



City Research Online

City St George's, University of London

Citation: Aboukhedr, M., Georgoulas, A., Marengo, M., Gavaises, M. & Vogiatzaki, K. (2018). Simulation of micro-flow dynamics at low capillary numbers using adaptive interface compression. *Computers & Fluids*, 165, pp. 13-32. doi: 10.1016/j.compfluid.2018.01.009

This is the accepted version of the paper.

This version of the publication may differ from the final published version. To cite this item please consult the publisher's version.

Permanent repository link: <https://openaccess.city.ac.uk/id/eprint/19881/>

Link to published version: <https://doi.org/10.1016/j.compfluid.2018.01.009>

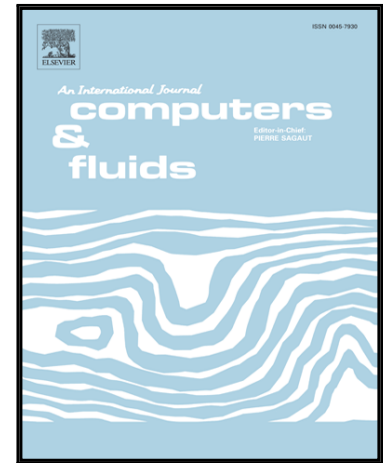
Copyright and Reuse: Copyright and Moral Rights remain with the author(s) and/or copyright holders. Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge, unless otherwise indicated, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way. For full details of reuse please refer to [City Research Online policy](#).

Accepted Manuscript

Simulation of micro-flow dynamics at low capillary numbers using adaptive interface compression

M. Aboukhedr, A. Georgoulas, M. Marengo, M. Gavaises, K. Vogiatzaki

PII: S0045-7930(18)30009-4
DOI: [10.1016/j.compfluid.2018.01.009](https://doi.org/10.1016/j.compfluid.2018.01.009)
Reference: CAF 3692



To appear in: *Computers and Fluids*

Received date: 12 July 2017
Revised date: 15 November 2017
Accepted date: 13 January 2018

Please cite this article as: M. Aboukhedr, A. Georgoulas, M. Marengo, M. Gavaises, K. Vogiatzaki, Simulation of micro-flow dynamics at low capillary numbers using adaptive interface compression, *Computers and Fluids* (2018), doi: [10.1016/j.compfluid.2018.01.009](https://doi.org/10.1016/j.compfluid.2018.01.009)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

1 Highlights

- 2 • Multiphase flow solver using adaptive compression scheme has been introduced.
- 3 • Wide range of conditions using well-established benchmark cases has been tested.
- 4 • The adaptive compression facilitates simulating flows at low capillary numbers.
- 5 • The adaptive nature of the coef. counter balances the need for very fine grids.
- 6 • Using the mentioned method gives accurate results in estimating bubble formation.

ACCEPTED MANUSCRIPT

Simulation of micro-flow dynamics at low capillary numbers using adaptive interface compression [☆]

M. Aboukhdr^{a,*}, A. Georgoulas^b, M. Marengo^b, M. Gavaises^a, K. Vogiatzaki^b

^aDepartment of Mechanical Engineering, City, University of London, UK

^bSchool of Computing, Engineering and Mathematics, Advanced Engineering Centre, University of Brighton, Brighton, UK

Abstract

A numerical framework for modelling micro-scale multiphase flows with sharp interfaces has been developed. The suggested methodology is targeting the efficient and yet rigorous simulation of **complex interface motion at capillary dominated flows (low capillary number)**. Such flows are encountered in various configurations ranging from micro-devices to naturally occurring porous media. The methodology uses as a basis the Volume-of-Fluid (VoF) method combined with additional sharpening smoothing and filtering algorithms for the interface capturing. These algorithms help the minimisation of the parasitic currents present in flow simulations, when viscous forces and surface tension dominate inertial forces, like in porous media. The framework is implemented within a finite volume code (OpenFOAM) using a limited Multi-dimensional Universal Limiter with Explicit Solution (MULES) implicit formulation, which allows larger time steps at low capillary numbers to be utilised. **In addition, an adaptive interface compression scheme is introduced for the first time in order to allow for a dynamic estimation of the compressive velocity only at the areas of interest and thus has the advantage of avoiding the use of a-priori defined parameters. The adaptive method is found to increase the numerical** accuracy and to reduce the sensitivity of the methodology to tuning parameters. The accuracy and stability of the proposed model is verified against five different benchmark test cases. Moreover, numerical results are compared against analytical solutions as well as available experimental data, which reveal improved solutions relative to the standard VoF solver.

Keywords: CFD, interFoam, two-phase flows, microfluidics, surface tension forces, parasitic currents, micro-scale modelling

[☆]This document is a collaborative effort

*Corresponding author

Email address: mahmoud.aboukhdr.1@city.ac.uk (M. Aboukhdr)

List of Nomenclature

u	Velocity
p	Pressure
p_c	Capillary pressure
p_d	Dynamic pressure
f	External forces
f_g	Gravitational forces
f_s	Surface tension force
ρ	Density
μ	Dynamic viscosity
$u_{r,f}$	Relative velocity at cell faces
σ	Surface tension
ϕ_f	Volumetric flux
ϕ_c	Compression volumetric flux
ϕ	Capillary flux
$\phi_{threshold}$	Threshold volumetric flux
V_i	Volume per grid cell
S_f	Outward-pointing face area
κ	Interface curvature
κ_f	Filtered interface curvature calculated based on smooth function α_{smooth}
$\kappa_{s,i+1}$	Smooth interface curvature calculated based on smooth function κ_f
κ_{final}	Weighted interface curvature calculated based on smooth function $\kappa_{s,i}$
η_s	Normal vector to the interface
δ_s	Dirac delta function
α	Volume fraction
α_{smooth}	Volume fraction using Laplacian formulation
α_{sh}	Sharp inductor function
$C_{compr.}$	Constant interface compression coefficient
C_{adp}	Adaptive interface compression
C_{sh}	Sharpening coefficient
U_f	filtering coefficient
$\langle \eta_s \rangle_f$	Face centred normal vector
$\langle \nabla \alpha \rangle_f$	Volume fraction interpolated from cell centre to face centre
δ_n	Small value

31 **1. Introduction**

32 Flows through "narrow passages" such as micro-channels or pore-scale flows whose dimensions are
33 less than $O(\text{mm})$ and greater than $O(\mu\text{m})$ differ from their macroscopic counterparts at important aspects:
34 the small size of the geometries makes molecular effects such as wall slip or wettability more important,

35 while amplifies the magnitudes of certain ordinary continuum effects associated with strain rate and shear
36 stress. Such flows are present in various natural formations (rocks and human organs) as well as man-made
37 applications (micro-conductors, micro-emulsions, etc.). Thus, microscale physics attracts the interest of
38 various disciplines including cosmetic and pharmaceutical industries as well as biomedical and petroleum
39 engineering. For more details on the application of microscale geometries, the reader is referred to [1].
40 Among all these applications transportation of droplets in microchannels at low Capillary ($Ca = \frac{\mu u}{\sigma}$) num-
41 bers has attracted the interest of researchers from the theoretical and experimental point of view [2, 3, 4].
42 For example, understanding the dynamics of immiscible fluids in micro-devices can facilitate the creation
43 of monodisperse emulsions. Droplets of the same size move with low velocities through microchannel
44 networks and are used as micro-reactors to study very fast chemical kinetics [5]. Another example of low
45 Ca flow dynamics in micro-scale can be seen at trapped oil blobs in porous reservoirs. Understanding the
46 trapping flow dynamics at the pore scale level can be the key to minimising the trapping of a non-wetting
47 phase and enhancing recovery systems of hydrocarbons, [6]. Although a large number of methods has been
48 developed for simulating multiphase flows at macro-scale including the well known Level Sets (LS) [7] and
49 Volume of Fluid (VoF) methods [8], the extension of these methods to micro-scale is not always straightfor-
50 ward. The main weakness of the LS methods is that they do not preserve mass. As a result, poorly resolved
51 regions of the flow are typically susceptible to mass loss behaviour and loss of signed distance property due
52 to advection errors. Various modification have been suggested focusing on solving the conservation issues
53 [9], extending the method to high Reynolds numbers [10] and to unstructured meshes [11, 12]. While using
54 a re-initialization procedure as discussed by [13] is a solution to the mass conservation issue, it increases
55 the computational cost and creates an artificial interface displacement that may affect mass conservation,
56 see the review by Russo and Smereka [14] for details. Similarly the VoF method is based on the numerical
57 solution of a transport equation that distinguishes the two fluids in the domain, and it represents the volume
58 percentage of each fluid phase in each cell over the total volume of the cell. The interface between the two
59 phases is defined in the cells where the VoF function takes a value between (0, 1). In incompressible flows,
60 the mass conservation is achieved by using either a geometrical reconstruction coupled with a geometrical
61 approximation of the volume of fluid advection or a compressive scheme as discussed by Rusche [15] and
62 implemented by Weller et al. [16]. The VoF method has been the most widely used interface capturing

63 method due to ease of implementation as reviewed by Wörner [2].

64 Within the VoF framework two commonly used methods for interface representation exist: (a) a com-
65 pressive method and (b) a geometric method. Both VoF methods are used in order to calculate the discrete
66 volume fraction of each phase within a cell, which is then transported based on the underlying fluid ve-
67 locity. Compressive VoF methods discretise the partial differential equation describing the transport of the
68 volume fraction of each phase using algebraic differencing schemes [17, 18]. The key for the accuracy of
69 these methods is that, in order to keep the interface sharp and without distortion, the temporal and spatial
70 discretisation should be performed using higher order schemes and careful tuning. Otherwise the method
71 may suffer from excessive diffusion of the interface region which also affects the calculation of the interface
72 curvature and the normal interface vectors. Park et al. [19] and Gopala and van Wachem [20] showed the
73 compressive VoF methods capabilities of advecting sharp interface, and they also underlined the difficulties
74 in retaining the shape and sharpness of the interface. Using a geometric method, an explicit representa-
75 tion of the interface is advected, reconstructed from the VoF volume fraction field. The piecewise linear
76 methods so-called (PLIC) is the most developed reconstruction method found in the literature [21, 22]. Ge-
77 ometric methods advect the interface very accurately, but their main drawback is their complexity for 3D
78 applications, in particular when used in conjunction with an unstructured mesh [23].

79 Recently, the coupling between VoF and LS, the so-called Coupled Level Set Volume Of Fluid (CLSVoF)
80 method [24] has also received significant attention since it combines the advantages of both methods, i.e.,
81 the VoF mass conservation and the LS interface sharpness [24, 25]. On the downside, this approach also
82 combines the weaknesses of each method since techniques to keep the VoF interface sharp and reinitialise
83 the distancing function are needed. Based on various published results for both methods [20, 26, 27, 28] the
84 existent frameworks reviewed in the previous paragraph - regardless of the various modifications available
85 - still suffer from their inherent severe drawbacks. These drawbacks are more pronounced in low Ca flows,
86 and, as discussed in detail in Popinet and Zaleski [29], Tryggvason et al. [30] and Bilger et al. [31], stem
87 from the fact that sharp discontinuities such as interfaces are represented by finite volume integrals [8]. The
88 most common issue is that in all implicit interface capturing methods, the interface location is known by
89 defining the normal and the curvature implicitly. For the VoF methods, in particular, which are based on the
90 representation of the discontinuous interface with continuous colour function, the calculation of the proper-

91 ties of each phase is possible, given an accurate numerical scheme for solving the colour function transport
92 equation is available. However, the accuracy of the calculated interface curvature (that is then required
93 for the calculation of the capillary pressure force) depends on determining the derivative of the introduced
94 discontinuous colour function, which is considered to be difficult from a numerical point of view, and may
95 leads to numerical instabilities [32].

