability studies by Olsson et al. [41] and Pringuey and Cant [42] recommend values of $\epsilon > 0.5\overline{\Delta x}$, where $\overline{\Delta x}$ is the characteristic mesh element size. Since a sharp interface provides a better representation of the physics, it is preferable to model the interface to be as thin as possible, as well as to minimise the effect of smearing of the material properties and the surface tension. Nevertheless, the interface needs to have a minimal thickness, so that the gradient of ψ and the interface normal are accurately calculated in the CSF method. According to the CSF method, the material properties over the whole multiphase domain are given by:

$$\rho = \rho_g + (\rho_l - \rho_g)\psi \tag{7}$$

$$\mu = \mu_g + (\mu_l - \mu_g)\psi \tag{8}$$

The function ψ in Eq. 6 is advected in a conservative manner, in a zero-divergence velocity field, using an arbitrarily high-order Weighted Essentially Non-Oscillatory (WENO) scheme developed for three-dimensional mixed-element unstructured meshes by Pringuey and Cant [43] in order to handle complex real-world geometries:

$$\frac{\partial\psi}{\partial t} + \nabla \cdot (\mathbf{u}\psi) = 0 \tag{9}$$

WENO schemes have the ability to preserve the required sharpness of the interface in front-propagating problems and to cope efficiently with the large gradients associated with discontinuities in material properties resulting from the presence of the phase interface. Although the scheme can formally reach arbitrarily high-order in space, here it is restricted to a third-order polynomial reconstruction. The numerical fluxes are calculated using an exact Riemann solver for both Eq. 9 and Eq. 10 [42]. Accurate calculation of the face-averaged gradient $\overline{\nabla\phi}$ for Eq. 5 and surface tension forces (Eq. 3) is important to avoid spurious oscillations [42, 44].

Even high-order numerics such as the WENO scheme will eventually diffuse the interface. As a result, there is a need to re-initialise the level set profile ψ to maintain the interface thickness at a constant value. Here a re-initialisation algorithm is applied in which a compressive flux and a diffusive flux are applied in the direction normal to the interface. The technique ensures mass-conservation to machine accuracy:

$$\frac{\partial\psi}{\partial\tau} + \nabla \cdot (\psi(1-\psi)\boldsymbol{\eta_S}) - \epsilon\nabla \cdot (\nabla\psi) = 0 \tag{10}$$

where $\boldsymbol{\eta_S}$ is the normal to the interface and τ is the artificial time in which the equation is solved until the initial level set profile is recovered.

A third-order Total Variation Diminishing (TVD) Runge–Kutta (RK) scheme for temporal discretisation is used for both the advection and re-initialisation steps [42]. However, WENO schemes are not necessarily Total Variation Bounded (TVB), even when coupled with a Runge–Kutta time integration. Non-physical values of ψ such that $\psi < 0$ or $\psi > 1$ cannot be tolerated. As a result, the Multidimensional Universal Limiter with Explicit Solution (MULES) of Weller et al. [11] is employed in OpenFoam to maintain the boundedness of the conservative level-set profile. A description of the MULES limiter is given in Pringuey and Cant [42].

In summary, the numerical parameters of the RCLS method are: the order of the polynomial reconstruction of the high-order WENO scheme r, the coefficient controlling the thickness of the interface ϵ, and the interval of the re-initialisation N_S.

The relative performance of the RCLS method with respect to established multiphase numerical methods such as OpenFoam multiphase flow solver *interFoam* and the Accurate Conservative Level Set (ACLS) method of [44, 45] has been assessed and discussed in previous work [42] using canonical numerical test cases, and *RCLSFoam* was shown to perform well in terms of accuracy and robustness.

2.3. Computational framework for the modified VoF method

The VoF two-phase incompressible flow solver *interPore* developed at City University and the University of Brighton [29] is described briefly in this section. The approach is based on a compressive/smoothing treatment of the phase interface, in which the indicator function γ introduced in Eq. 5 is used to represent the liquid volume fraction α in each computational cell. It should be noted that the default two-phase flow solver *interFoam* as supplied with OpenFoam is also based on a VoF formulation, but lacks a numerical formulation for correcting capillary forces that would permit simulations at very low capillary numbers and avoid non-physical velocities.

2.3.1. Numerical treatment of the indicator function (compression and smoothing scheme)

An artificial compression term is used in the present VoF approach to sharpen the interface. The indicator function is advected according to Eq. 9, however an extra compression term is added to it. The new evolution equation for the liquid volume fraction α in the VoF framework reads:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (u\alpha) - \nabla \cdot \left\{ u_r \alpha \big((1-\alpha) \big) \right\} = 0 \tag{11}$$

where u_r is the relative velocity at the faces of each mesh cell, formulated based on the maximum velocity magnitude at the interface and its direction:

$$u_r = \min\left(C_{\text{comp}} \frac{|\phi_f|}{|S_f|}, \max\left[\frac{|\phi_f|}{|S_f|} \right] \right) (\boldsymbol{\eta_s} \cdot S_f) \tag{12}$$

where ϕ_f is the volumetric flux, S_f is the outward-pointing cell-face area vector taken at the centre of the face, and C_{comp} is the scalar compression factor for limiting the artificial compression velocity as implemented in *interFoam*.

The relative velocity u_r applies only within the phase interface region. It is calculated in the normal direction to the interface to minimise any dispersion errors, and it acts in such a way that the local flow steepens the gradient of the phase indicator function. Physically, u_r is the relative velocity between the two fluids, arising from the density and viscosity changes across the interface.

After solving Eq. 11, the indicator function α is updated at the cell centres and interpolated linearly to the cell faces in order to estimate $\boldsymbol{\eta_S}$ and κ. As a result of the compression step, the steep change in the value of

α creates errors in the estimation of the normal vector (Eq. 5) and the interface curvature. These errors act to induce non-physical spurious currents in the interfacial region, and these can be suppressed by computing the interface curvature from a smoothed liquid volume fraction α_{smooth}, as proposed by Lafaurie et al. [26], Georgoulas et al. [46] and Raeini et al. [47]. Here the approach implemented by Georgoulas et al. [46] is used. The liquid volume fraction α is first smoothed by interpolating it from cell centres to face centres and back recursively:

$$\alpha_{i+1} = \frac{1}{2}\left\langle \langle \alpha_i \rangle_{c \to f} \right\rangle_{f \to c} - \frac{1}{2}\alpha_i \tag{13}$$

Then, the function α_{i+1} (Eq. 13) is transformed into a smoother function α_{smooth} via a Laplacian model:

$$\alpha_{\text{smooth}} = \frac{\sum_{f=1}^{n}(\alpha_{i+1})_f S_f}{\sum_{f=1}^{n} S_f} \tag{14}$$

where n represents the number of smoothing loops and $(\alpha_{i+1})_f$ is the linearly-interpolated face-centre value. The smoothing operation can be repeated several times, although smoothing tends to level out high curvature regions and increase the interface thickness. Hence it should be applied only as often as strictly necessary to suppress spurious currents. There is a fine balance between smoothing the interface and altering its curvature.

The solution to Eq. 14 is then used to compute the interface normal (Eq. 5) and the interface curvature. The optimum number of smoothing loops is found to be of the order of 10.

After compressing and smoothing α an additional procedure is implemented in order to maintain the interface sharpness. A sharpened function, α_{sh} is introduced, as suggested by Raeini et al. [47] for the modelling of capillary pressure in the case of flows through porous media:

$$\alpha_{sh} = \frac{1}{1 - C_{\text{sh}}}\left[\min\left(\max(\alpha, 1 - \frac{C_{\text{sh}}}{2}), 1 - \frac{C_{\text{sh}}}{2} \right) - \frac{C_{\text{sh}}}{2} \right] \tag{15}$$

where C_{sh} is a sharpening coefficient. It limits the effect of unphysical values at the interface by imposing a restriction on α. For $C_{\text{sh}} = 0$ the original VoF formulation is recovered. As C_{sh} approaches 1, α_{sh} becomes sharper, hence the estimation of capillary forces becomes more accurate.

2.3.2. Capillary pressure jump modelling

An important aspect that differentiates the present approach from that implemented in OpenFoam is the implicit treatment of the capillary pressure jump. Following the calculation of interface normal and interface curvature, the capillary forces are calculated at the face centres based on Eq. 3. However in order to avoid difficulties associated with the discretisation of the singular force term $\boldsymbol{f_c}$, the terms on the right hand side of the momentum equation are rearranged following Raeini et al. [47]:

$$\frac{D}{Dt}(\rho\boldsymbol{u}) - \nabla \cdot T = -\nabla p_d + \boldsymbol{f'} \tag{16}$$

where

$$\boldsymbol{f'} = \rho g + \boldsymbol{f_c} - \nabla p_c \tag{17}$$

This approach, in which the capillary force term appears explicitly in the Navier–Stokes equations, enables the filtering of the numerical errors related to inaccurate calculation of $\boldsymbol{f_c}$. Pressure jumps across a phase interface

result from normal stresses, and do not contribute to a jump in tangential stress. Consequently, tangential surface stresses can only be balanced by viscous stresses associated with fluid motion. Therefore, a boundary condition such that:

$$\frac{\partial p_c}{\partial \eta} = 0 \tag{18}$$

is applied, where η is the normal direction to the boundary. Spurious currents will still exist close to non-flat (curved or edged) boundaries. These non-physical velocities are parallel to the fluid interface. Therefore, components of the capillary forces that result in spurious currents parallel to the fluid interface are filtered using the following model:

$$\phi_{\text{filter}} = |S_f| \, (f - \nabla p_c) - \max \Big(\min \Big(|S_f|(f - \nabla p_c), \phi_{\text{threshold}} \Big), -\phi_{\text{threshold}} \Big) \tag{19}$$

Eq. 19 is a simple threshold scheme to filter the capillary flux components of the capillary forces that result in spurious currents parallel to the fluid interface. This filtering will explicitly set the capillary fluxes to zero when their magnitude is of the order of the numerical errors. The filtered capillary flux threshold is defined as:

$$\phi_{\text{threshold}} = U_f \, |f|_{\text{avg}} \, |S_f| \tag{20}$$

where $\phi_{\text{threshold}}$ and U_f are threshold values below which the capillary fluxes are set to zero. The filtering coefficient U_f is used to minimise the errors in the capillary fluxes and $|f|_{\text{avg}}$ is the average value of capillary forces over all faces. By eliminating only a small fraction of the capillary fluxes, the filtering operator minimises numerical errors in the calculation of capillary forces that cause instabilities or introduce large errors in the velocity field. It also reduces the stiffness of the problem by eliminating high-frequency capillary waves when the capillary forces are close to equilibrium with capillary pressure, allowing larger time-steps to be used when modelling interfacial motion at low capillary numbers. Finally, non-physical velocities that are parallel to the fluid interface are filtered, so that the term $f_c - \nabla p_c$ that appears in the Navier–Stokes equations (Eq. 1) reduces to zero.

Crank-Nicholson and Euler schemes are used for time-advancement of the liquid volume fraction. The time-step is limited dynamically by introducing a numerical capillary time scale; the numerical solution is stable when the time-step resolves the propagation of capillary waves:

$$\delta t \leq \left[\frac{\rho_{av} \delta x^3}{2 \pi \sigma} \right]^{\frac{1}{3}} \tag{21}$$

The Courant–Friedrichs–Lewy (CFL) number is kept below 0.5 in order to maintain accuracy and stability. The solution procedure begins by advecting the liquid volume fraction for half of the time-step using the fluxes at the start of the time-step. Then the liquid volume fraction for the second half of the time-step, the capillary pressure, momentum and dynamic pressure are solved iteratively in two loops. The phase-fraction equation (Eq. 11) is solved for 2-3 sub- timesteps using the MULES limiter along with u_r from Eq. 12. Then the smoothing and sharpening operations are applied to α and the curvature and the normal are calculated. Filtering of the capillary flux follows. Once the updated phase field is obtained, the updated surface tension force is calculated and the pressure and velocity are corrected using the PISO algorithm, within the OpenFoam environment, with 2-3 iterations at each time-step. The main tuning parameters of the present VoF method

8

are the capillary filtering threshold U_f (Eq. 20), the number of smoothing loops n (Eq. 14) and the amount of sharpening C_{sh} (Eq. 15).