96 An additional issue is the generation of non-physical velocities at the interface which are known as
97 "spurious" or "parasitic" currents. The primary sources of spurious currents have been identified as the
98 combination of inaccurate interface curvature and lack of a discrete force balance as discussed by Francois
99 et al. [33]. It should be stressed that the local force imbalance between the capillary pressure and the pressure
100 arising from the normal component of the surface tension force vectors (due to the imprecise evaluation of
101 the local curvature) can create the non-physical velocities, (spurious currents") which are commonly small
102 in absolute values in inertia dominated flows, but become very problematic in capillary dominated flows.

103 Numerical challenges related to the advection of the interface in the context of VoF are well documented
104 by Tryggvason et al. [30]. Intrinsic to the method, regardless if geometric reconstruction or interface com-
105 pression is used, is the numerical diffusion of the interface, which is highly dependent on the mesh size [18].
106 The numerical diffusion can be reduced by using a geometrical reconstruction coupled with a geometrical
107 approximation of the VoF advection as discussed by Roenby et al. [34]. Alternatively, using a compressive
108 algorithm, the convective term of the VoF equation can be discretised using a compressive differencing
109 scheme designed to preserve the interface sharpness. Examples include the HRIC by Muzafferija and Peric
110 [35], or the compressive model available within OpenFoam [16]. Compression schemes do not require any
111 geometrical reconstruction of the interface and extension to three dimensions and unstructured meshes is
112 straightforward. However, compression schemes are not always sufficient to eliminate numerical diffusion
113 completely and additional treatment is needed [36].

114 Various remedies that still have room for development have been suggested, and they can be sum-
115 marised as following: (i) ensuring an accurate balance between local pressure and surface tension gradient.
116 In Francois et al. [33] a cell-centered framework has been introduced. It is demonstrated that this algorithm
117 can achieve an exact balance of between local pressure and surface tension gradient using structured mesh.
118 Moreover, Francois et al. [33] and [37] discussed the origin of spurious currents within the introduced

119 balanced-force flow algorithms, as they highlighted the deficiencies introduced at the interface curvature
120 estimation. (ii) sharp representation of the interface, with accurate curvature estimation and introduction
121 of a so-called "compression velocity" to damp diffusion. Ubbink and Issa [18] introduced the compressive
122 discretisation scheme so-called Compressive Interface Capturing Scheme for Arbitrary Meshes CICSAM
123 that makes a use of the normalised variable diagram concept introduced by Leonard [38]. Popinet [39]
124 generalised a height-function and CSF formulations to an adaptive quad/octree discretisation to allow re-
125 finement along the interface for the case of capillary breakup of a three-dimensional liquid jet. Moreover,
126 [39] discusses the long-standing problem of "parasitic currents" around a stationary droplet in contrast to
127 the recent study of Francois et al. [33], where the issue is shown to be solved by the combination of appro-
128 priate implementations of a balanced-force CSF approach and height-function curvature estimation. (iii)
129 implicit or semi-implicit treatment of surface tension, Denner and van Wachem [40] reviewed the time-step
130 requirements associated with resolving the dynamics of the equations governing capillary waves, to deter-
131 mine whether explicit and implicit treatments of surface tension have different time-step requirements with
132 respect to the (1) dispersion of capillary waves, and (2) the formulation of an accurate time-step criterion for
133 the propagation of capillary waves based on established numerical principles. The fully-coupled numerical
134 framework with implicit coupling of the governing equations and the interface advection, and an implicit
135 treatment of surface tension proposed by [40] was used to study the temporal resolution of capillary waves
136 with explicit and implicit treatment of surface tension.

137 In the present work, a new framework for modelling immiscible two-phase flows for low Ca applications
138 dominated by surface tension is suggested. The standard multiphase flow solver of OpenFOAM 2.3x has
139 been extended to include sharpening and smoothing interface capturing techniques suitable for low Ca
140 numbers flow. In addition a new generalised methodology that utilises an adaptive interface compression is
141 introduced for the first time. **While existing compression** schemes are based on an a priori tuned parameter,
142 which is typically kept constant throughout the simulations, in the present study compression is **activated**
143 **only in areas that the interface is prone to** diffusion and the parameter is thus defined adaptively. This
144 adaptive scheme is proved to limit the interface diffusion and to keep parasitic currents to minimal levels
145 while reducing the computational time. The proposed framework for interface advection aspires to offer
146 better modelling of flows in microscale that up to date have been proven problematic. The paper is structured

147 as following: Initially the numerical framework underlining the modifications suggested over the traditional
 148 VoF methodology in order to achieve better representation of the interface is introduced. The effect of
 149 each parameter used in the proposed framework is then evaluated individually based on a wide range of
 150 benchmark cases. The first test case refers to single and multiple droplet relaxations in a zero velocity field,
 151 aiming to assess the capability of the framework to damp spurious currents using various combination of
 152 control parameter. The evaluation of the solver for an advection test using the Zalesak disk [41] is also
 153 presented followed by results relevant to the motion of circle in a vortex field (Roenby et al. [34], Rider and
 154 Kothe [42]). Finally, a numerical study of the generation of bubbles in a T-junction is studied to evaluate
 155 the introduced framework in **simulating more complex two-phase flows at a low Ca numbers**.

156 2. Numerical method

157 The method presented in this section is implemented within the open source CFD toolkit OpenFOAM
 158 [43]. An incompressible and isothermal two-phase flow with constant phase densities ρ_1 and ρ_2 and vis-
 159 cosities μ_1 and μ_2 is considered. The two phases are treated as one fluid and a single set of equations is
 160 solved in the entire computational domain. The volume fraction, α of each phase within a cell is defined
 161 by an additional transport equation. The formulation for the conservation of mass and momentum for the
 162 phase mixture is given by the following equations:

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

$$\frac{D}{Dt}(\rho \mathbf{u}) = \nabla \cdot \mathbf{T} - \nabla p + \mathbf{f} \quad (2)$$

163 where \mathbf{u} is the fluid velocity, p is the pressure and ρ is the density. The pressure-velocity coupling is
 164 handled using the Pressure-Implicit with Splitting Operators(PISO) method of [44, 45]. The term $\nabla \cdot \mathbf{T} =$
 165 $\nabla \cdot (\mu \nabla \mathbf{u}) + \nabla \mathbf{u} \cdot \nabla \mu$ is the viscous stress tensor. The term $\mathbf{f} = \mathbf{f}_g + \mathbf{f}_s$ corresponds to all the external
 166 forces, i.e. $\mathbf{f}_g = \rho \mathbf{g}$ is the gravitational force and \mathbf{f}_s represents the capillary forces for the case of constant
 167 surface tension coefficient σ . The global properties are weighted averages of the phase properties through
 168 the volume fraction value that is calculated in each cell:

$$\rho = \rho_1 + (\rho_2 - \rho_1)\alpha \quad (3)$$

$$\mu = \mu_1 + (\mu_2 - \mu_1)\alpha \quad (4)$$

169 The sharp interface Γ represents a discontinuous change of the properties of the two fluids. The surface
 170 tension force must balance the jump in the stress tensor along the fluid interface. At each time step, the
 171 dynamics of the interface are determined by the Young-Laplace balance condition as;

$$\Delta P_{exact} = \sigma\kappa \quad (5)$$

172 accounting for a constant surface tension coefficient σ along the interface. The term κ represents the inter-
 173 face curvature. The term on the right-hand side of Eq. 5 is effectively the source term in the Navier–Stokes
 174 equations for the singular capillary force, that is only present at the interface. In the proposed numerical
 175 method, the Continuum Surface Force (CSF) description of Brackbill et al. [8] is used to represent the
 176 surface tension forces in the following form:

$$f_s = \sigma\kappa_{final}\delta_s \quad (6)$$

177 where the term κ_{final} represents the interface curvature at the final stage of smoothing as discussed in section
 178 2.2, δ_s is a delta function defined on the interface, and η_s is the normal vector to the interface α_{smooth} as
 179 discussed in section 2.2 and is calculated by the following equation:

$$\eta_s = \frac{\nabla\alpha_{smooth}}{|\nabla\alpha_{smooth}|} \quad (7)$$

180 The terms δ_s and κ_f are associated with the artificially smoothed and sharpened indicator function fields that
 181 will be discussed in details in the following section. In the VoF method, the indicator function α represents
 182 the volume fraction of one of the fluid phases in each computational cell. The indicator function evolves
 183 spatially and temporally according to an advection transport equation of the following general form:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0 \quad (8)$$

184 Ideally, the interface between the two phases should be massless since it represents a sharp discontinuity.
 185 However, within VoF formulation the numerical diffusion of Eq. 8 results in values of α that vary between
 186 0 and 1.

187 The framework described above reflects the generalised framework of VoF methods that has been used in
 188 an extensive range of two-phase flow problems with various adjustments and different degrees of success.
 189 In the following sub-sections, an enhanced version of this basic framework is presented; its validity is
 190 demonstrated through a range of benchmark cases that addresses some numerically challenging problems
 191 reported in the relevant literature.

192 2.1. Adaptive Compression Scheme (Implicit)

193 To deal with the problem of numerical diffusion of α , an extra compression term is used in order to limit
 194 the convection term of Eq. 8 and consequently the thickness of the interface. Its numerical significance
 195 relays on defining local flow (u) at the interface and preventing the increase of the gradient when alpha is
 196 not constant, (i.e. the absolute value of the time derivative increases to counterbalance). The model for the
 197 compression term makes use of the two-fluid Eulerian approach, where phase fraction equations are solved
 198 separately for each individual phase, assuming that the contributions of two fluids velocities for the free
 199 surface are proportional to the corresponding phase fraction. These phase velocities (u_1 and u_2) relate with
 200 the global velocity of the one fluid approach u as:

$$u = \alpha u_1 + (1 - \alpha) u_2 \quad (9)$$

201 Replacing the above equation to Eq. 8 one gets:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \{(\alpha u_1 + (1 - \alpha) u_2) \alpha\} = 0 \quad (10)$$

202 Considering a relative velocity between the two phases ($u_r = u_1 - u_2$) which arises from the density and
 203 viscosity stresses changes across the interface, the above equation can be written in terms of the velocity of

204 the fluid:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (u_1 \alpha) - \underbrace{\nabla \cdot \{u_{r,f} \alpha (1 - \alpha)\}}_{\text{compression term}} = 0 \quad (11)$$

205 It should be noticed that in the above equation in the calculation of $\nabla \cdot (u \alpha)$ term the unknown velocity
 206 u_1 appears instead of u creating an inconsistency with the basic concept of the one fluid approach. However,
 207 since the compression term in reality is active only at the interface, continuity imposes $u_1 = u_2 = u$ and thus
 208 u_1 by u can be replaced. The discretisation of the compression term in Eq. 11 is not based directly on the
 209 calculation of the relative velocity u_r at cell faces from Eq. 9 since u_1 and u_2 are unknown. It is instead
 210 formulated based on the maximum velocity magnitude at the interface region and its direction, which is
 211 determined from the gradient of the phase fraction:

$$u_{r,f} = \min \left(C_{compr.} \frac{|\phi_f|}{|S_f|}, \max \left[\frac{|\phi_f|}{|S_f|} \right] \right) (\langle \eta_s \rangle_f) \quad (12)$$

212 where the term ϕ_f is the volumetric flux and S_f is the outward-pointing face area vector and $\langle \eta_s \rangle_f$ is
 213 the face centred interface normal vector. $\langle \cdot \rangle_f$ is used to denote interpolation from cell centres to face centres
 214 using a linear interpolation scheme, and defined as following:

$$\langle \eta_s \rangle_f = \frac{\langle \nabla \alpha \rangle_f}{|\langle \nabla \alpha \rangle_f + \delta_n|} \cdot S_f \quad (13)$$

and

$$\delta_n = \frac{1e^{-8}}{\left(\frac{\sum_N V_i}{N} \right)^{1/3}} \quad (14)$$

215 where δ_n is a small number to ensure that the denominator never becomes zero, N is the number of
 216 computational cells, for each grid block i and V_i is its volume

217 The compressive term is taken into consideration only at the interface region and it is calculated in the
 218 normal direction to the interface. The maximum operation in Eq. 12 is performed over the entire domain,
 219 while the minimum operation is done locally on each face. The constant ($C_{compr.}$) is a user-specified value,
 220 which serves as a tuning parameter. Depending on its value, different levels of compression result are
 221 calculated. For example, there is no compression for $C=0$ while there is moderate compression with

222 $C \leq 1$ and enhanced compression for $C \geq 1$. In most of the simulations presented here ($C_{compr.}$) is taken as
 223 unity, after initial trial simulations. Values higher than unity in this case may lead to non-physical results.
 224 Generally, this compression factor can take values from 0 (no compression) up to 4 (maximum compression)
 225 as suggested in the literature; the selected values are case specific. To overcome the need for a priori tuning,
 226 in the present numerical framework a new adaptive algorithm has been implemented that is based on the idea
 227 of introducing instead of a constant value for $C_{compr.}$ a dynamic one C_{adp} through the following relation:

$$C_{adp} = \left| -\frac{u_n \cdot \nabla \alpha}{|u_n| |\nabla \alpha|} \right| \quad (15)$$

$$\phi_c = \max(C_{adp}, C_{compr.}) \frac{|\phi_f|}{|S_f|} \quad (16)$$

228 where ϕ_c is the compression volumetric flux calculated, u_n represents each phase velocity normal to the
 229 interface velocity. It is expressed as

$$u_n = (U \cdot n_s) x(n_s) x(|\alpha - 0.01| * |0.99 - \alpha|) \quad (17)$$

The concept of using u_n is shown in Fig. 1: when the interface profile becomes diffusive (wide) C_{adp} value will increase accordingly in the zone of interest, while when the profile is already sharp and additional compression is not necessary C_{adp} will go to zero. Note that the compression term in Eq. 11 is only valid for the cells at the interface. However, to solve Eq. 15, a wider region of α is required. Therefore, the facial cell field is extrapolated to a wider region using the expression (near interface) in Eq. 17 as $(|\alpha - 0.01| * |0.99 - \alpha|)$. The new calculated, adaptive compression coefficient ϕ_c then substitutes the original $C_{compr.} \frac{|\phi_f|}{|S_f|}$ and Eq. 12 can be rewritten as:

$$u_{r,f} = \min\left(\phi_c, \max\left[\frac{|\phi_f|}{|S_f|}\right]\right) (\langle \eta_s \rangle_f) \quad (18)$$

230 The new equation still has a user defined value $C_{compr.}$ in cases when the adaptive coefficient is not sufficient.

231 2.2. Smoothing Scheme (Explicit)

232 By solving the transport equation for the volume fraction (Eq. 11), the value of (α) at the cell is updated.
 233 In order to proceed with the calculation of the interface surface scalar fields for the calculation of η_s and κ ,

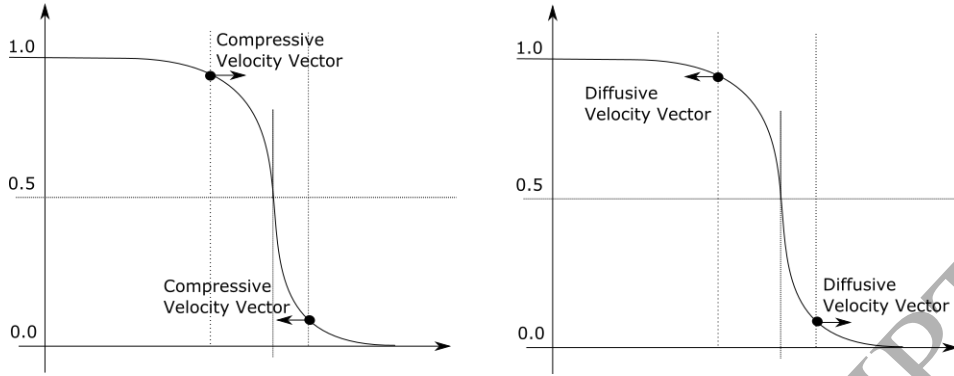


Figure 1: Schematic to represent the adaptive compression C_{adp} selection criteria

234 linear extrapolation from the cell centres is used. At this stage, the value of α sharply changes over a thin
 235 region as a result of the compression step. This abrupt change of the indicator function creates errors in
 236 calculating the normal vectors and the curvature of radius of the interface, which will be used to evaluate the
 237 interfacial forces. These errors induce non-physical parasitic currents in the interfacial region. A commonly
 238 followed approach in the literature to suppress these artefacts is to compute the interface curvature from
 239 a smoothed function α_{smooth} , which is calculated by the smoother proposed by Lafaurie et al. [17] and
 240 applied in OpenFOAM by Georgoulas et al. [46] and Raeini et al. [47]. The indicator function is artificially
 241 smoothed by interpolating it from cell centres to face centres and then back to the cell centres recursively
 242 using the following equation:

$$\alpha_{i+1} = 0.5\langle(\alpha_i)_{c \rightarrow f}\rangle_{f \rightarrow c} - 0.5\alpha_i \quad (19)$$

243 Initial trial simulations indicated that the recursive interpolation between the cell and face centres can
 244 be repeated up to three times, in order to prevent decoupling of the indicator function from the smoothed
 245 function. After smoothing is implemented, the interface normal vectors in the cells in the vicinity of the
 246 interface, are filtered using a Laplacian formulation. Equation 20 in Georgoulas et al. [46] is used in order
 247 to transform the VOF function (α_{i+1}) to a smoother function (α_{smooth}):

$$\alpha_{smooth} = \frac{\sum_{f=1}^n (\alpha_{i+1})_f S_f}{\sum_{f=1}^n S_f} \quad (20)$$

248 where the subscript denotes the face index (f) and (n) the times that the procedure is repeated in order

249 to get a smoothed field. The value at the face centre is calculated using linear interpolation. It should
 250 be stressed that smoothing tends to level out high curvature regions and should therefore be applied only
 251 up to the level that is strictly necessary to sufficiently suppress parasitic currents. After calculating the
 252 (α_{smooth}), the interface normal vectors are computed using 7, and the interface curvature at the cell centres
 253 can be obtained by $\kappa_f = -\nabla \cdot (\eta_s)$. Then in order to model the motion of the interfaces more accurately,
 254 an additional smoothing operation is performed to the curvature. The interface curvature in the direction
 255 normal to the interface is calculated, recursively for two iterations:

$$\kappa_{s,i+1} = 2\sqrt{\alpha_{smooth}(1-\alpha_{smooth})}\kappa_f + (1-2\sqrt{\alpha_{smooth}(1-\alpha_{smooth})}) * \frac{\langle\langle\kappa_{s,i}\sqrt{\alpha_{smooth}(1-\alpha_{smooth})}\rangle\rangle_{c\rightarrow f}\rangle_{f\rightarrow c}}{\langle\langle\sqrt{\alpha_{smooth}(1-\alpha_{smooth})}\rangle\rangle_{c\rightarrow f}\rangle_{f\rightarrow c}} \quad (21)$$

256 This additional smoothing procedure diffuses the variable κ_f away from the interface. Finally, the
 257 interface curvature at the face centres κ_{final} is calculated using a weighted interpolation method that is
 258 suggested by Renardy and Renardy [37]:

$$\kappa_{final} = \frac{\langle\kappa_{s,i}\sqrt{\alpha_{smooth}(1-\alpha_{smooth})}\rangle}{\langle\sqrt{\alpha_{smooth}(1-\alpha_{smooth})}\rangle} \quad (22)$$

259 where the interface curvature κ_{final} is obtained at face centres.

260 2.3. Sharpening Scheme (Explicit)

261 Recalling Eq. 6, the surface tension forces are calculated at the face centres based on the following
 262 equation:

$$f_s = (\sigma\kappa\delta_s)_f \dot{\eta}_s = \sigma\kappa_{final}\delta_{sf} \quad (23)$$

263 In order to control the sharpness of the surface tension forces, the delta δ_s is calculated from a sharpened
 264 indicator function α_{sh} as $\delta_s = \nabla_f^\perp \alpha_{sh}$, where ∇_f^\perp denotes the gradient normal to the face f . In Eq. 23 the
 265 surface tension force term is non-zero only at the faces across which the indicator function α_{sh} has values.
 266 The α_{sh} represents a modified indicator function, which is obtained by curtailing the original indicator
 267 function α as follows;

$$\alpha_{sh} = \frac{1}{1 - C_{sh}} \left[\min \left(\max \left(\alpha, 1 - \frac{C_{sh}}{2} \right), 1 - \frac{C_{sh}}{2} \right) - \frac{C_{sh}}{2} \right] \quad (24)$$

268 where C_{sh} is the sharpening coefficient. From Eq. 24 one can notice that, as the sharpening coefficient (C_{sh})
 269 value increases, the unphysical interface diffusion decreases (i.e., it limits the effect of unphysical values
 270 at the interface, by imposing a restriction on alpha α - as demonstrated). A zero value of C_{sh} will lead to
 271 the original CSF formulation, while as C_{sh} value increases the interface becomes sharper. As expected, the
 272 continuous α_{smooth} - approach has a smooth (and diffused) transition across the interface, whereas the sharp
 273 α_{sh} - approach has a more abrupt transition with larger extremes. At high values of C_{sh} (0.5 to 0.9), Eq.
 274 24 limits the indicator function α - where values between (0 to 0.4) are summed to zero and values between
 275 (0.6 to 1) are summed to be one. This implementation introduces a sharper approach of the surface tension
 276 forces as discussed by Aboukhedr et al. [48]. Values in the range of (0.5) C_{sh} were observed to give the best
 277 results for the most of our test cases.

278 2.4. Capillary Pressure Jump Modelling

279 In order to avoid difficulties associated with the discretisation of the capillary force f_c , rearrangement of
 280 the terms on the right hand side of the momentum equation is conducted following the work of [47], where
 281 Eq. 2 is rewritten in terms of the microscopic capillary pressure p_c :

$$\frac{D}{Dt}(\rho u) - \nabla \cdot T = -\nabla p_d + f', \quad (25)$$

$$f' = \rho g + f_s - \nabla p_c \quad (26)$$

282 where the dynamic pressure is defined as $p_d = p - p_c$. This approach includes explicitly the effect of
 283 capillary forces in the Navier-Stokes equations and allows for the filtering of the numerical errors related to
 284 the inaccurate calculation of capillary forces. Considering a static fluid configuration for a two phase flow,
 285 the stress tensor reduces to the form ($n \cdot \tau \cdot n = -p$), and the normal stress balance is assumed to have the
 286 form of ($p_c = \sigma \nabla \cdot n$) [49]. Then, the pressure jump across the interface is balanced by the curvature force
 287 at the interface.

$$\nabla \cdot \nabla p_c = \nabla \cdot f_s \quad (27)$$

288 Assuming that pressure jumps can sustain normal stress jumps across a fluid interface, they do not
 289 contribute to the tangential stress jump. Consequently, tangential surface stresses can only be balanced by
 290 viscous stresses. Therefore one can apply a boundary condition of:

$$\frac{\delta p_c}{\delta n_s} = 0 \quad (28)$$

291 where n_s is the normal direction to the boundaries. By including this set of equation to the Navier-Stokes
 292 equations, one can have a better balancing of momentum, hence filtering the numerical errors related to
 293 inaccurate calculations of the surface tension forces.