3. Numerical results

3.1. Selection of test cases

In the following sections, numerical simulations of two benchmark cases are presented that demonstrate the performance of the two different solvers which have been developed to tackle very different applications. *RCLSFoam* was developed for primary breakup modelling in airblast atomisers (high capillary numbers) while *interPore* was developed for oil extraction in porous media (low capillary numbers). The capillary number Ca is defined as:

$$\mathrm{Ca} = \frac{\mu U}{\sigma} \tag{22}$$

The dynamic viscosity μ and characteristic velocity U are those of the liquid and σ is the interfacial tension between the two fluid phases. The capillary number is a measure of the relative importance of viscous forces and capillary forces.

It is interesting to note that the capillary number is equal to the ratio of the Weber number to the Reynolds number (i.e. Ca = We/Re). Hence the capillary number encapsulates the major physical effects that operate in the two-phase flow configurations that are of interest in the present study. It should be noted that methods based on achieving optimum interface sharpness at higher Ca numbers are more prone to spurious currents at low Ca numbers. By the same token, methods that are more resistant to interface artefacts are less accurate in terms of sharpness. The natural tendency is to assess novel methods against benchmark cases that are suitable for the "real world" application that motivates the development in the first place. Hence it is unclear whether existing methods are suitable for a wide range of applications, going beyond their original design objectives. Here we have chosen to test each numerical method on at least one test case which is well outside the parameter range for which it was designed. In the first benchmark case, the determining factor is the elimination of spurious currents; the interface geometry is rather benign, however because of the low inertia environment in which it is evolving, it is particularly challenging to pick up the flow field. In the second benchmark case, the determining factor is the sharpness of the interface; a much more difficult interface geometry makes it challenging to track the small-scale structures.

The aim of the following sections is a) to compare the two frameworks in terms of their performance in two cases (one that matches the strength and one that challenges each case), b) to evaluate the effect of the control parameters and c) to extract conclusions relevant to the generality of the different methods suggested here.

3.2. Stationary deformation – circular droplet in equilibrium

When a square droplet of high density fluid is immersed in a low density fluid, the forces of surface tension cause it to deform and relax to a circular shape. In theory, the circular interface should then remain at rest, with the pressure jump at the interface exactly balancing the surface-tension force (Laplace's law). Depending on the method used for representing the surface tension force and the pressure gradient, an exact numerical balance is very difficult to obtain and spurious currents appear, triggered by this imbalance. Although the set-up is simple, the lack of inertia makes this case very challenging for methods not designed for this situation. In Harvie et al. [48] a correlation for the magnitude of spurious currents was presented as a function of the

physical and numerical parameters. It was demonstrated that spurious currents can severely limit the accuracy of the standard CSF technique.

In the following test case, a comparison between the RCLS method and the low Capillary number VoF method has been made, showing a numerical balance for Laplace's problem provided the shape of the interface is given enough time to relax to its numerical equilibrium shape. The numerical parameters for each method have been examined to demonstrate the efficiency of the two solvers, and are tabulated in Table 1.

Table 1: Table of varied numerical parameters for the Static Droplet simulation.

	RCLS			interPore				interFoam
Case 1	WENO3	$N_S = 5$	$\epsilon = 0.5\overline{\Delta x}$	$C_{\text{comp}} = 0$	$C_{\text{sh}} = 0.1$	$S = 10$	$U_f = 0.1$	$C_{\text{comp}} = 1$
Case 2	WENO3	$N_S = 5$	$\epsilon = 1.0\overline{\Delta x}$	$C_{\text{comp}} = 0$	$C_{\text{sh}} = 0.5$	$S = 10$	$U_f = 0.1$	$C_{\text{comp}} = 2$
Case 3	WENO3	$N_S = 5$	$\epsilon = 2.0\overline{\Delta x}$	$C_{\text{comp}} = 1$	$C_{\text{sh}} = 0.1$	$S = 10$	$U_f = 0.5$	

3.2.1. Numerical set up

The first test case models the relaxation process of an oil droplet of diameter $D_0 = 300$ μm in water at static equilibrium in the absence of gravity. A droplet with an initial cubic shape is introduced inside a three-dimensional cubic computational domain. The cubic droplet is allowed to relax to a static spherical shape as shown in Fig. 1. The force balance should converge to a correct solution of zero velocity everywhere and predict a pressure jump from a constant value p_0 outside the droplet to a value of $p_0 + 4\sigma/D_0$ inside the droplet. If this is not the case then non-physical vortex-like velocities can develop at the interface and result in the destabilisation of the droplet interface [8]. The fluid properties are as follows: the background (water) density ρ is 998 kgm^{-3}, and the viscosity ν is 1.004×10^{-6} m^{2}s^{-1}, while the droplet (oil) density ρ is 806.6 kgm^{-3}, and its viscosity ν is 2.1×10^{-6} m^{2}s^{-1}. The surface tension coefficient is fixed to 0.02 kgs^{-2}. Computational boundaries were modelled as no-slip walls and the domain was discretised using a uniform square mesh.

Different methods are used for initialisation for each solver. At $t = 0$, the liquid volume fraction α in *interPore* is initialised by setting $\alpha = 0$ in the cells containing water and $\alpha = 1$ in the cells containing oil, with a cubic shape as shown in Fig. 1. In the RCLS method the initialisation is not so straightforward. At $t = 0$, the signed distance function from the interface ϕ is calculated analytically. The function used to initialise ϕ is the implicit equation for a three-dimensional superquadric:

$$|x|^r + |y|^r + |z|^r = R^4 \tag{23}$$

In the limit of large r a cube is recovered. With this initialisation method, the phase interface is smeared over approximately 4 cells and remains constant during the simulation. The initial conservative level set field ψ of the RCLS method is then easily derived by applying Eq. 10. This allows ψ to vary from 0 (oil) to 1 (water) smoothly, with no interface discontinuity at initialisation.

The simulations with *RCLSFoam* were carried out on the Cambridge HPC cluster, which consists of 9600 2.60GHz Intel Sandy Bridge cores connected by Mellanox FDR Infiniband (600 nodes, 64GB of RAM per 16-cores node. A typical simulation on a $60 \times 60 \times 60$ grid with *RCLSFoam* (Case 1) required about 90 CPU hours to generate 0.01 seconds of simulated time. Calculations with *interPore* were made on an Intel Xeon machine with 64GB main memory possessing one 2.4Ghz CPU with 12 cores. A typical simulation on a $60 \times 60 \times 60$

grid with *interPore* (Case 3) required about 3.6 CPU hours to generate 0.01 seconds of simulated time. The disparity in computational cost for this test case indicates that *interPore* is very efficient at solving low inertia problems.
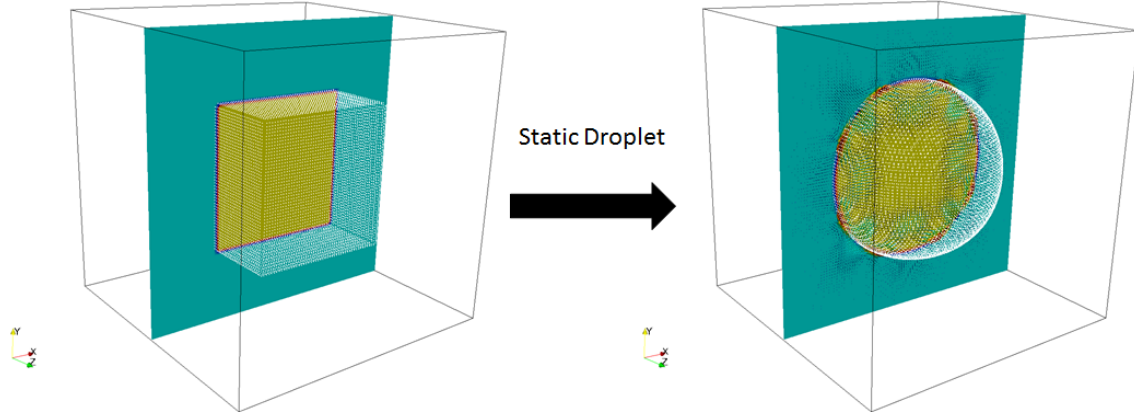

Figure 1: Solution domain for modelling the static droplet, (left) initial condition, a cube of size 600 μm, and (right) static shape of droplet at $t = 0.01$s.

3.2.2. Accuracy of the velocity field

Three simulations were run with each solver and with numerical parameters given in Table 1. As a first step of the comparison, the maximum magnitude of the spurious currents was recorded as a function of various numerical parameters, and then both numerical solvers were compared to the standard *interFoam* solver. Spurious currents are expected to be limited during the initial relaxation period by the transient inertial term. Looking at the velocity magnitude for the standard *interFoam* solver during the relaxation period (Fig. 2, *red lines*), it can be seen that even when only compression is used the spurious velocities remain fairly small. As the relaxation process progresses the compression term alone is not capable of restricting diffusion and the interface thickness increases. This has a stabilising effect in terms of spurious currents, but it creates a larger error in the prediction of the capillary pressure as noted in the next section.

The *interPore* solver (Fig. 2, *blue lines*) provides the lowest velocities (almost two orders of magnitude difference). By contrast, for *RCLSFoam* the spurious velocities (Fig. 2, *black lines*) are comparable with those of *interFoam*. It is worth mentioning that the maximum velocity magnitude recorded by *RCLSFoam* is sensitive to the value of the parameter ϵ. For higher values of ϵ, spurious velocities are reduced, with minimal overshoots during the relaxation. On the other hand, the *interPore* solver is found to reduce the maximum velocity down to very low values, regardless of the controlling parameters; however this happens over a longer time period. We now turn our focus to a) interface thickness, b) capillary pressure and c) the instantaneous velocity field. The results are compared in Fig. 3 and Fig. 4. For *RCLSFoam*, the value of ϵ shows a direct impact on the interface thickness (Fig. 3, *left column*) as expected and affects the pressure and the velocity of the surrounding field. For Case 3 (Fig. 3, *bottom row*), with $\epsilon = 2.0\overline{\Delta x}$, the spurious currents become very low (see also Fig. 2) but the interface becomes excessively diffused. For the *interPore* solver, comparisons in Fig. 4 show that the magnitude of the spurious currents is less sensitive to the interface thickness. For example for cases 1 and 3 (Fig. 4, *top row and bottom row*) the thickness of the interface is noticeably different while the order of magnitude of spurious currents (see also Fig. 2) remains comparable. Within *interPore* the interface capillary forces are filtered and thus spurious currents remain very weak, even with very thin interfaces. This feature of the

method is essential for the simulation of multiphase flows at low capillary numbers.

The influence of the sharp interface-tracking methodologies used in both solvers is shown in Fig. 5 (a), where the indicator function (α for *interPore* and ψ for *RCLSFoam*) is plotted across the x-axis of the droplet, after relaxation. When a low value of ϵ (smaller induced diffusion) is used for the RCLS method the results confirm the findings of Fig. 2. Since interface sharpness is increased and interface thickness is reduced, the spurious currents are increased. The spurious currents eventually act to move the droplet away from the centre (Case 1). With the *interPore* solver (Fig. 4), the magnitude of the spurious currents is reduced by two orders of magnitude compared to the standard *interFoam* solver when less sharpening is applied and the number of curvature smoothing loops is kept constant at $n = 10$ (Case 1). Moreover, the magnitude of the spurious currents consistently decreases when increasing the filtering threshold value (U_f) as shown in Fig. 2 (Case 3). Sensitivity analysis of the smoothing loop number n (not shown here) indicates that $n = 10$ is the optimum number that provides good results for a range of the other parameters.