294 2.5. Filtering numerical errors

295 As the result of the numerical unbalance discussed in the previous sections when modelling the move-
 296 ment of a closed interface, it is difficult to maintain the zero-net capillary force, while modelling the move-
 297 ment of the interface. Hence it is difficult to decrease the errors in the calculation of capillary forces to zero
 298 $\oint f_s \cdot A_s = 0$ where A_s is the interface vector area. Raeini et al. [47] proposed as a solution to filter the
 299 non-physical fluxes generated due to the inconsistent calculation of capillary forces based on a user defined
 300 cut-off. The cut-off uses a thresholding scheme, aiming to filter the capillary fluxes ($\phi = |S_f|(f_s - \nabla_f^\perp p_c)$)
 301 and eliminate the problems related to the violation of the zero-net capillary force constraint on a closed
 302 interface. The proposed filtering procedure explicitly sets the capillary fluxes to zero when their magnitude
 303 is of the order of the numerical errors. The filter starts from setting an error threshold as:

$$\phi_{threshold} = U_f |f_s|_{avg} |S_f| \quad (29)$$

304 where $\phi_{threshold}$ is the threshold value below which capillary fluxes are set to zero and $|f_s|_{avg}$ is the average
 305 value of capillary forces over all faces. The filtering coefficient U_f is used to eliminate the errors in the
 306 capillary fluxes. Here a different U_f is used, so for different cases the U_f value will be set, which implies
 307 that the capillary fluxes are set to zero. After selecting the threshold, the capillary flux is filtered as:

$$\phi_{filter} = |S_f|(f - \nabla_f^\perp p_c) - \max(\min(|S_f|(f - \nabla_f^\perp p_c), \phi_{threshold}), -\phi_{threshold}) \quad (30)$$

308 Using this filtering method, numerical errors in capillary forces causing instabilities or introducing large
 309 errors in the velocity field are prevented. By using the aforementioned filtering technique, the problem of
 310 stiffness is found to be reduced by eliminating the high frequency capillary waves when the capillary forces
 311 are close to equilibrium with capillary pressure. Consequently, it allows larger time-steps to be used when
 312 modelling interface motion at low capillary numbers

313 3. Algorithm Implementation

314 The modelling approach for compression has been implemented using the OpenFOAM- Plus finite
 315 volume library [16], which is based on the VoF-based solver interFoam [50]. No geometric interface recon-
 316 struction or tracking is performed in interFoam; rather, a compressive velocity field is superimposed in the
 317 vicinity of the interface to counteract numerical diffusion as already discussed in section 2.1. In the original
 318 VoF-based solver (interFoam), the time step is only adjusted to satisfy the Courant-Friedrichs-Lewy (CFL)
 319 condition. A semi-implicit variant of MULES developed by OpenFOAM is used here which combines op-
 320 erator splitting with application of the MULES limiter to an explicit correction. It first executes an implicit
 321 predictor step, based on purely bounded numerical operators, before constructing an explicit correction on
 322 which the MULES limiter is applied. This approach maintains boundedness and stability at an arbitrarily
 323 large Courant number. Accuracy considerations generally dictate that the correction is updated and applied
 324 frequently, but the semi-implicit approach is overall substantially faster than the explicit method with its
 325 very strict limit on time-step. The indicator function is advected using Crank-Nicholson scheme for half of
 326 the time step using the fluxes at the beginning of each time step. Then the equations for the advection of the
 327 indicator function for the second half of the time step are solved iteratively in two loops. The discretised
 328 phase fraction (Eq. 11) is then solved for a user-defined number of sub-cycles (typically 2 to 3) using the
 329 multidimensional universal limiter with the [MULES] solver. Once the updated phase field is obtained, the
 330 algorithm enters in the pressure-velocity correction loop.

331 4. Results, Validation and Discussion

332 In the following sections, numerical simulations are presented for a range of benchmark cases that
 333 assess the performance of the proposed model. As a first benchmark case, a stationary single droplet and a
 334 pair of droplets (in the absence of gravity) have been considered. The convergence of velocity and capillary
 335 pressure to the theoretical solution is demonstrated. This test case assesses the performance of solvers in
 336 terms of spurious currents suppression. Then two other cases, commonly used in the literature, namely
 337 the Notched disc in rotating flow Zalesak [51] and the Circle in a vortex field Roenby et al. [34], Rider
 338 and Kothe [42] are examined. Finally, a more indicative example of flows through narrow passages is
 339 considered. This includes the generation of millimetric size bubbles in a T-junction. For the T-junction case,
 340 the prediction of any non-smoothed and diffused interface is accompanied by the development of spurious
 341 velocities resulting in unphysical results in comparison with the available experimental data. Calculations
 342 with the standard VoF-based solver of OpenFOAM (interFoam) are also included for completeness.

343 4.1. Droplet relaxation at static equilibrium

344 When an immiscible cubic 'droplet' fluid is immersed in fluid domain (in the absence of gravity), surface
 345 tension will force the formation of the spherical equilibrium shape. The force balance between surface
 346 tension and capillary pressure should converge to an exact solution of zero velocity field. The corresponding
 347 pressure field should jump from a constant value p_0 outside the droplet to a value $p_0 + 2\sigma/R$ inside the
 348 droplet. Modelling the relaxation process of an oil droplet ($D_0 = 30 \mu m$) in water at static equilibrium serves
 349 as an initial demonstration case for testing the suggested methodology, [at a mesh resolution of \(60x60x60\)](#).
 350 The fluid properties of the background phase (water) density ρ_1 is 998 kg/m^3 , and the viscosity ν_1 is
 351 $1.004 \times 10^{-6} \text{ m}^2/\text{s}$, while the droplet phase (oil) density ρ_2 is 806.6 kg/m^3 , and the viscosity ν_2 is $2.1 \times 10^{-6} \text{ m}^2/\text{s}$,
 352 and surface tension of 0.02 kg/s^2 . These values result to ($\Delta P_c = \frac{2\sigma}{R} = 2666 \text{ Pa}$). The calculation set up
 353 includes a single cubic fluid element patched centrally to the computational domain and it is allowed to
 354 relax to a static spherical shape as shown in Fig. 2. It has been shown in the literature [52] that under these
 355 conditions and depending on the accuracy of the interface tracking/capturing scheme, non-physical vortex-
 356 like velocities may develop in the vicinity of the interface and can result in its destabilization. Tables 1 and 2
 357 demonstrate the different controlling parameters that have been tested. The main testing parameters shown

358 in the table are: (i) the flux filtering percentage U_f as presented in Eq. 29, (ii) the number of smoothing
 359 loops n as presented in Eq. 20, (iii) the sharpening coefficient C_{sh} as presented in Eq. 24 and finally (iv)
 360 the compression coefficient $C_{compr.}$ as presented in Eq. 12. Each series of test cases is designed to examine
 361 the effect of the mentioned models on parasitic currents and pressure jump calculation accuracy. Cases (S)
 362 examine the effect of smoothing loops number in the absence of interface sharpening and filtering. Cases
 363 (A) are designed to study the effect of error filtering percentage in the absence of smoothing loops and
 364 interface sharpening. Cases (B) examine the combined effect of filtering and smoothing in the absence
 365 of interface sharpening, while cases (SE) and (SF) are designed to test the combined effect of smoothing
 366 and filtering in the presence of interface sharpening and interface compression, respectively. The adaptive
 367 compression scheme introduced in the previous section, is not activated in this case in order to investigate
 368 the effect of different pre-specified compression levels on the parasitic current development.

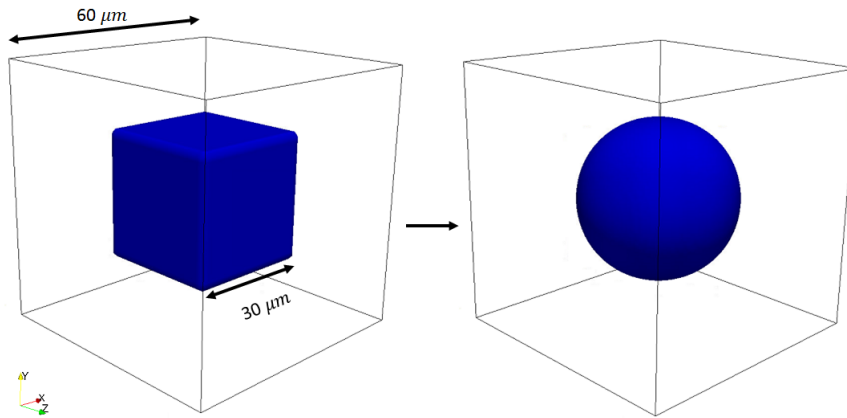


Figure 2: Computational domain for modelling static droplet, (left) initial condition a cube of size $D_0 = 30 \mu\text{m}$, and (right) static shape of droplet. Mesh size $R/\delta x = 15$ at $t = 0.0025$ s.

	$U_f\%$	n (Eq. 20)		$U_f\%$	n	Filter	$U_f\%$	n (Eq. 20)
Case S1	0	2	Case A1	0.01	0	Case B1	0.05	2
Case S2	0	5	Case A2	0.05	0	Case B2	0.05	5
Case S3	0	10	Case A3	0.1	0	Case B3	0.05	10
Case S4	0	20	Case A4	0.2	0	Case B4	0.05	20

Table 1: Case set-up testing the influence of smoothing and capillary filtering values ($U_f\%$ and n) without the effect of sharpening or compression coefficients (C_{sh} and C_{comp} are set to zero)

369 The maximum velocity magnitude in the computational domain is presented as a function of various
 370 numerical parameters. If inertial and viscous terms balance in the momentum equation then parasitic veloc-

	$U_f\%$	n (Eq. 20)	C_{sh} (Eq. 24)	C_{comp}
Case SE1	0.05	10	0.1	0
Case SE2	0.05	10	0.5	0
Case SE3	0.05	5	0.1	0
Case SE4	0.05	5	0.5	0
Case SF1	0.05	10	0.5	0.5
Case SF2	0.05	10	0.5	1
Case SF3	0.05	10	0.5	2
Case SF4	0.05	10	0.5	3

Table 2: Case set-up testing the influence of smoothing and capillary filtering values ($U_f\%$ and n) including the effect of sharpening or compression coefficients

ities should be zero. However, the CSF technique introduces an unbalance by replacing the surface force by a volume force which acts over the small region surrounding the continuous phase interface. The surface force suggested by Brackbill et al. [8] includes a density correction as $1/(We \frac{\rho}{\langle \rho \rangle} \kappa n)$ for modelling systems where the phases have unequal density, where ρ is the local density and $\langle \rho \rangle$ is the average non-dimensional density of the two phases. Including these two variables does not affect the total magnitude of force applied, but weights the force more towards regions of higher density. This tends to produce more uniform fluid accelerations across the width of the interface region. Such a force is irrotational and so it can be represented as the gradient of a scalar field. Referring to the momentum equation 2 the surface tension force has to be precisely balanced by the pressure gradient term, with all velocity dependent terms, and thus velocities, being zero. The commonly used VoF numerical implementation of this system differs from this ideal implementation of α , which when discretised represents the volume fraction integrated over the dimensions of a computational mesh cell and varies by a small amount in the radial direction. This results in n -(the normal to the interface) not being precisely directed in the radial direction, κ value varying slightly and the complete interface volume force having a rotational component. The rotational component of the surface tension force cannot be balanced by the irrotational pressure gradient term. So it must be balanced instead by one or more of the three other velocity dependent terms. As these velocity terms (inertial transient, inertial advection and viscous) all require non-zero velocities if they themselves are to be non-zero, spurious currents develop. Looking into the parasitic velocity magnitude for the standard (interFoam) solver during the relaxation period (Fig. 3a), parasitic velocities are high and depend on the compression level. As the value of C_{compr} . increases, the maximum velocity also increases. This might appear to be counter intuitive

391 since increased compression should result in sharper interfaces, nevertheless, in this work the smooth α field
 392 is only used for accurate curvature calculation, but for the rest of the equations the sharpened field had been
 393 used curvature κ and the normal vectors. However the sharper the interface the more numerical challenging
 394 becomes the calculation of derivatives. Fig. 3a indicates this paradox while Figure 3b presents a graphical
 395 explanation. It can be seen that as $C_{compr.}$ increases then vortex like structures develop randomly around the
 396 interface that prevent the droplet from relaxing to equilibrium.

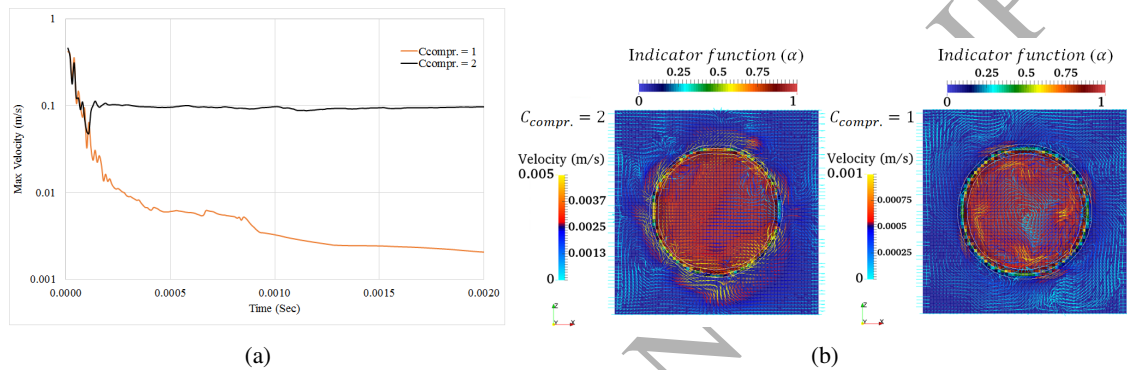


Figure 3: (a) Evolution of maximum velocity during droplet relaxation using the standard (interFoam) solver with two different interface compression ($C_{compr.}$). (b) values Snapshot of the interface shape after the relaxation of the oil droplet using the standard (interFoam). Velocity vectors near to the interface for different interface compression values are presented.

397 Testing the smoothing effect presented in Eqs. (19, 20 and 21) using the modified solver by varying
 398 the number of smoothing loops (n) as shown of Table (1) is also performed in the presented sub-section.
 399 The mentioned set-up in cases S1,S2,S3,S4 is used to investigate the effect of smoothing loops on the
 400 parasitic currents, isolated from the other examined controlling parameters. It is evident from Fig. 4e that
 401 by increasing the number of smoothing loops, the magnitude of the parasitic currents decreases. However,
 402 it should be pointed out that this reduction of parasitic currents, comes at the cost of a corresponding
 403 increase in the interface region thickness. Increasing the smoothing loops to 20, the interface thickness
 404 increases almost 4 times (6 cells) and parasitic currents tend to develop again and increase by time at a
 405 certain point after the relaxation of the droplet. The effect of varying the coefficient U_f for filtering the
 406 capillary forces parallel to the interface (see Eq. 30) is revealed from cases A1 to A4 of Table 1; a decrease
 407 of the parasitic currents due to the wrong flux filtering near to the interface can be noticed. In the absence
 408 of smoothing loops and just changing the filter value U_f , a significant decrease of the parasitic currents
 409 is observed as shown in Fig. 4b. Moreover, an optimum decrease in parasitic currents using a value of

410 $U_f = 0.05$ is observed (Table 1). The decrease of parasitic currents magnitude in this case is a combination
 411 of the interface treatment of Eq. 19 and the flux filtering without any smoothing loops being performed.
 412 Looking at Fig. 4b one can observe the asymmetric distribution of the velocity vector field with almost
 413 zero velocity inside the droplet. **By examining the isolated filtering coefficient U_f and smoothing loops**
 414 **n , the suggested framework has been noticed to reduce the spurious velocities**, by almost four orders of
 415 magnitude, over a relatively long period. Cases B1 to B3 of Table 1 reveal the effect of combining both
 416 techniques (smoothing and flux filtering) for damping the parasitic currents; one of the parameters has kept
 417 constant - in this case, U_f . Comparing cases (B2) presented in Figures 4c with the previously presented
 418 cases S and A, a major improvement in velocity reduction can be seen. In Fig 5 (B) a reduction of almost
 419 four orders of magnitude, when compared with the standard solver, has been achieved. **By examining the**
 420 **deviation from the theoretical results compared to the standard interFoam using filtering and smoothing**
 421 **models as shown in Table 3, the suggested models reduce the maximum velocity field as seen in cases (S2**
 422 **and A1), then it start to increase, due to the excessive interface smoothing or the un-balanced capillary**
 423 **forces**. Selecting the best smoothing and the filtering coefficient combination ($5 < n < 10$ and $U_f = 0.05$),
 424 the effect of the sharpening model Eq. 24 is now examined. In Table 2 cases (SE1 to SE4), the C_{sh} has
 425 been varied. Looking at Fig. 4a, a great reduction in the interface thickness can be seen reaching almost
 426 one grid cell. By combining the effect of sharpening, filtering and smoothing techniques, the same order
 427 of magnitude for parasitic currents with a significant decrease in interface thickness has been achieved. It
 428 has also been found that in SF1 case specifically, a very good balance in the velocity vector field with zero
 429 velocity inside the droplet (Fig. 5) has been achieved.

430 As mentioned before, the literature review has revealed the negative effect of increasing the value of
 431 compression coefficient, since as the value of $C_{compr.}$ increases the magnitude of parasitic currents also
 432 increases. Using the same droplet test case, the effect of increasing the $C_{compr.}$ value on the parasitic current
 433 is demonstrated, but this time after applying the smoothing and flux filter models. It should be noted, the
 434 aforementioned adaptive compression model is not tested in this case yet, as it will be tested in the next
 435 section. In Table 2 cases (SF1 to SF4), the cases using the best combination of the previously mentioned
 436 smoothing and filter values coefficient are used with different compression values. The overall maximum
 437 velocity values are higher compared to those archived using no compression; **nevertheless, these are still**

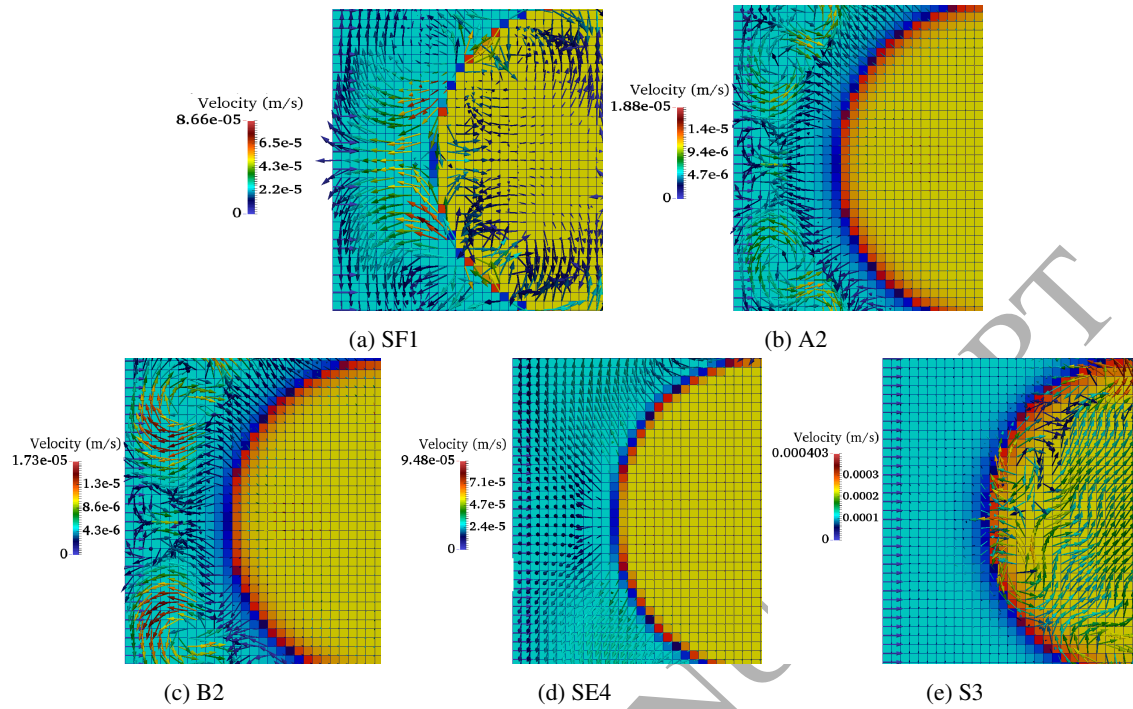


Figure 4: Effect of varying model coefficients described in table 1 and 2 on parasitic currents, all figures are showing velocity vector field at $t=0.0024$ sec. Figures are coloured with indicator function α_{sharp} as yellow showing oil phase inside the droplet and bright blue showing water outside the droplet

438 **lower than those achieved using** the standard solver. A swirling behaviour around the external diagonal
 439 direction of the droplet had been noticed as shown in Fig. 4b and 4c. The observed small swirling velocity
 440 confirms that the unbalanced surface tension force may increase parasitic currents at one specific location
 441 due to this swirling behaviour around the droplet interface. At the same time the effects of the smoothing
 442 and the filtering can have positive effect on decaying these swirling velocities.