It is interesting to note that the spurious currents tend to form distinctive patterns that are similar for both methods, as shown in the third column of Fig. 3 and Fig. 4. The patterns tend to emphasise the coordinate axes, where the alignment between the mesh and the interface is at its best, and also the diagonals, where the alignment is at its worst. Weak spurious vortices are formed in between these principal directions.
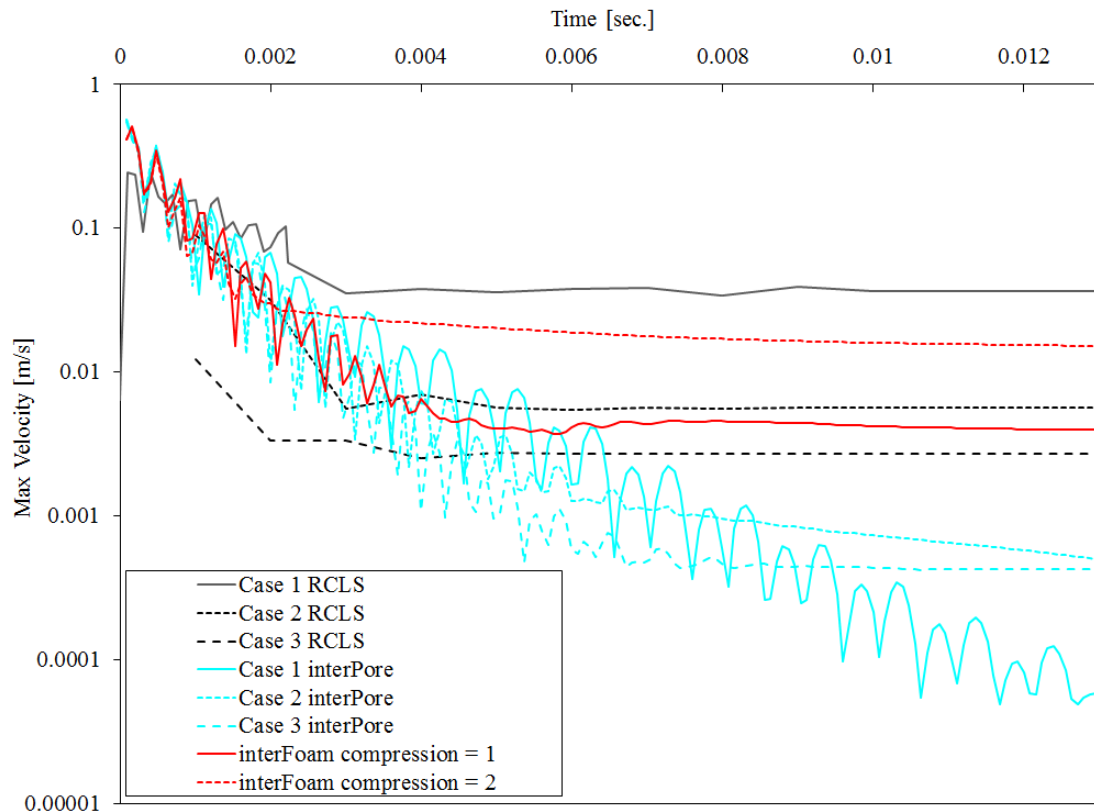

Figure 2: Comparison between the numerical predictions for spurious currents by the standard *interFoam* solver, the modified VoF solver (*interPore*) and the Conservative Level-Set solver (*RCLSFoam*).

3.2.3. Accuracy of the capillary pressure

Once equilibrium has been established, the shape of the droplet should be spherical and all fluid velocities should be zero. In this instance, the curvature everywhere on the interface surrounding the droplet should be

uniform and the interface normal should be directed everywhere in the radial direction. The non-dimensional
pressure should be uniform both inside and outside the droplet but undergo a step change of magnitude across
the interface. After the relaxation period, the surface tension force should be balanced precisely by the pressure
gradient. The pressure jump should be equivalent to the Laplace pressure for the droplet.

The numerical results differ from the ideal description, as shown in Fig. 5 (a). Small errors occur in calculating
the interface normal. This results in the normal vector not being precisely radial. The interface curvature will
therefore vary slightly, and the density will also vary due to the variations in the interface location. Finally, the
interface volume force will include a non-physical rotational component. In this situation, the surface tension
cannot be balanced by the irrotational pressure gradient term, so instead it will be balanced by one or more
of the inertial forces, all of which are velocity dependent. As these inertial effects (inertial transient, inertial
advection and viscous) all require non-zero velocities if they are to be non-zero, spurious currents develop. The
spurious currents increase with time and cause errors in the calculated pressure field. Fig. 5 (a) shows the
numerically predicted pressure difference between the relaxed spherical droplet and the ambient liquid along
the droplet diameter axis for each of the 6 cases, in comparison with the theoretical value predicted from the
Laplace equation [49].

The results show that *RCLSFoam* (Fig. 5 (a), *black lines*), for low ϵ values (Case 1 and 2), achieves a pressure
jump almost equivalent to the theoretical one. However, it fails to predict the correct pressure jump for Case
3 due to the high diffusion of the interface, which leads to an imbalance of the surface tension forces with the
pressure forces. The *interPore* (Fig. 5 (a), *blue lines*) solver provides a consistent prediction of the theoretical
pressure difference (Fig. 5 (a), *red line*) regardless of the interface thickness (Fig. 5 (b)).

These results can also be presented in terms of the error in the predicted capillary pressure, $Error(P_c)$, defined
as follows:

$$Error(P_c) = \frac{P_c - P_{c,theoretical}}{P_{c,theoretical}} \tag{24}$$

The results from Eq. 24 are tabulated in Table 2 and used to demonstrate that the calculated capillary
pressures converge to the theoretical solution when varying the interface sharpness. When the interface curvature
is obtained from a smoothed indicator function as seen in Case 1 and 2 with both *RCLSFoam* and *interPore*, it is
clear that both solvers can achieve a very low error, less than 3%, due to the precise calculation of curvature. By
contrast, the standard *interFoam* solver cannot achieve the desired theoretical pressure jump, and the average
error goes up to 12% (see Table 2).

Table 2: Errors in predicted capillary pressure

	RCLS	*interPore*	*interFoam*
Case 1	$Error(P_c) = 0.01499$	$Error(P_c) = 0.03868$	$Error(P_c) = 0.109605263$
Case 2	$Error(P_c) = 0.00921$	$Error(P_c) = 0.03556$	$Error(P_c) = 0.131954887$
Case 3	$Error(P_c) = 0.05359$	$Error(P_c) = 0.04311$	

It is clear that both solvers accurately reproduce the dynamics of a square droplet relaxing in a static field
given the correct selection of the controlling parameters. The *interPore* solver that is designed for very low
Ca numbers offers good predictions with very low spurious currents and small interface thickness, with the
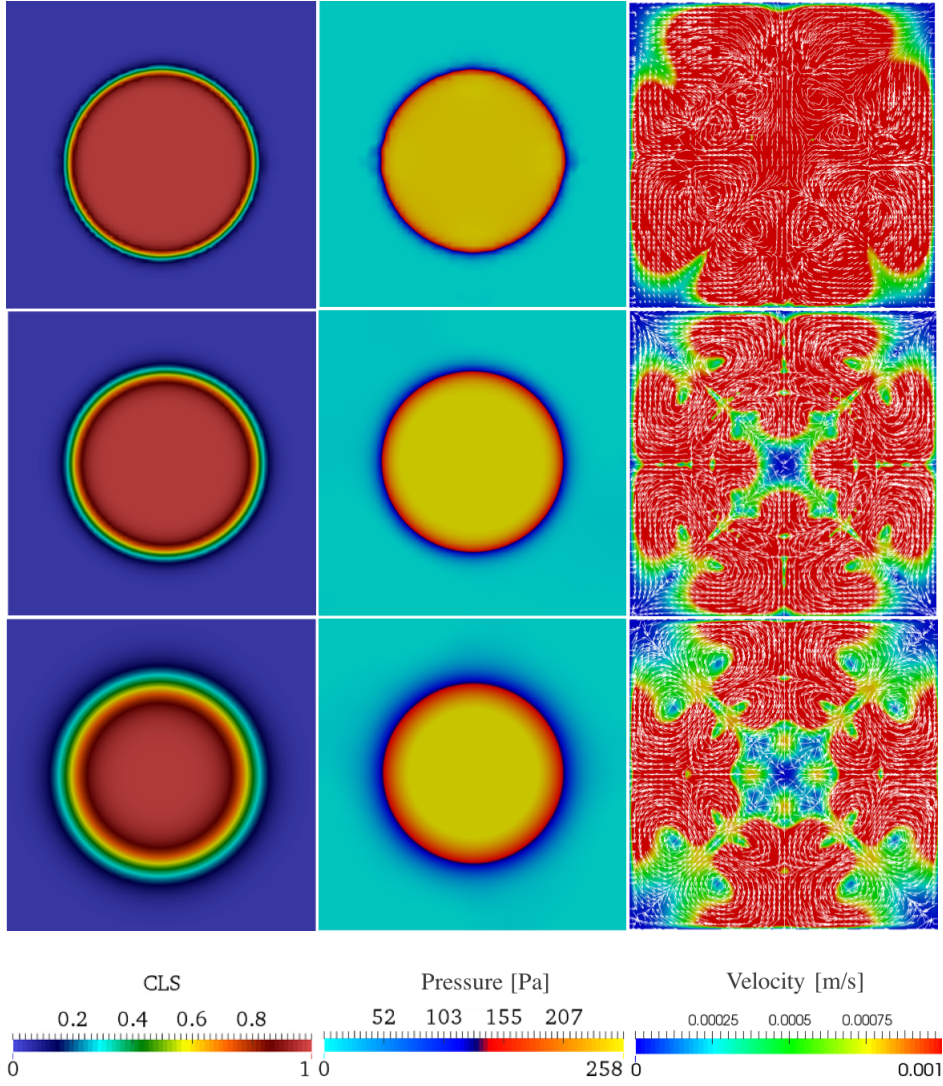additional advantage of showing reduced sensitivity to the control parameters. On the other hand, *RCLSFoam*

Figure 3: Results predicted by *RCLSFoam* for the static droplet at $t = 0.013$s. Top to bottom: Case 1, Case 2, Case 3. From left to right: indicator function, pressure field and velocity vector field.

offers better agreement with theoretical values for the capillary pressure, although it displays greater sensitivity to the control parameters in terms of interface thickness and the magnitude of spurious currents.

In a previous study mesh independence of the solution was checked more specifically for the RCLS method [42]. Here we conducted a mesh convergence study on the droplet relaxation case with both solvers down to a mesh element size of 6μm. Both methods are found to produce mesh independent solutions at the mesh element sizes used for the results shown below.

3.3. Rayleigh–Taylor instability with surface tension

The second test configuration consists of a Rayleigh–Taylor problem with homogeneous layers of a high-density fluid penetrating into a low-density fluid under the influence of gravity alone. A hydrodynamically-unstable interface is formed between the stratified laminar immiscible fluid layers, initially at rest. The Rayleigh–Taylor instability (RTI) is of great fundamental interest in physics and fluid mechanics. For example, in geological flows, it can represent situations where water is suspended above oil. In spray atomisation, it partly drives the breakup of liquids due to aerodynamic forces on the phase interface. For the present unstable configuration, infinitesimally small perturbations of the interface will grow with time due to the continuous
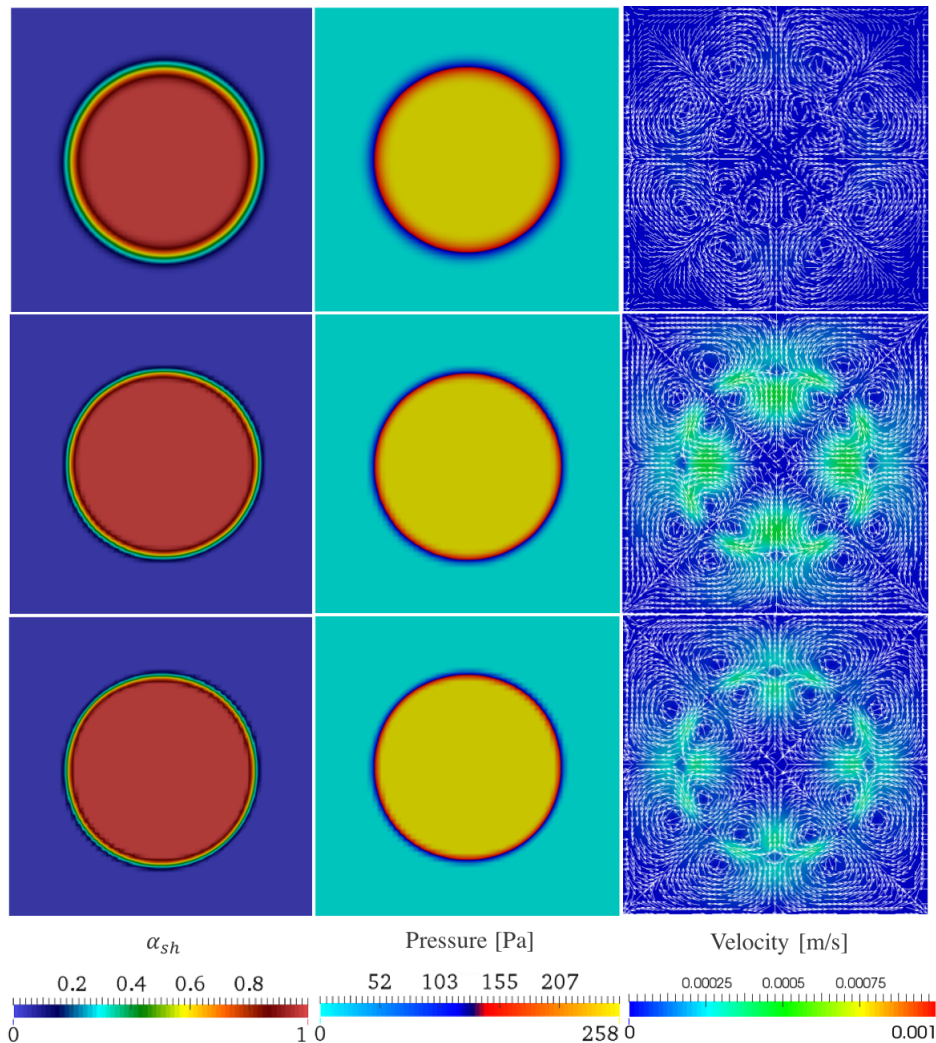
Figure 4: Results predicted by *interPore* for the static droplet at $t = 0.013$s. Top to bottom: Case 1, Case 2, Case 3. From left to right: indicator function, pressure field and velocity vector field.