443 The behaviour of the droplet when different parameters are considered is important in assessing the
 444 impact that the parasitic currents have on the results. Similar simulations but with varying domain sizes
 445 (not included in this study) showed that when the parasitic currents were inertia-driven at the deformation
 446 phase they spread further across the computational domain. Depending on the nature of the simulation
 447 being considered, this may mean that inertia-driven parasitic currents have a greater impact on the results.
 448 Quantifying this effect would be difficult, as any integral measure of the parasitic currents – such as the
 449 total kinetic energy within the domain for example – would be dependent on additional geometrical factors,
 450 such as the domain size and interfacial area. While the form of the velocity field is changing with time

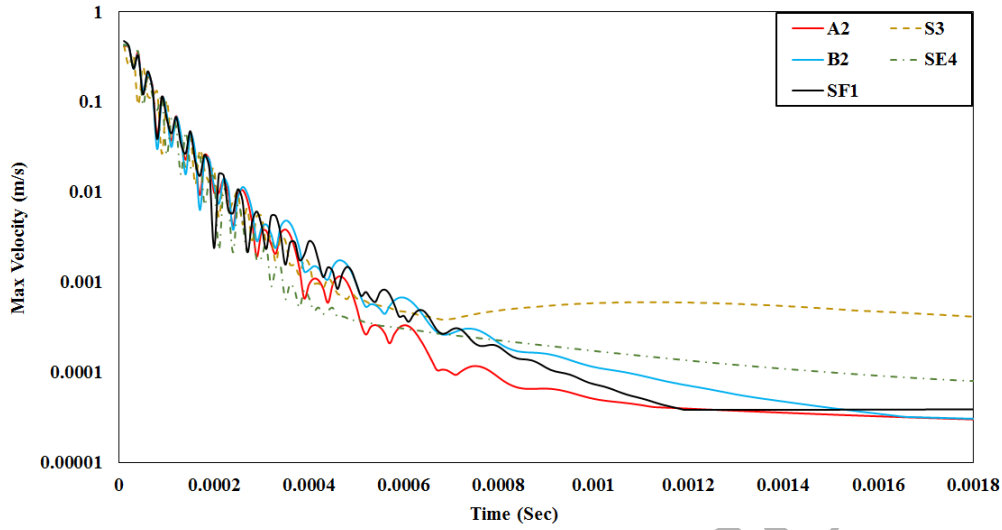


Figure 5: Effect of varying models coefficients presented in table 1 and 2 on maximum parasitic currents over period of time

451 one can conclude that the parasitic currents are dominated by inertia. The assessment of the effect of
 452 different parameters on the maximum velocity can also be presented in the percentage of divergence from
 453 the standard solver results as illustrated by Eq. 31;

$$E_{parasitic} = \frac{\min(U)}{\min(U)_{C_{\alpha}=2}} \quad (31)$$

454 where $E_{parasitic}$ represents the error calculated by the $\min(U)$ to be the minimum velocity in the domain
 455 achieved using modified solver and $\min(U)_{C_{\alpha}=2}$ to be minimum velocity using standard solver at $C_{compr.} = 2$
 456 during the droplet relaxation over a long time interval. Table 3 shows that the magnitude of parasitic currents
 457 decreases to minimal in case (B2) where compression and sharpening are null; one can also achieve the same
 458 level of reduction in parasitic currents after applying sharpening, as in case (SE3) and with only a slight
 459 further increase by adding compression as in case (SF1). Table 3 shows numerically predicted pressure
 460 difference between the relaxed spherical droplet and the ambient liquid along the droplet diameter axis for
 461 each of the 20 simulated cases, in comparison with the theoretical value predicted from the Laplace equation
 462 [see [53] for more details]. The results are presented in terms of the errors in predicted capillary pressure,
 463 $Error_{p_c}$, defined as follows:

$$Error_{p_c} = \frac{p_c - (p_c)_{theoretical}}{(p_c)_{theoretical}} / \left(\frac{P - P_{theoretical}}{P_{theoretical}} \right)_{interFoam_{c_{\alpha}=2}} \quad (32)$$

464 where p_c is the calculated capillary pressure using the developed solver, and the P is the calculated pressure
 465 using the standard interFoam with compression value of two. The $Error_{p_c}$ presents the deviation of the cal-
 466 culated capillary pressure using the developed solver and the standard solver with respect to the theoretical
 467 capillary pressure. Equation 32 shows the reduction in error between the developed solver and the standard
 468 solver using compression ($C_{compr.} = 2$). In all the presented cases, reduction in predicting the capillary
 469 pressure by 40% can be seen.

<i>Smooth</i>	S1	S2	S3	S4
$Error_{p_c} \%$	41.43	40.57	39.64	33.38
$E_{parasitic}$	0.0051	0.0053	0.0080	0.0112
<i>Filter</i>	A1	A2	A3	A4
$Error_{p_c} \%$	45.55	45.51	45.51	45.63
$E_{parasitic}$	0.0031	0.0006	0.0011	0.0014
<i>Filter</i>	B1	B2	B3	B4
$Error_{p_c} \%$	44.36	43.39	42.20	40.91
$E_{parasitic}$	0.0005	0.0006	0.0013	0.0032
<i>Sharp</i>	SE1	SE2	SE3	SE4
$Error_{p_c} \%$	43.04	45.14	43.97	46.11
$E_{parasitic}$	0.0008	0.0024	0.0007	0.0015
<i>Sharp</i>	SF1	SF2	SF3	SF4
$Error_{p_c} \%$	49.79	50.20	50.12	49.95
$E_{parasitic}$	0.0008	0.0045	0.0057	0.0067

Table 3: Reduction in predicted capillary pressure and parasitic currents compared to the standard interFoam

470 4.2. Interacting Parasitic Currents of two relaxing droplets

471 In this section the effect of parasitic current interaction for the case of two stagnant droplets that undergo
 472 the same relaxation process is discussed. The same droplet properties as in the previous test case have been
 473 used (see Section 4.1). When two droplets are found in the same domain in close proximity, the parasitic
 474 currents may interact resulting in artificial movement of the droplets and eventually merging. Figure 7
 475 shows the velocity magnitude on the droplet represented by the 0.5 liquid volume fraction iso-surface. The
 476 same set of parameters are utilised as in (A2, B2, SE3 and SF1) cases mentioned in Tables 1 and 2. One
 477 can notice in Fig. 7a to Fig. 7c that the two droplets have merged to one big droplet located at the centre of

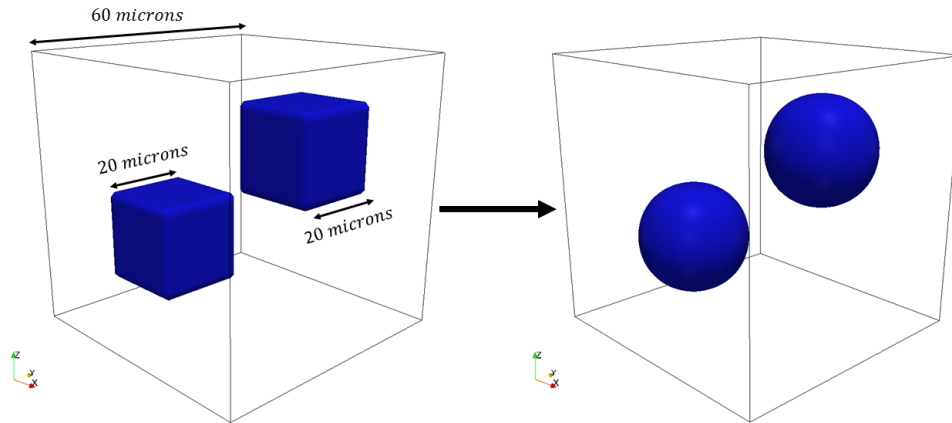


Figure 6: Computational domain showing two static droplets , (left) initial condition a cube of size $D_0 = 20 \mu\text{m}$ each, and (right) static shape of droplet as two boxes.

478 the computational domain. In contrast Fig. 7d shows that the two droplets remain in their initial position as
 479 they should. This can be considered as a demonstration that optimising compression for one case does not
 480 necessarily mean that can offer optimum results for other similar cases and the solver should automatically
 481 adapt the needed compression. Hence, in the next sections that consider cases with higher deformation of
 482 the interface we are going to introduce the adaptive solver.

483 4.3. Notched disc in rotating flow

484 In addition to the static droplet test cases, the rotation test of the slotted disk, which is known as the
 485 Zalesak problem [51] has been tested. The Zalesaks circle disk is initially slotted at the centre $(0.5,0,0.75)$ of
 486 a 2D unit square domain. The disk is subjected to a rotational movement under the influence of a rotational
 487 field that is defined by the following equations:

$$u(x) = -2\pi(x - x_0) \quad (33)$$

$$w(z) = 2\pi(z - z_0) \quad (34)$$

488 where $u(x)$, $w(z)$ are the imposed velocity components. By applying this velocity, one complete rotation
 489 of the disk is completed within $t = 1\text{sec}$. For all simulations performed for this test case, a fixed time-step has
 490 been used, keeping the Courant number equal to 0.5. The initial disk configuration used for the simulation

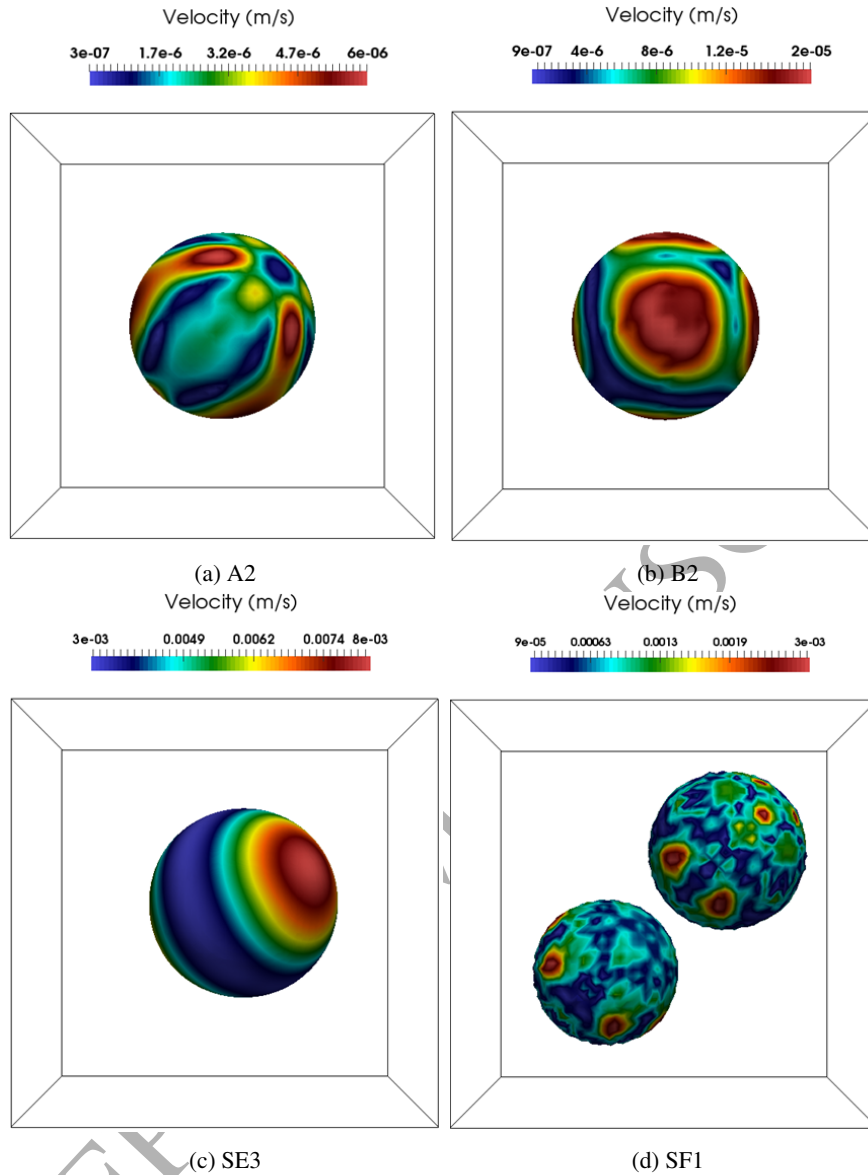


Figure 7: Effect of combined flux filtering and smoothing in the presence of sharpening model on the interaction of parasitic velocity field. All figures are showing the velocity field at $t = 0.0024$ sec on the indicator function α_{sharp} iso-contour = 0.5

491 is presented in Fig. 8. Three different mesh densities were used consisting of 64x64, 200x200 and 400x400
 492 cells, respectively.