generation of vorticity that leads to a pattern of bubbles of light fluid penetrating the heavy fluid (upwards), and spikes of heavy fluid penetrating the light fluid (downwards). This process may lead to the formation of a turbulent mixing layer. The RTI was initially investigated by Lord Rayleigh in 1883 [50] and later theoretically studied by Taylor [51]. An overview of the subject has been given by Sharp [7]. In the following sections the capability of the VoF method as implemented in the *interPore* solver is evaluated along with the conservative LS method as implemented in the RCLS solver, for the modelling of a single "finger", or "spike", of the RTI with surface tension. The numerical demonstration of the RTI is a common test case when examining the numerical convergence and the capability of a solver to capture the interface physics in the presence of high density ratio and surface tension [52, 9, 53]. We refer the reader to a previous study by Pringuey and Cant [42] that assessed the performance of *RCLSFoam* against *interFoam*. To the best of the authors knowledge, there has not been a systematic comparative study of state-of-the-art LS and VoF numerical methods on the RT problem.

3.3.1. Numerical set-up

The two isothermal fluids are at ambient temperature and pressure. The viscosities are assumed equal, so that there is no jump in viscosity at the interface and thus no jump in the velocity gradient, as viscosity controls
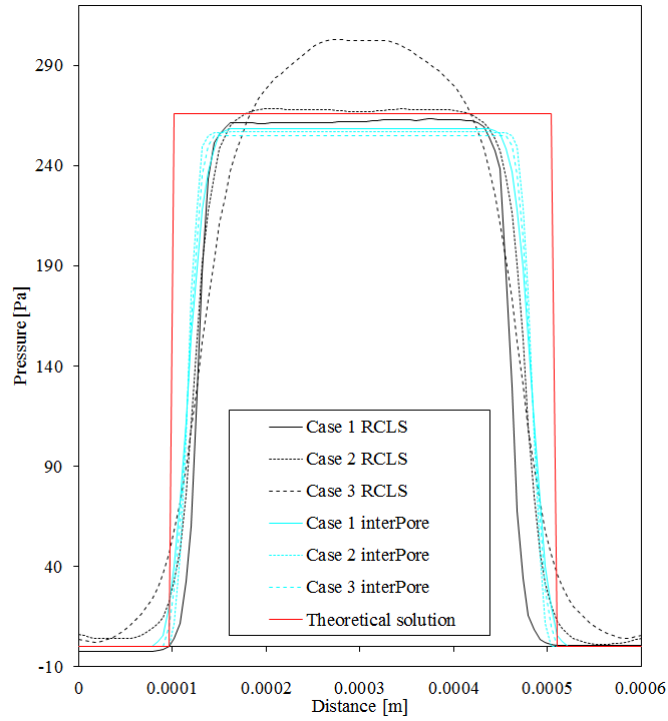
(a) Laplace pressure jump across the droplet interface
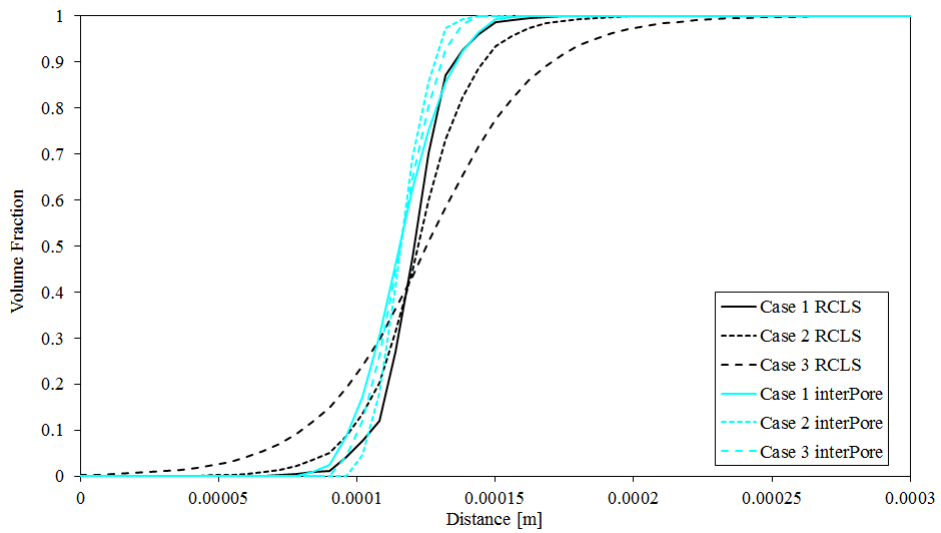
(b) Indicator function profile across the droplet interface

Figure 5: Comparison between the numerical predictions of both solvers showing a) the Laplace pressure jump across the droplet interface, and b) the liquid volume fraction profile.

Table 3: Table of numerical parameters for the Raleigh–Taylor simulation for two different mesh sizes.

RCLS			interPore				60×224	128×512
WENO3	$N_S = 5$	$\epsilon = 0.5\overline{\Delta x}$	$C_{\text{comp}} = 0$	$C_{sh} = 0.1$	$S = 10$	$U_f = 0.1$	Case 1	Case 4
WENO3	$N_S = 5$	$\epsilon = 1.0\overline{\Delta x}$	$C_{\text{comp}} = 0$	$C_{sh} = 0.5$	$S = 10$	$U_f = 0.1$	Case 2	Case 5
WENO3	$N_S = 5$	$\epsilon = 2.0\overline{\Delta x}$	$C_{\text{comp}} = 1$	$C_{sh} = 0.1$	$S = 10$	$U_f = 0.5$	Case 3	Case 6

the shear stress. The two adjacent fluids are initially quiescent but the sharp horizontal density mismatch induces a density gradient (upwards) opposite to the pressure field and gravitational field (downwards). The interface is hydro-dynamically accelerated by the pressure gradient [7]. The gravitational forces act to destabilise the interface, while the surface tension forces act to stabilise it.

The present set of simulations for comparison and validation are based on the settings for RTI given in Popinet and Zaleski [9] and Puckett et al. [53] on a comparatively coarse mesh of 64×224 hexahedral elements and a finer one of 128×512 elements. The two-dimensional domain is $[-0.5, -2] \times [0.5, 2]$ m (Fig. 6). The physical parameters for the fluids are $\rho_l = 0.1694 \text{kgm}^{-3}$, $\mu_l = 3.13 \times 10^{-3} \text{kgm}^{-1}\text{s}^{-1}$ and $\rho_h = 1.255 \text{kgm}^{-3}$, $\mu_h = 3.13 \times 10^{-3} \text{kgm}^{-1}\text{s}^{-1}$ (density ratio of 7.4), with a constant surface tension of $\sigma = 0.01 \text{Nm}-1$. The acceleration due to gravity is set as $g = (0, -9.8 \text{ms}^{-2}, 0)$.

An initial surface perturbation is applied with a wave number of 2π and wavelength of 1 m. The initial disturbance in the free surface is given by the expression:

$$y = -0.05 \cos(2\pi x). \tag{25}$$

For the phase fraction and level-set, a homogeneous Neumann boundary condition was set at the bottom boundary of the domain and symmetry conditions were set at both sides. The same boundary conditions were set for the relative pressure $p_d = p + \rho g h$, as defined in the framework of OpenFoam. In multiphase flow dynamics, the time-step Δt must be small enough to resolve the propagation of the capillary waves that develop at the interface. The time-step was set initially as $\Delta t = 0.001$s, but it was allowed to adapt automatically at run time, which assured a Courant number below 0.5 throughout the run. Finally, the boundary conditions for the velocity were set as no-slip walls for the top and bottom boundaries and symmetry planes at both sides. The chosen set of simulation parameters is summarised in Table 3. Calculations with *RCLSFoam* were made on an Intel Xeon machine with 32GB main memory possessing one 2.4Ghz CPU with 8 cores. The simulations with *interPore* were carried out on an Intel Xeon machine with 64GB main memory possessing one 2.4Ghz CPU with 12 cores. The total execution time on a 128×512 grid for *RCLSFoam* was 3.6 CPU hours, while that for the interPore *interPore* was 17 CPU hours to generate 1.5 seconds of outputted results. It is interesting to note that for this test case the disparity in computational cost between the two codes is reversed. Here *RCLSFoam* is more efficient at dealing with the rapidly-evolving interface, as well as the fast-changing velocity and field.

3.4. Qualitative Description

The relative performance of *RCLSFoam* with respect to *interPore* is presented in Fig. 7. This figure shows the interfacial history obtained by the two solvers on the same rectangular mesh of 60×224 cells, for eight different times in the interval $[0; 1.3]$ seconds. It can be seen that as time passes, the initial perturbation of the interface grows. The light fluid moves into the heavy fluid in the form of rising round bubbles on either side
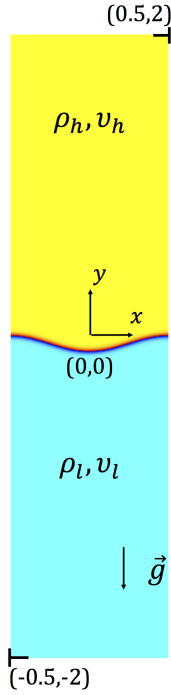
Figure 6: Computation domain for the simulation of the RTI.

of the domain, while the heavy fluid sinks into the light fluid under gravity, between the bubbles, building up a central falling spike(Fig. 7 (a) (i) $t = 0.55$s). A mushroom-shaped structure develops at $t = 0.7$s. Then, the mushroom is stretched and filaments start to detach forming isolated liquid structures (at $t = 0.95$s). Towards $t = 1.3$s, a settling process begins in which the denser fluid falls back down as droplets.

The complex phenomenology associated with the evolution of a RT unstable interface is reproduced reasonably well here by both solvers. The evolution of the instability compares well with the calculations presented in the literature [9, 53]. Indeed, the solutions predicted by both numerical approaches remain close to the reference solutions obtained with an accurate front tracking method [9]. The two numerical methods appear to provide the same interface topology for most of the simulation. Nevertheless, differences are evident in terms of interface thickness, penetration rate and secondary structure formation (filaments,droplets etc) as will be discussed in the following sections.

3.4.1. Interface capture and sharpness

The sharpness of the phase boundary between the two fluids is an indication of the numerical diffusion of the scheme used. Also, the amount of fine-scale structure is sensitive to how well the algorithm preserves the sharpness of the interface. The works of Popinet and Zaleski [9] and Puckett et al. [53] suggested that the correct capture of the thin ligaments has a strong influence on the overall simulation results in terms of penetration of the dense phase and opening of the mushroom cap. As can be seen from Fig. 7 both simulation methods are able to capture the thin liquid structures, and maintain a sharply defined interface. In the RCLS method, as the interface thickness parameter ϵ is increased from $0.5\overline{\Delta x}$ to $2\overline{\Delta x}$, the interface becomes smeared over an increasing number of cells; there is a loss of sharp numerical resolution of very small secondary liquid structures (Fig. 7 (a)). The re-initialisation procedure integrated into *RCLSFoam* helps in capturing correctly the formation of droplets. The more diffused the interface, i.e. the bigger the interface thickness parameter ϵ, the less visible are the droplets (compare Case 1 to Case 3 in Fig. 7 (a)). In addition, the rate of high-density

18

fluid penetration also decreases: for Case 3, at $t = 1.3$s the mushroom tip does not yet touch the bottom wall (Fig. 7 (a) (iii)).

Attention is then turned to the solution obtained with the *interPore* solver (Fig. 7 (b)). A higher interface sharpening coefficient C_{sh} with no numerical compression (Fig. 7 (b) (ii)) is able to capture a sharper interfacial jump than with a lower C_{sh}. When numerical compression is added (Fig. 7 (b) (iii)) the interface is even sharper. Overall the comparison of the interface predicted by the two solvers for the first four times ($t = 0.55, 0.70, 0.85, 0.95$ seconds) demonstrates the similar performance of both simulation methods. The *interPore* solver appears less sensitive to the tuning parameters than *RCLSFoam*, which is consistent with observations in the case of the static droplet. A disadvantage is while the VoF method is capable of reproducing the ligaments formed by the elongation of the mushroom sides when low compression is used these ligaments disappear as C_{sh} increases and compression is added. The RCLS method on the other hand seems to be better able to treat the small-scale inclusions of one phase into the other regardless of the ϵ parameter. Indeed, for times beyond $t = 0.95$s, i.e. at $t = 1.10$s, $t = 1.20$s, and $t = 1.30$s, small differences in the solution can be perceived between the two methods (Fig. 7): for *interPore* the interface sharpening algorithm with no compression is able to capture droplets up to a time of 0.95s. After that, filaments continue to detach from the mushroom but fragment into liquid structures that are not trackable, and these get diffused numerically (Case 1 and Case 2 in Fig. 7 (b)).

3.4.2. Stem symmetry and numerical convection

With the *RCLSFoam* solver, the symmetry of the flow structure is well captured (Fig. 7 (a)). The interface obtained for the central stem is slightly asymmetrical in the case of *interPore* with added numerical compression (Case 3 in Fig. 7 (b)). This may be due to the compression scheme in OpenFoam [11]. As a result, *interPore* does not fully recover the physical solution published in the literature [9, 53] after $t = 0.95$s. On the other hand, the non-physical wiggles usually seen with VoF solvers [54] in the neck of the stem close to the mushroom-shaped structure have disappeared for all *interPore* runs, with compression factors $C = 0$ and with $C = 1$ (Fig. 7 (b)).

3.4.3. Rate of penetration of the heavy phase into the light phase

For both solvers, the dense phase penetrates into the light phase at a rate that is in accordance with the simulations of Popinet and Zaleski [9]. As ϵ is increased from $0.5\overline{\Delta x}$ to $2\overline{\Delta x}$ in the RCLS formulation, the start of the instability is marginally delayed at $t = 0.7$s, hence the rate of penetration of the spike is reduced overall. On the other hand, the penetration rate does not seem to be altered when varying the numerical parameters of *interPore* (Fig. 7 (b)).

3.4.4. Mushroom-shaped structure development

Fig. 8 (ii) shows the distribution of the velocity field at a number of discrete time intervals, along with the corresponding interface contour (Fig. 8 (i)). The spike itself has a positive velocity downwards, consistent with the direction of acceleration it experiences from the pressure gradient. There is a stagnation point on the lower tip of the spike. Upstream of the mushroom cap, there is a low velocity region. This location is consistent with a high-pressure region. The highest velocities are located on either side of the mushroom cap, extending along and at the tip of the ejected fluid ligaments. The light fluid is thus accelerated faster upwards than the spike and mushroom cap are accelerated downwards. There is evidence of interfacial shearing instabilities (Fig. 8 (iii)) associated with the formation of a mushroom-shaped structure. It is believed [7] that the Kelvin–Helmholtz

instability is the reason for the development of the mushroom cap at the tip of the spike. It is characterised by the development of structures on the spike, arising mainly due to the velocity shear between the two fluid layers. Once the mushroom cap is formed, the effect of drag forces on the spike is increased.

As ϵ is increased from $0.5\overline{\Delta x}$ to $2\overline{\Delta x}$ in the RCLS numerical formulation, the mushroom cap becomes less open (Fig. 7 (a)). With the modified VoF method *interPore*, a change in the sharpening coefficient does not seem to alter the openness of the mushroom cap, but adding numerical compression does make the cap more open (Fig. 7 (b) (iii)). At $t = 0.70$s, the roll-up of the edges of the mushroom cap under the shearing instability is predicted similarly by both solvers (Fig. 8 (iii)). At $t = 0.85$s however, droplets have already detached from the two ligaments in the *interPore* solution (Fig. 8 (b) (i)), causing a noticeable difference in the development of the shearing instability compared to the RCLS formulation. In the latter, the mushroom cap is more open and at the point of droplet separation from the two ligaments a strong upwards velocity is visible (Fig. 8 (a) (iii)).

3.4.5. Ligament breakup into droplets

For all the runs with the RCLS method, at $t = 0.95$s, the filaments elongated on either side of the mushroom have not yet fragmented into droplets, in accordance with Popinet and Zaleski [9]. This is true also for Case 1 and Case 3 simulated with the *interPore* solver (Fig. 7 (b) (i and iii)). However, in *interPore* Case 2, at $t = 0.95$s, the two droplets have already detached from the end of the filaments (see Fig. Fig. 8 (b) (i)). This may be due to the extra sharpening employed in this solver in Case 2.

3.4.6. Mesh refinement to 128×512 cells

Fig. 9 presents the volume fractions and interface predicted by *RCLSFoam* and *interPore* on a refined mesh of 128×512 cells. The RCLS method with $\epsilon = 0.5\overline{\Delta x}$ is not particularly affected by mesh refinement: the interface is sharper but just as many liquid structures are resolved as with the previous coarser mesh (Fig. 9 (a)). Hence it is possible to run the RCLS numerical scheme on coarser meshes without loss of physical accuracy, further highlighting the relevance of this capability. On the other hand, mesh refinement does slightly improve the performance of *interPore*, with extra droplet capture at later times (see Case 5 in Fig. 9(b) versus Case 2 in Fig. 7(b)(ii)). This behaviour is expected. While LS methods provide interface sharpness inherently, VoF methods require addtional treatment to limit diffusion, and mesh refinement acts as an additional sharpening treatment. When compression is added (Case 6 in Fig. 9(b)), ligaments and ligament breakup are well resolved, however, numerical wiggles start forming along the stem and mushroom cap. There appears to be an optimal numerical set-up for the *interPore* solver, in which the traditional OpenFoam numerical compression scheme is not used, but in which the interface sharpening algorithm provides good physical accuracy and resolution of the interface (Case 2 and Case 5), similarly to the conservative level-set method with $\epsilon = 0.5\overline{\Delta x}$ (Case 1 and Case 4).