493 Figures 9 and 10 show the comparison between the standard solver using different compression ($C_{compr.}$)
 494 values and the developed adaptive solver using different sharpening (C_{sh}) values. In each plot, the exact
 495 initial and final interface shape is presented. In all the figures, the iso-contours values of indicator function
 496 alpha α of (0.1, 0.5 and 0.9) after one revolution of the disk are shown. The reason of presenting three

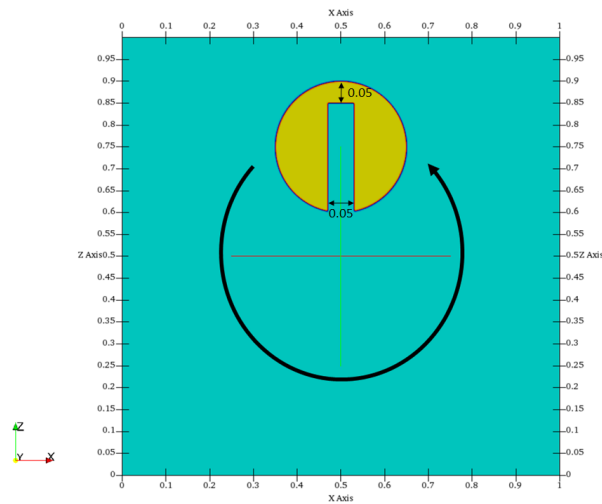


Figure 8: Schematic representation of two dimensional Zalesak's Disk benchmark test case described at [54].

497 contour lines is to better explore the effect of the adaptive compression model on both the interface diffusion
 498 and the overall disk shape. For the coarse mesh (64x64) neither using the standard interFoam with three
 499 compression values ($C_{compr.} = 0, 1$ and 4), nor the three values for C_{sh} , ($C_{sh} = 0.1, 0.5$ and 0.9) for the
 500 adaptive modified solver, can provide a satisfactory interface representation. One can even notice that due
 501 to the large interface deformation and diffusion, the interface iso-contour of $\alpha = 0.9$ at Fig. 9(a) has
 502 disappeared for the standard solver. Nevertheless, for the adaptive modified solver cases, the modified
 503 solver can keep the main geometrical features as seen in Figs. 10(a,d,g). By using high compression as
 504 in Fig. 9(g), one can notice a reduction in the interface thickness, although a rather high deformation
 505 and corrugated shape of the final disk shape has been noticed. Comparing Fig. 9(g) to Fig. 10(g) one
 506 can notice the effectiveness of the adaptive model that preserves the geometrical outline of the disk while
 507 the sharpening model decreases the interface thickness. Moving to a finer mesh (200x200), high interface
 508 diffusion using the standard interFoam with no compression ($C_{compr.} = 0$) Fig. 9(b) has been noticed. The
 509 higher grid resolution is not adequate to provide remedies to the previously mentioned deficiencies noticed
 510 in the coarser mesh using interFoam. The highly diffusive interface using the standard interFoam also did
 511 not maintain the 0.9 iso-contour making two oval shapes at the sides. For higher compression values Fig.
 512 9(e,h) although the disk shape is preserved by the standard solver, the interface is significantly deformed
 513 near the outer disk boundary. Use of the adaptive solver Fig. 10(b,e,h) shows better consistency for the shape
 514 regardless of the imposed sharpening level. Moreover, the adaptive compression eliminates any irregular

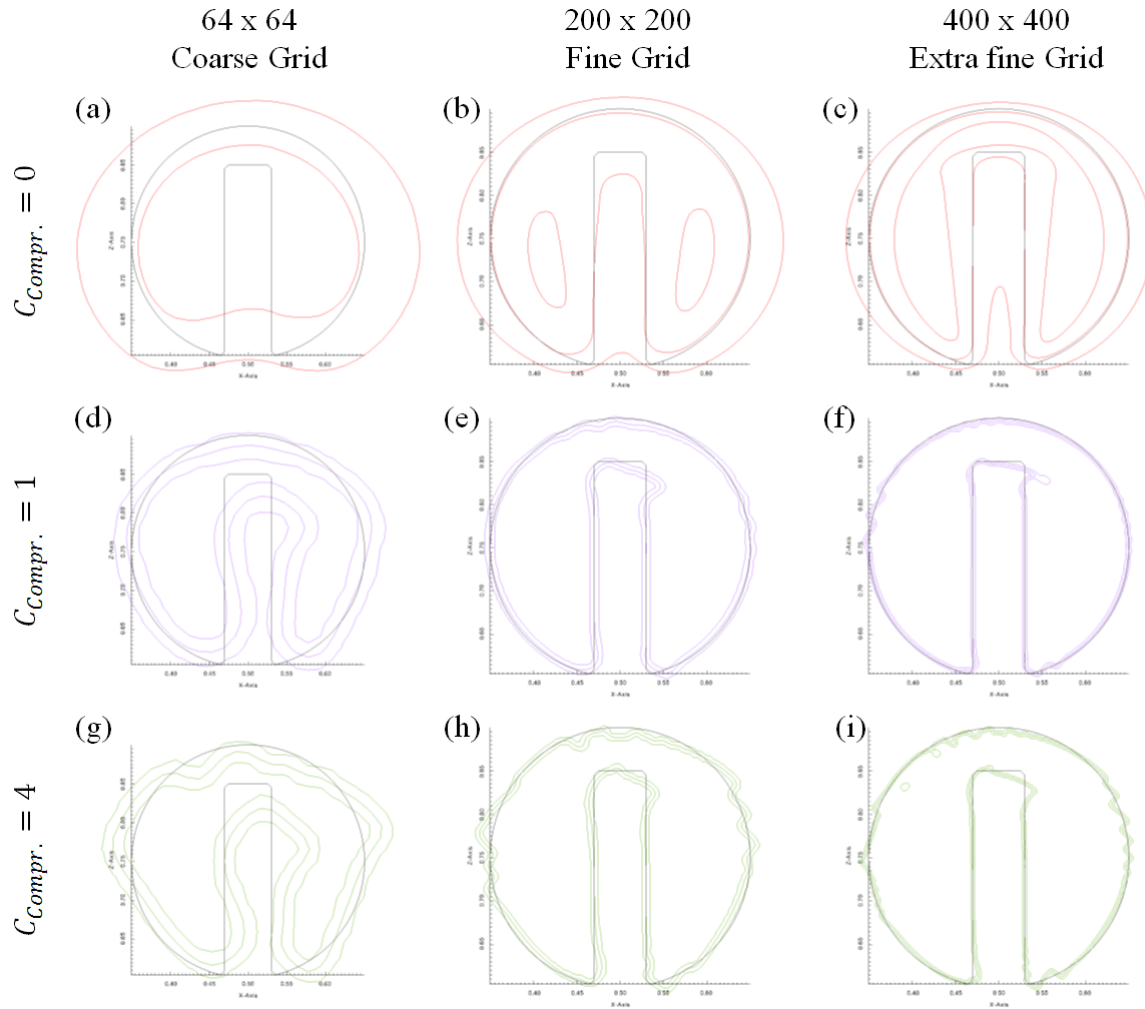


Figure 9: Zalesak disk after one revolution. Iso-contours for indicator function α ($\alpha = 0.1, 0.5$ and 0.9) are plotted for the standard interFoam using different compression values, together with the reference shape.

515 shapes compared to the standard solver. Figure 10(h) especially shows an excellent agreement with the
 516 original circular shape layout. This test case also demonstrates the role of the sharpening value C_{sh} which
 517 can help in controlling the interface diffusion depending on the case under consideration. To examine our
 518 adaptive solver mesh dependency, the mesh has been doubled to 400×400 . Even for this fine grid resolution
 519 case the standard solver gives inaccurate disk shape regardless of the compression value used, as none of
 520 them is adequate to balance the interface shape. A zero compression value using the standard interFoam
 521 preserves the characteristic shape for the first time (see Fig. 9(c), compared to Fig. 9(a,b)). For the higher
 522 compression values as in Fig. 9(f,i), high corrugated regions at the interface have been observed. Using the

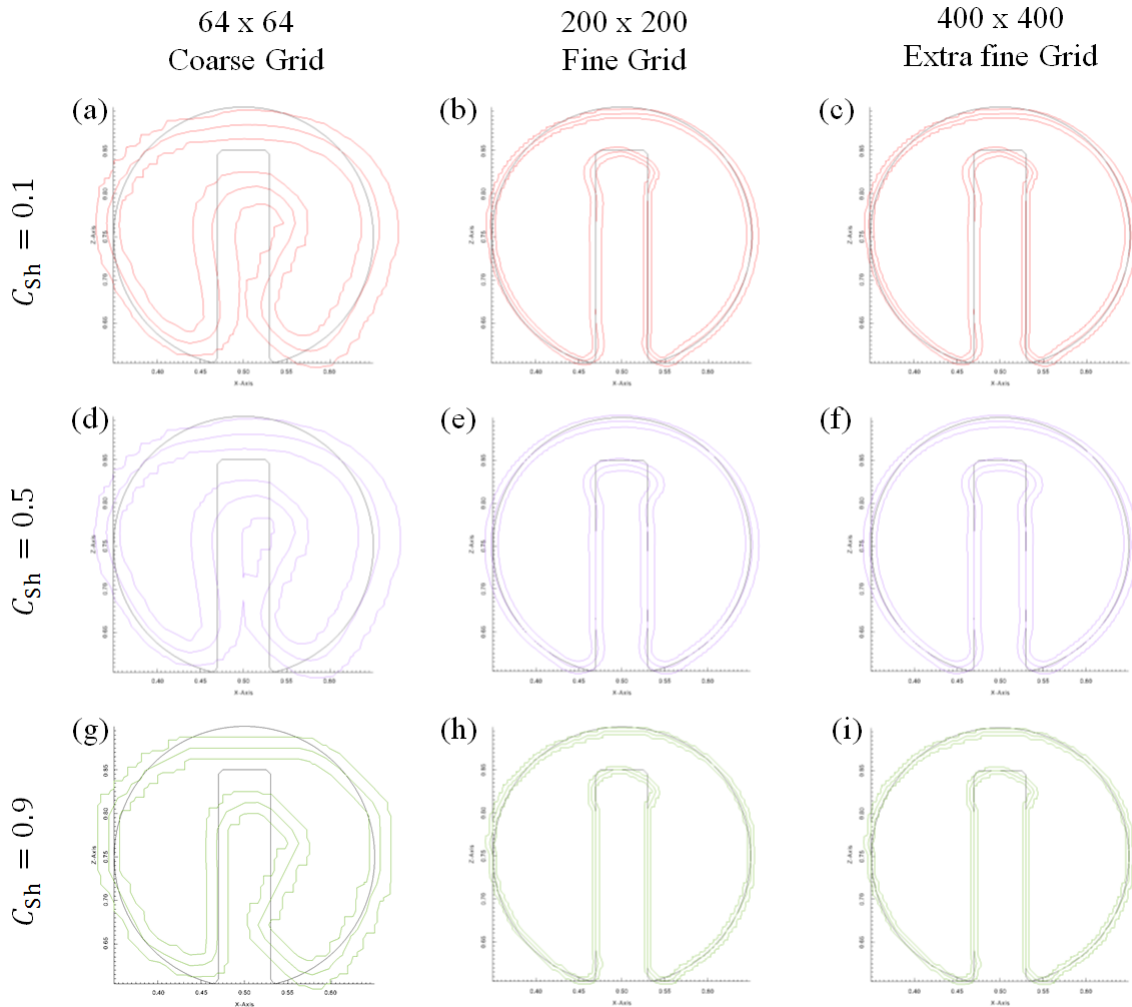


Figure 10: Zalesak disk after one revolution. Iso-contours of indicator function alpha sharp ($\alpha_{sh} = 0.1, 0.5$ and 0.9) are plotted for the adaptive modified solver using different sharpening coefficients, together with the reference shape.