3.5. Quantitative Results and Discussion

The growth of a RTI can be described using a number of stages [7, 55] as follows. When the perturbation amplitudes η are small compared to their wavelength λ, the early stages in the growth of the instability can be analysed using the linearised equations of motion. The result is that small initial amplitude perturbations of wavelength λ increase in magnitude, exponentially with time [7]. Substantial deviations from the linear
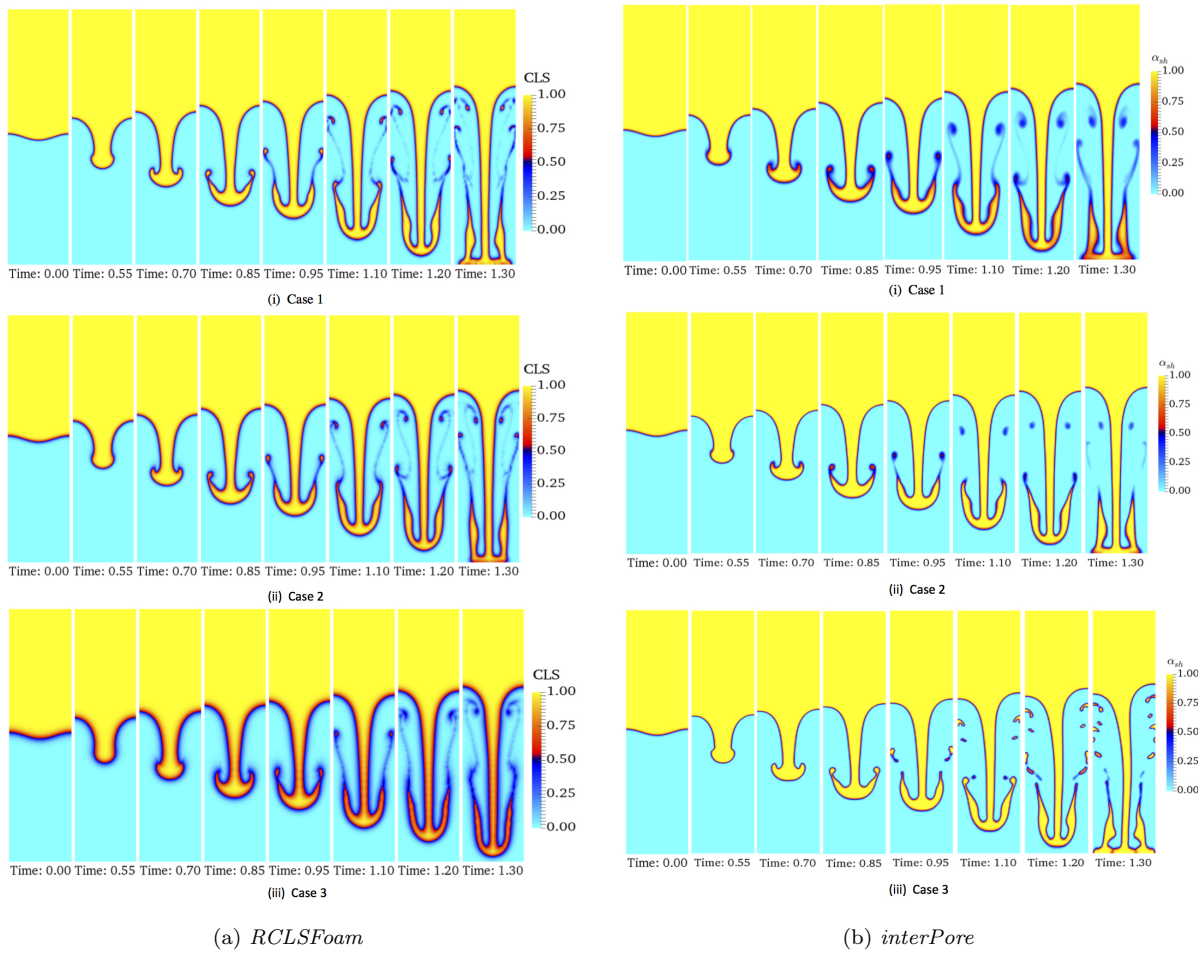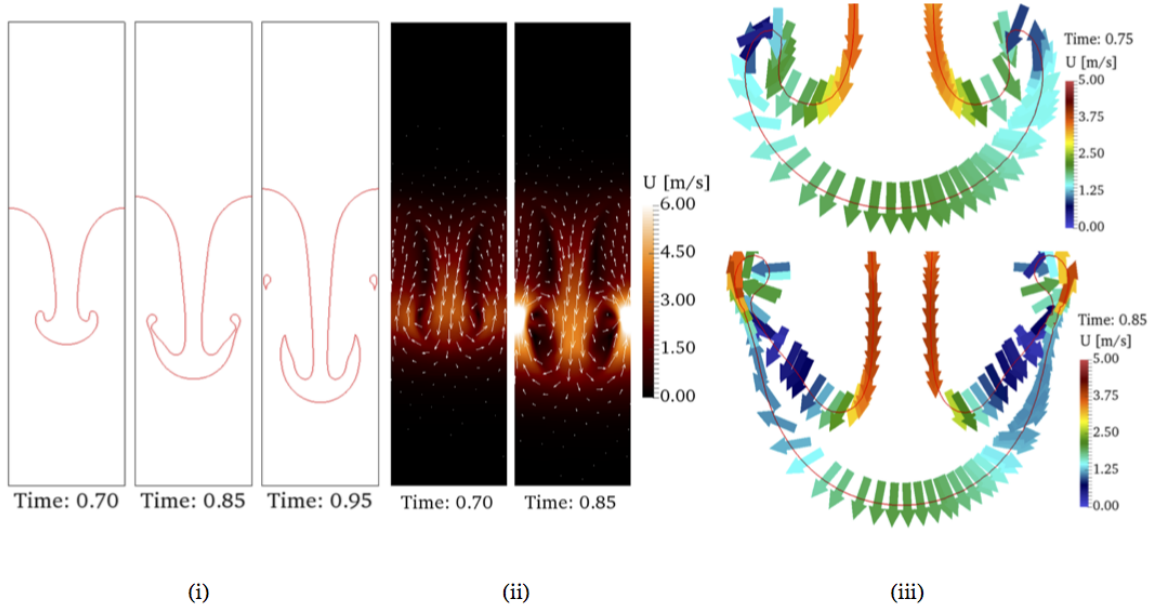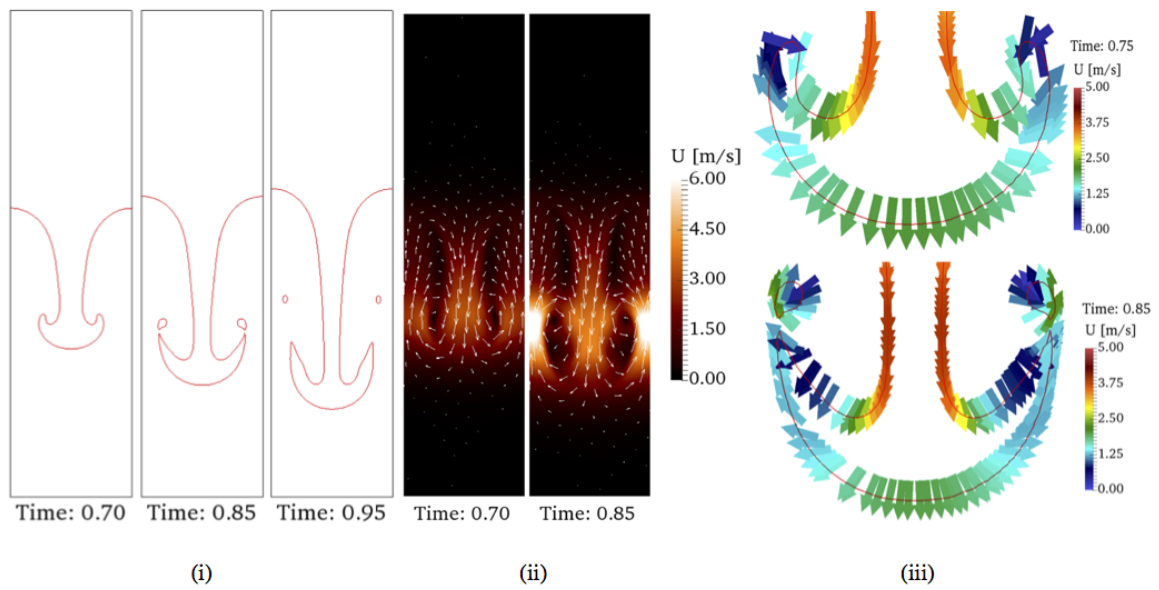
Figure 7: Volume fractions predicted by *RCLSFoam* and *interPore* for the Rayleigh–Taylor instability with 60×224 mesh elements.
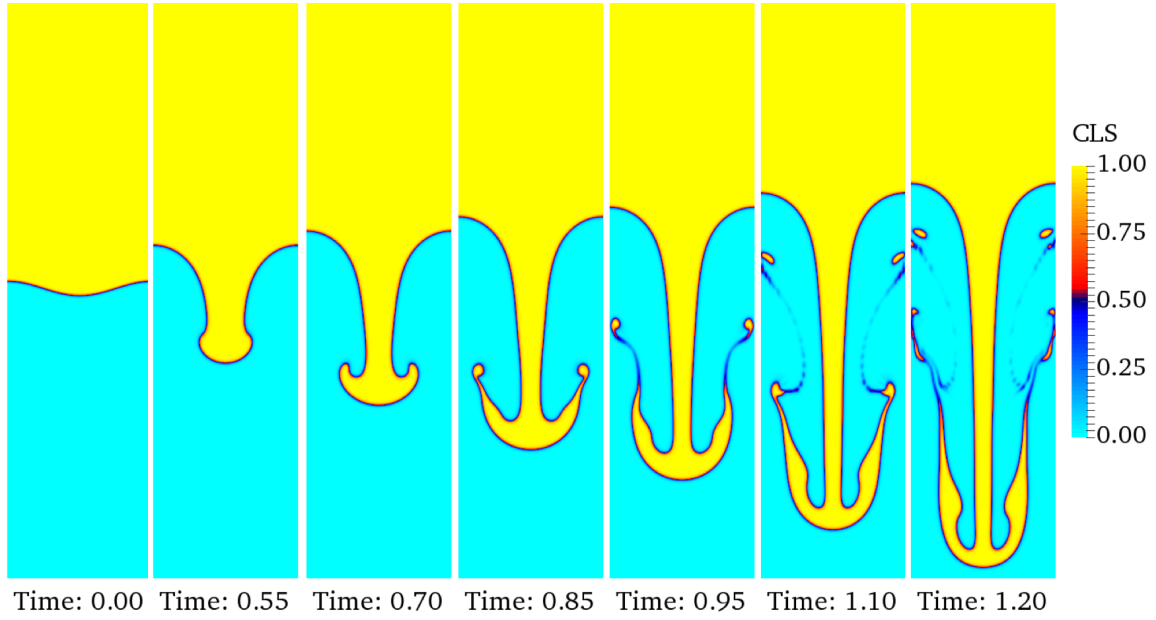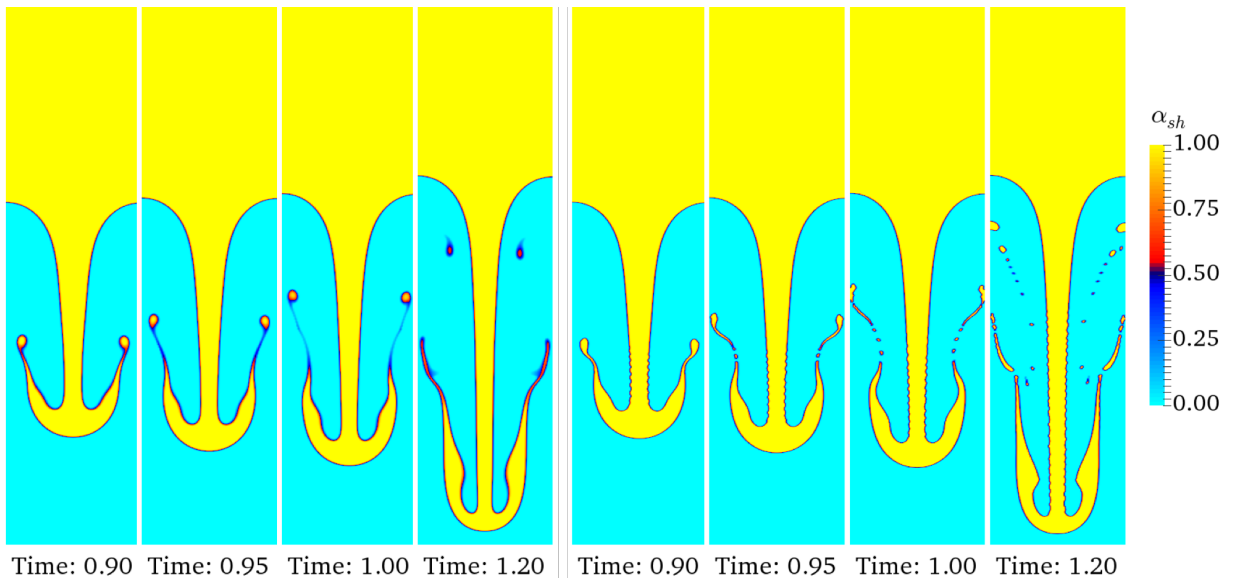
(a) *RCLSFoam*, Case 1

(b) *interPore*, Case 2

Figure 8: Interface (0.5 iso-contour of both indicator functions) predicted by both solvers for the Rayleigh–Taylor instability with 60×224 mesh elements, velocity field and velocity vectors overlaying the interface contour.

(a) *RCLSFoam*, Case 4

(b) *interPore*, Case 5 (*left*) and Case 6 (*right*)

Figure 9: Volume fractions predicted by both solvers for the Rayleigh–Taylor instability with 128×512 mesh elements.

theory are observed when non-linear effects begin to appear. Their development is strongly influenced by three-dimensional effects such as the formation of well-known structures: spikes, bubbles and mushrooms. At later times, spikes and bubbles develop their own mushroom-shaped structures at their tips, which eventually interact and merge. At this stage, non-linear effects can no longer be ignored. The system enters a regime of turbulent or chaotic mixing of the two fluids. In order to describe the aforementioned stages a more quantitative description of the RTI is presented through the amplitude growth rate and the volume fraction profiles.

3.5.1. Amplitude growth rate

The initial growth rate of small amplitude perturbations can be affected by various physical properties including viscosity and surface tension. In linear theory, viscosity tends to reduce the growth rate appreciably, whereas surface tension stabilises wavelengths shorter than a critical wavelength.

The first stability analyses of Rayleigh [50] and Taylor [51] on RTI have been extended in various directions to include additional physical effects such as, for instance, surface tension and viscosity [7]. The work of Chandrasekhar [56, Chapter 10] sought to extend the inviscid linear theory and has led to a rather complete understanding of RTI in incompressible viscous fluids, including the effects of surface tension. At time t, a fluid element of cross section $dydx$ displaced at a distance $\eta(t)$ below $y = 0$, feels a downwards force due to gravity and a downward force from surface tension (Eq. 3). The fluid element net downward acceleration in the $-y$ direction is a combination of the downward gravity force and the downward surface tension force:

$$g\frac{\rho_h - \rho_l}{\rho_h + \rho_l} - \frac{\sigma}{\rho_h + \rho_l}k^2. \tag{26}$$

The simple harmonic solution is given by:

$$\ddot{\eta} - \alpha^2(k)\eta = 0 \tag{27}$$

where

$$\alpha^2(k) = gk\frac{\rho_h - \rho_l}{\rho_h + \rho_l} - \frac{\sigma}{\rho_h + \rho_l}k^3. \tag{28}$$

Eq. 28 is the well-known dispersion equation for interfacial waves of wavenumber k when viscosity is neglected. As expected the interface is now unstable, and the solution to Eq. 28 for fluids initially at rest is:

$$\eta(t) = \eta(0)cosh(\alpha t) \tag{29}$$

where $\eta(0)$ is the perturbation amplitude at $t = 0$. The interfacial wave amplitude grows like $e^{\alpha t}$. The description of the instability is, of course, valid only so long as the amplitude remains small. It is evident from Eq. 28 that surface tension can prevent the instability for sufficiently small wavelengths.

Chandrasekhar also applied normal mode analysis to the linearised Navier–Stokes equations to derive an implicit fourth-order ODE for the amplitude of the y-velocity component as a viscous eigenvalue problem that needs to be solved numerically.

Fig. 10 shows the results of the linear stability analysis discussed above as well as some of the simulation results. The numerical solutions reproduce accurately the theoretical prediction of Chandrasekhar [56] at early times, when the system is still behaving linearly. *RCLSFoam* (Case 1) and *interPore* (Case 2) perform equally well.
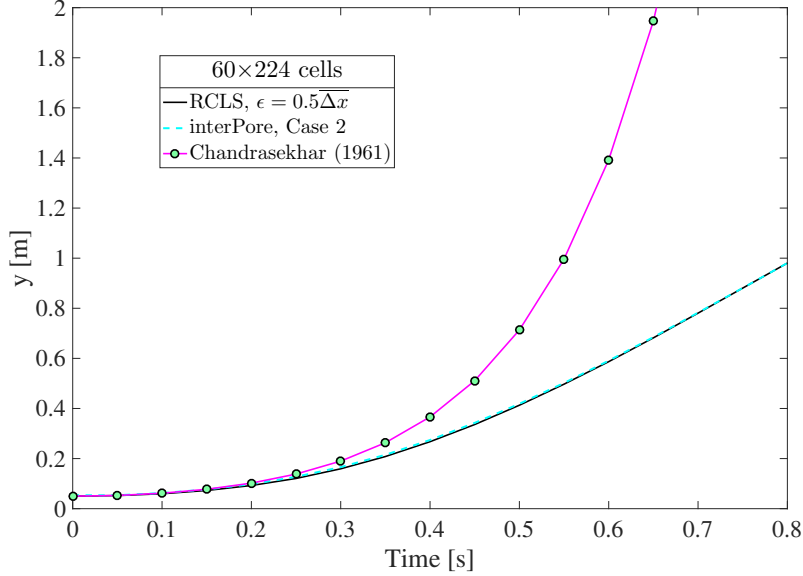
Figure 10: Amplitude growth for the Rayleigh–Taylor instability in the 60×224 computational domain compared to the theoretically predicted growth rates of *RCLSFoam* (Case1) and *interPore* (Case 2).

3.5.2. Volume fraction profiles

Fig. 11 shows volume fraction profiles along the y-direction centreline of the computational domain at two different times: $t = 0$s and $t = 0.2$s and for two different mesh resolutions. As ϵ is increased from $0.5\overline{\Delta x}$ to $2\overline{\Delta x}$ in the RCLS method, at $t = 0$s, the ψ field is initially smeared over a chosen number of cells. The number of cells within the interface thickness remains constant during the simulations. A change in mesh resolution does not change the diffusion of the interfacial thickness. In the modified VoF method with interface sharpening however, the initialised interfacial jump happens over a single computational cell. The later development of the interface thickness over the course of a simulation is affected by the order of accuracy of the numerical scheme and the mesh resolution employed.

At $t = 0.2$s, and for the 60×224 mesh size, both solvers marginally under-predict the theoretical amplitude, at a liquid phase fraction of 0.5. As Chandrasekhar's linear theory does not take into account viscosity, the authors postulate that this is the reason for the under-prediction of the perturbation amplitude compared to the estimated theoretical value (grey vertical line in Figure 11 *bottom-left*). The *interPore* solver shows a lower sensitivity to the numerical parameters (i.e. numerically-added sharpening and/or filtering). The *interPore* solver with high sharpening or compression (Case 2 and Case 3) predicts the same perturbation amplitude as the base case (Case 1) but the profiles are steeper. Over the range of interface parameters ϵ tested for the RCLS method, as ϵ is increased the amplitude at the 0.5 iso-contour diminishes in magnitude, i.e. departs from the theoretical value, in accordance with a less sharply defined interface location. Indeed, the interface needs to have a minimal thickness, so that the gradient of ϕ and the interface normal are accurately calculated.

At $t = 0.2$s, and for the 128×512 mesh size, the 0.5 volume fraction iso-contour of the perturbation amplitude is slightly under-predicted for both solvers, probably for the same reason described in the paragraph above. For the VoF solver with interface sharpening *interPore*, a refined mesh does not change the calculated amplitude of the perturbation. *RCLSFoam* predicts an amplitude very slightly closer to the theoretical value.
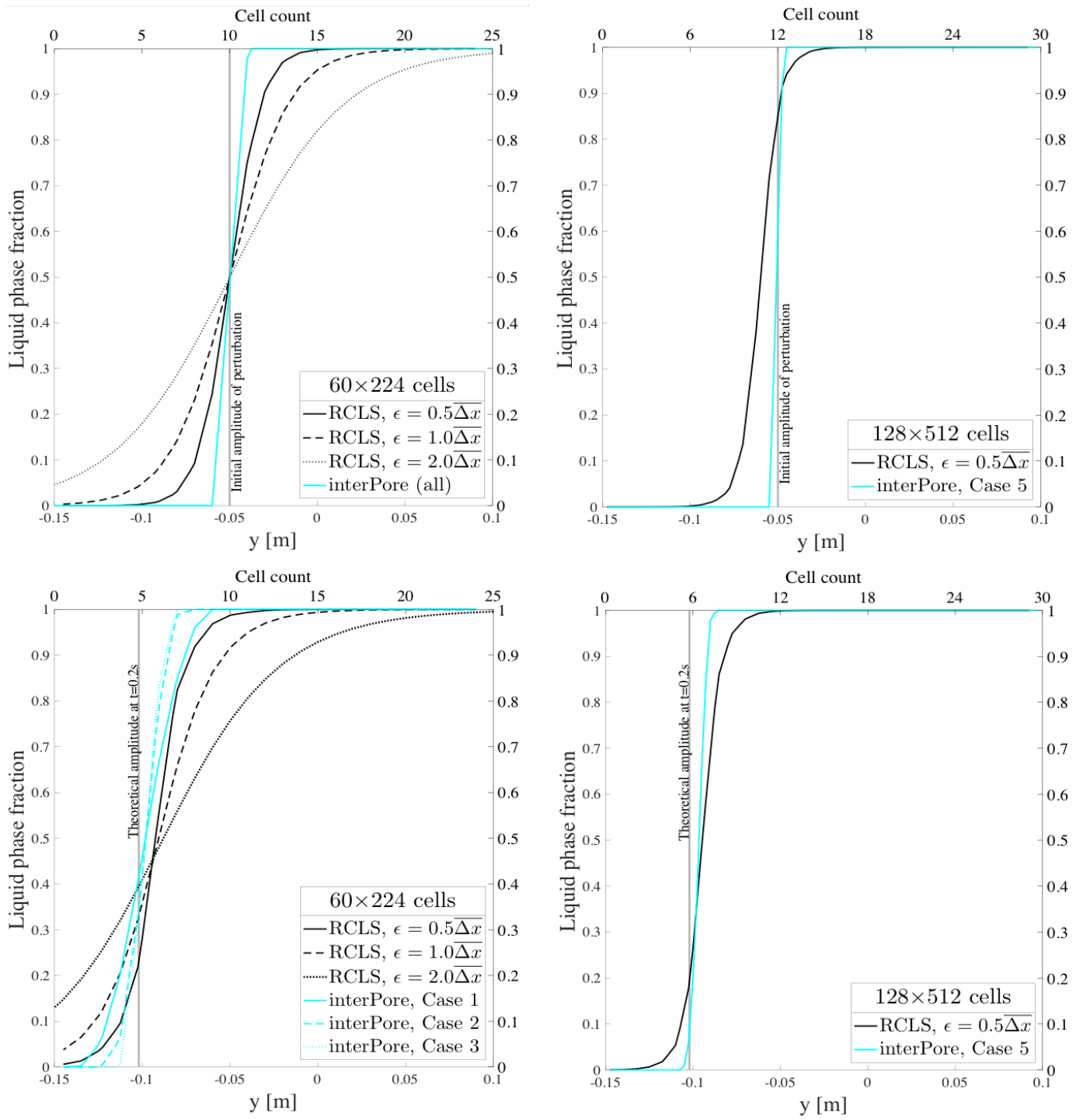
25

Figure 11: Indicator functions smeared over cell count and y-location at $t = 0$ (*top row*) and $t = 0.2$s (*bottom row*) for two mesh sizes: 60×224 and 128×512 mesh elements.
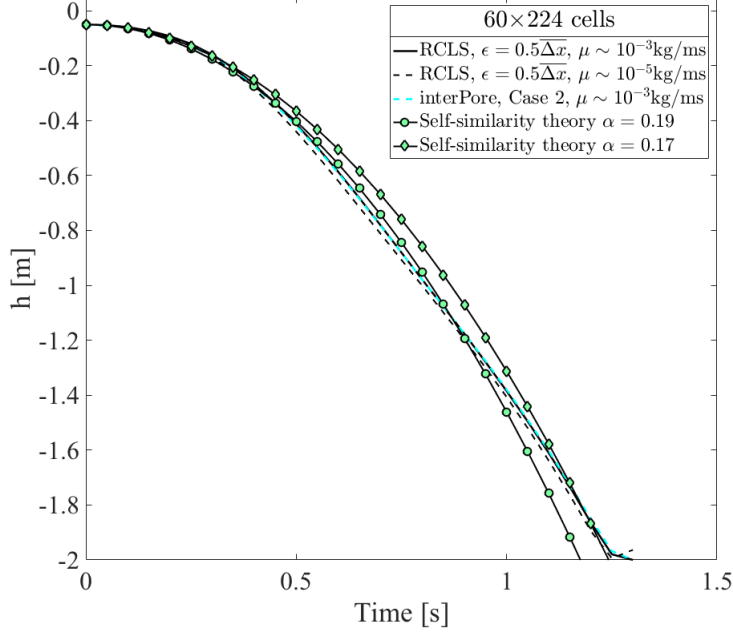
Figure 12: Theoretically predicted self-similar solution for the Rayleigh–Taylor instability against our numerically predicted solutions.

3.6. Self-similarity solution

The RTI marks only the onset of a complex interpenetration process, leading ultimately to the growth of mixing regions between neighbouring fluids [57]. At later times, once the instability has become fully nonlinear, it is difficult to make quantitative comparisons. However, the flow may enter a self-similar growth phase [58]. In the present case, self-similarity can be described by the following equation:

$$\frac{dh}{dt} = 2\sqrt{\alpha Agh} \tag{30}$$

where h is the height of the mixing region and α is a dimensionless growth parameter [59]. The parameter α is the subject of extensive research [60, 59, 61]. The physically realisable solution to Eq. 30 is

$$h(t) = h(0) + \sqrt{\alpha Agh(0)}t + \alpha Agt^2 \tag{31}$$

taking $t = 0$ as the moment in time when the flow first achieves self-similarity. Then $h(0)$ corresponds to the thickness of the mixing region at that instant. The last term dominates the right-hand side of Eq. 31 and hence the mixing thickness becomes:

$$h(t) \sim \alpha Agt^2. \tag{32}$$

Fig. 12 shows the numerically calculated mixing height versus time, as well as the predicted mixing height (Eq. 32) for two different numerical values of the parameter α. We chose here to compare what resulted through our previous results in the optimised runs for each solver in terms of numerical parameters (Table 3). For $t = 0.4 - 0.8$s, both solvers present a similar behaviour, slightly under-predicting the two theoretical curves, while for $t \geq 0.9$s, our numerical predictions fall in-between the two theoretical curves. It is argued that the system becomes self-similar from the very early stages of the instability. A numerical value of α between 0.19

and 0.17 appears to match very well with the numerical results. As viscosity is not taken into account by the model [59], an additional case was run with *RCLSFoam* in which the dynamic viscosity was reduced artificially by two orders of magnitude. This reduction in viscosity does not result in a change of the amplitude growth of the instability (Fig. 12).

Table 4: Advantages and disadvantages of *interPore* and *RCLSFoam* on the current test cases

Approach	Stationary droplet (low Ca)	RT instability (high Ca)
interPore	– low spurious currents – rapid convergence – low sensitivity to solver parameters	– acceptable solutions obtained for a suitable choice of solver parameters
RCLSFoam	– higher spurious currents – reasonable convergence and acceptable solution for suitable choice of solver parameters	– small-scale features well captured – high resistance to numerical wiggles – good symmetry of solution – good representation of droplet pinch-off

4. Conclusions

Two different numerical methods for the solution of two-phase flow problems have been compared in detail on two different test cases, chosen specifically to highlight the strengths and weaknesses of each method (summarised in Table 4). The work was intended to overcome a natural tendency to evaluate numerical methods using only test cases close to the specific application for which they were designed in the first place. In the present context, methods based on achieving optimum interface sharpness at higher Ca numbers were believed to be more prone to spurious currents at low Ca numbers. By the same token, methods that are more resistant to interface artefacts may well be less accurate in terms of sharpness.

In this study, the conservative LS method implemented in the solver *RCLSFoam* and the VoF method implemented in the solver *interPore* have both demonstrated their ability to capture the phase interface and to achieve a good representation of surface tension effects, for suitable settings of the operating parameters within each method. In a static droplet test case, both methods are able to minimise the development of spurious currents, with the *interPore* solver showing significantly smaller errors and with the *RCLSFoam* solver showing a greater sensitivity to operating parameters. In a Rayleigh–Taylor test case, both methods are able to capture the development of the instability, with the *RCLSFoam* solver showing a slightly better ability to capture droplet pinch-off. Both methods compare favourably with the standard *interFoam* solver on both test cases. The results indicate that a well-tuned implementation of either the LS or the VoF method is able to work well across a broad range of relevant conditions, and that the underlying techniques for interface capture have many features in common. Future work will take advantage of those common features in order to move towards a more unified approach for the high-fidelity simulation of two-phase flows. Further testing will involve other well-known test cases such as the spiral in a deformation field or the Zalesak slotted disk, as tested already using the RCLS method in Pringuey and Cant [42] or droplet spreading on surfaces, as tested in Aboukhedr et al. [29] and Aboukhedr et al. [62]. Over time, the aim is to develop the capability to simulate two-phase flow problems regardless of the test case conditions and without the need to adjust solver parameters.

5. Acknowledgements

6. References

1. Shinjo J, Umemura A. Detailed simulation of primary atomization mechanisms in diesel jet sprays (isolated identification of liquid jet tip effects). *Proceedings of the Combustion Institute* 2011;33(2):2089–97.

2. Blunt M, King MJ, Scher H. Simulation and theory of two-phase flow in porous media. *Physical Review A* 1992;46:7680–99.

3. Tryggvason G. Numerical simulations of the Rayleigh-Taylor instability. *Journal of Computational Physics* 1988;75(2):253–82.

4. Francois MM, Cummins SJ, Dendy ED, Kothe DB, Sicilian JM, Williams MW. A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework. *Journal of Computational Physics* 2006;213(1):141–73.

5. Tryggvason G, Esmaeeli A, Lu J, Biswas S. Direct numerical simulations of gas/liquid multiphase flows. *Fluid Dynamics Research* 2006;38(9):660–81.

6. Renardy Y, Renardy M. Prost: A parabolic reconstruction of surface tension for the volume-of-fluid method. *Journal of Computational Physics* 2002;183(2):400–21.

7. Sharp DH. An overview of Rayleigh–Taylor instability. *Physica D: Nonlinear Phenomena* 1984;12:3.

8. Brackbill J, Kothe D, Zemach C. A continuum method for modeling surface tension. *Journal of Computational Physics* 1992;100:335–54.

9. Popinet S, Zaleski S. A front-tracking algorithm for accurate representation of surface tension. *International Journal for Numerical Methods in Fluids* 1999;30(6):775–93.

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

11. Weller HG, Tabor G, Jasak H, Fureby C. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Journal of Computational Physics* 1998;12:620–31.

12. Popinet S. Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *Journal of Computational Physics* 2003;190(2):572–600.

13. Sethian JA, Smereka P. Level set methods for fluid interfaces. *Annual Review of Fluid Mechanics* 2003;25:341–72.

14. Scardovelli R, Zaleski S. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics* 1999;31:567–603.

15. Anderson DM, McFadden GB, Wheeler AA. Diffuse-interface methods in fluid mechanics. *Annual Review of Fluid Mechanics* 1998;30:139–65.

16. Hirt CW, Nichols BD. Volume of Fluid (VoF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981;39(1):201–25.

17. Ubbink O, Issa R. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics* 1999;153(1):26–50.

18. Roenby J, Bredmose H, Jasak H. A computational method for sharp interface advection. *arXiv preprint arXiv:160105392* 2016;.

19. Muzaferija S, Peric M. Computation of free-surface flows using interface-tracking and interface-capturing methods. *Advanced Fluid Mechanics* 1999;24:59–100.

20. Jasak H, Weller HG. Interface-tracking capabilities of the InterGamma differencing scheme. Tech. Rep.; Imperial College London; 1995.

21. Osher S, Sethian JA. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics* 1988;79:12–49.

22. Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics* 1994;114(1):146–59.

23. Losasso F, Fedkiw R, Osher S. Spatially adaptive techniques for level set methods and incompressible flow. *Computers & Fluids* 2006;35(10):995–1010.

24. Chopp DL. Computing minimal surfaces via level set curvature flow. *Journal of Computational Physics* 1993;106(1):77–91.

25. Russo G, Smereka P. A remark on computing distance functions. *Journal of Computational Physics* 2000;163(1):51–67.

26. Lafaurie B, Nardone C, Scardovelli R, Zaleski S, Zanetti G. Modelling merging and fragmentation in multiphase flows with surfer. *Journal of Computational Physics* 1994;113(1):134–47.

27. Rudman M. Volume-tracking methods for interfacial flow calculations. *International Journal for Numerical Methods in Fluids* 1997;24(7):671–91.

28. Raessi M, Pitsch H. Modeling interfacial flows characterized by large density ratios with the level set method. *Center for Turbulence Research Annual Research Briefs* 2009;:159–69.

29. Aboukhedr M, Gavaises M, Georgoulas A, Marengo M, Vogiatzaki K. Numerical investigation of droplet spreading on porous and non-porous surfaces. In: *ILASS Europe – Proceedings of the 23rd Annual Conference on Liquid Atomization and Spray Systems*. Brighton, UK; 2016:.

30. Sussman M, Puckett EG. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of Computational Physics* 2000;162:301–37.

31. Ménard T, Beau PA, Tanguy S, Berlemont A. Primary break-up: DNS of liquid jet to improve atomization modelling. *Computational Methods in Multiphase Flow* 2005;3:343–53.

32. Ménard T, Tanguy S, Berlemont A. Coupling Level Set/VOF/ghost fluid methods: Validation and application to 3D simulation of the primary break-up of a liquid jet. *International Journal of Multiphase Flow* 2007;33:510–24.

33. Yokoi K. A practical numerical framework for free surface flows based on CLSVOF method, multi-moment methods and density-scaled CSF model: Numerical simulations of droplet splashing. *Journal of Computational Physics* 2013;232(1):252–71.

34. Arienti M, Sussman M. An embedded level set method for sharp-interface multiphase simulations of diesel injectors. *International Journal of Multiphase Flow* 2014;59:1–14.

35. Ferrari A, Magnini M, Thome JR. A Flexible Coupled Level Set and Volume of Fluid (flexCLV) method to simulate microscale two-phase flow in non-uniform and unstructured meshes. *International Journal of Multiphase Flow* 2017;.

36. Dianat M, Skarysz M, Garmory A. A Coupled Level Set and Volume of Fluid method for automotive exterior water management applications. *International Journal of Multiphase Flow* 2017;91:19–38.

37. Jasak H. OpenFOAM: Open source CFD in research and industry. *International Journal of Naval Architecture and Ocean Engineering* 2009;1(2):89–94.

38. Deshpande SS, Anumolu L, Trujillo MF. Evaluating the performance of the two-phase flow solver interFoam. *Computational Science & Discovery* 2012;5(1):014016.

39. Issa R. Solution of the implicitly discretised fluid flow equations by Operator-Splitting. *Journal of Computational Physics* 1985;62:40–65.

40. Olsson E, Kreiss G. A conservative level set method for two phase flow. *Journal of Computational Physics* 2005;210:225–46.

41. Olsson E, Kreiss G, Zahedi S. A conservative level set method for two phase flow II. *Journal of Computational Physics* 2007;225:785–807.

42. Pringuey T, Cant RS. Robust Conservative Level Set method for 3D Mixed-Element Meshes – Application to LES of primary liquid-sheet breakup. *Communications in Computational Physics* 2014;16:403–39.

43. Pringuey T, Cant RS. High Order Schemes on Three-Dimensional General Polyhedral Meshes - Application to the Level Set Method. *Communications in Computational Physics* 2012;12:1–41.

44. Desjardins O. Numerical methods for liquid atomization and application in detailed simulations of a diesel jet. Ph.D. thesis; Stanford University; 2008.

45. Desjardins O, Moureau V, Knudsen E, Herrmann M, Pitsch H. Conservative Level Set/ghost fluid method for simulating primary atomization. In: *ILASS Americas – Proceedings of the 20th Annual Conference of the Institute for Liquid Atomization and Spray Systems*. Chicago, IL; 2007:.

46. Georgoulas A, Koukouvinis P, Gavaises M, Marengo M. Numerical investigation of quasi-static bubble growth and detachment from submerged orifices in isothermal liquid pools: The effect of varying fluid properties and gravity levels. *International Journal of Multiphase Flow* 2015;74:59 – 78.

47. Raeini AQ, Blunt MJ, Bijeljic B. Modelling two-phase flow in porous media at the pore scale using the volume-of-fluid method. *Journal of Computational Physics* 2012;231(17):5653–68.

48. Harvie DJ, Davidson M, Rudman M. An analysis of parasitic current generation in volume of fluid simulations. *Applied Mathematical Modelling* 2006;30(10):1056–66.

49. Williams M, Kothe D, Puckett E. Accuracy and convergence of continuum surface tension models. *Fluid Dynamics at Interfaces, Cambridge University Press, Cambridge* 1998;:294–305.

50. Rayleigh JWS. Investigation of the character of the equilibrium of an incompressible heavy fluid of variable density. *Proceedings of the Royal Society of London* 1883;14:170–7.

51. Taylor GI. The instability of liquid surfaces when accelerated in a direction perpendicular to their planes. *Proceedings of the Royal Society* 1950;201:192.

52. Yokoi K. A numerical method for free-surface flows and its application to droplet impact on a thin liquid layer. *Journal of Scientific Computing* 2008;35(2-3):372–96.

53. Puckett EG, Almgren AS, Bell JB, Marcus DL, Rider WJ. A High-Order Projection Method for Tracking Fluid Interfaces in Variable Density Incompressible Flows. *Journal of Computational Physics* 1997;130:269–82.

54. Gopala VR, van Wachem BGM. Volume of fluid methods for immiscible-fluid and free-surface flows. *Chemical Engineering Journal* 2008;141:204–21.

55. Lamb SH. Hydrodynamics. Cambridge University Press; 1895.

56. Chandrasekhar S. Hydrodynamic and Hydromagnetic Stability. Oxford University Press; 1961.

57. Kull HJ. Theory of the Rayleigh–Taylor instability. *Physics Reports* 1991;206:197–325.

58. Fermi E, von Neumann J. Taylor instability of incompressible liquids. Part 1. Taylor instability of an incompressible liquid. Part 2. Taylor instability at the boundary of two incompressible liquids. Tech. Rep.; Los Alamos Scientific Laboratory; 1953.

59. Ristorcelli JR, Clark TT. Rayleigh–Taylor turbulence: self-similar analysis and direct numerical simulations. *Journal of Fluid Mechanics* 2004;507:213–53.

60. Dimonte G, Youngs DL, Dimits A, Weber S, Marinak M, Wunsch S, Garasi C, Robinson A, Andrews MJ, Ramaprabhu P, Calder AC, Fryxell B, Biello J, Dursi L, MacNeice P, Olson K, Ricker P, Rosner R, Timmes F, Tufo H, Young YN, Zingale M. Density ratio dependence of Rayleigh–Taylor mixing for sustained and impulsive acceleration histories. *Physics of Fluids* 2004;16(5):1668–93.

61. Kadau K, Germann TC, Hadjiconstantinou NG, Lomdahl PS, Dimonte G, Holian BL, Alder BJ. Nanohydrodynamics simulations: An atomistic view of the Rayleigh–Taylor instability. *Proceedings of the National Academy of Sciences of the United States of America* 2004;101(16):5851–5.

62. Aboukhedr M, Mitroglou N, Georgoulas A, Marengo M, Vogiatzaki K. Simulation of droplet spreading on micro-CT reconstructed 3D real porous media using the volume-of-fluid method. In: *ILASS Europe – Proceedings of the 28th Annual Conference on Liquid Atomization and Spray Systems*. Valencia, Spain; 2017:Accepted for publication.

790