523 adaptive modified solver a better disk shape representation has been obtained, regardless of the sharpening
 524 coefficient value C_{sh} (see Fig. 10(c,f,i)). Moreover, by using the three different sharpening coefficients C_{sh}
 525 a thickness of approximately 1-2 cells has been preserved. Also a minimum difference between the fine and
 526 the extra fine grid in terms of interface thickness has been observed, and sharpening algorithm shows the
 527 perfect fit to the internal notch. These observations indicate that adaptive compression is less sensitive to
 528 tuning parameters such as the sharpening (see Eq. 24), which is not effective for coarse grid resolution.

529 For completeness, results included in [20] are also shown. In [20] various commonly used interface
 530 capturing methods have been presented for the same test case; these include the standard compression

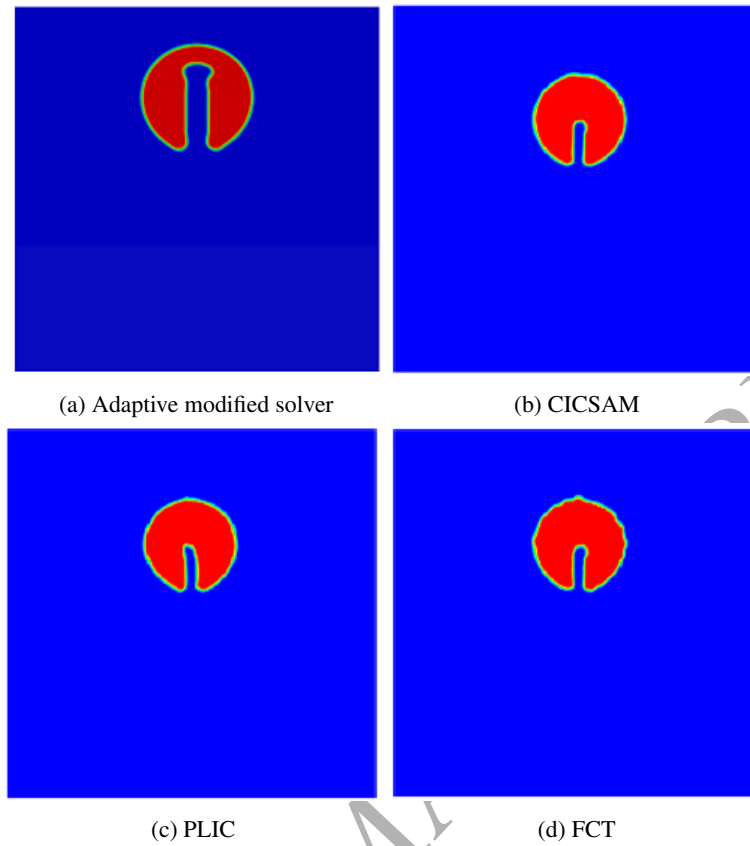


Figure 11: Comparison between the used framework and available method reviewed by Gopala and van Wachem [20]. (a) is showing modified solver with adaptive compressive scheme, (b) is showing the compressive interface capturing scheme for arbitrary meshes (CICSAM), (c) is showing piecewise linear interface construction (PLIC) and (d) is showing flux-corrected transport FCT. All presented in mesh a domain of 200 by 200

531 scheme used by OpenFOAM, the compressive interface capturing scheme for arbitrary meshes (CICSAM)
 532 employed by FLUENT commercial code, the piecewise linear interface construction (PLIC) and the flux-
 533 corrected transport (FCT)). In this test cases, the notched disk was a bit different than what is presented in
 534 the standard Zalesak [51] test case, yet it has the same overall characteristics. Looking at this comparison,
 535 one can relate and compare the overall behaviour for the different solvers as seen in Fig. 11. Nevertheless,
 536 one can spot out the difference in geometrical layout between our test case and the test cases presented in
 537 [19]; the mesh was kept the same as in [20] (200x200). By comparing the results from the developed solver
 538 to those reported in [20], it can be concluded that a good solution has been achieved.

539 4.4. Circle in a vortex field

540 In this section, the solver performance is tested in a vortex flow as presented by Rider and Kothe [42]
 541 and Roenby et al. [34]. The aim of this benchmark test is to verify the ability of the model to deal with
 542 severe interface stretching. The test case includes an initially static circular fluid disk with radius of $R =$
 543 0.15 mm centred at (0.5,0,0.75) in a unit square domain. The disk is subjected to a vortex as shown in Fig.
 544 12. The axis of rotation is located in the centre of the field, and can be described by the following stream
 545 function;

$$u(x, z, t) = \cos((2\pi t)/T)(-\sin^2(\pi x) \sin(2\pi z), \sin(2\pi x) \sin^2(\pi z)) \quad (35)$$

546 where u is the field rotational velocity and T is the period of the flow during rotation. Due to the flow
 547 direction, the disc is stressed into a long thread until time $t = 4s$ forming a spiral shape. The interface
 548 thickness of the deformed disk shape, as well as the numerical diffusion of values located at the tail of the
 549 fluid body during its spiral motion are of interest. The results presented in Fig. 13 and 14 are for three
 550 different grid sizes using the standard (interFoam) and the newly developed adaptive modified solver. On
 551 each figure, the final interface shape is shown with three iso-contours values for the indicator function (α)
 552 of (0.1, 0.5 and 0.9) after one revolution of the disk ($t = 4$ s).

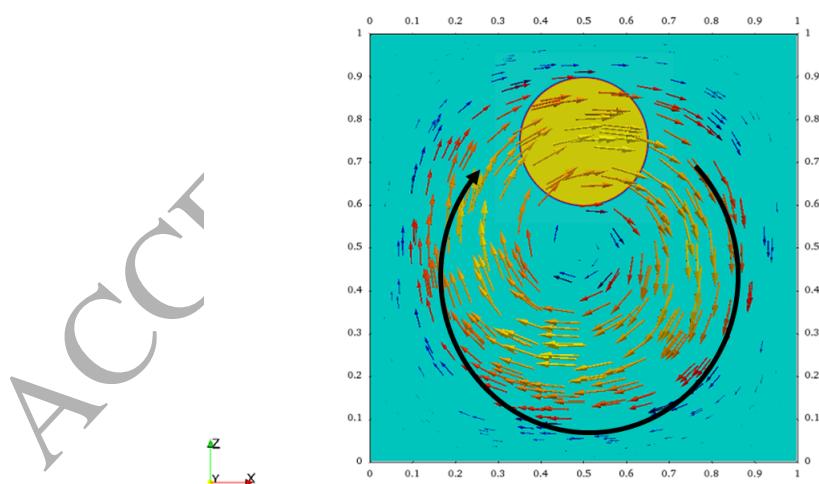


Figure 12: Schematic representation the initial configuration of the shearing flow test with the value of the color function is one inside the circle and zero outside

553 The standard solver failed to capture the full spiral shape after the disk rotation using the coarse mesh

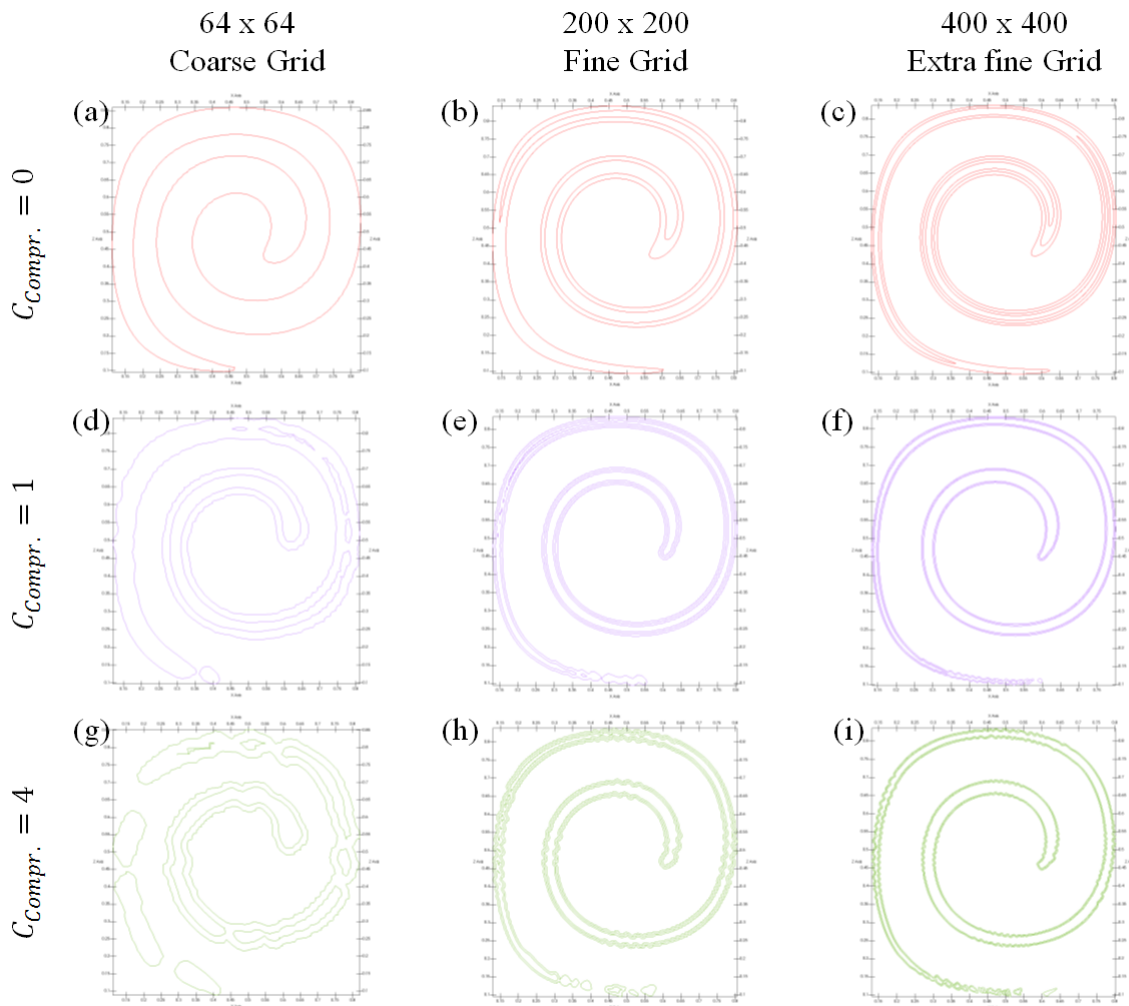


Figure 13: Circle in a vortex field after one revolution. Iso-contours for indicator function alpha ($\alpha = 0.1, 0.5$ and 0.9) is plotted for the standard interFoam using different compression values, together with the reference shape.

554 (see Fig. 13(a,d,g)). Due to the very high diffusion and the absence of compression, iso-contours of 0.1 and
 555 0.5 volume fraction have disappeared from the computational domain (see Fig. 13 (a)). Using the adaptive
 556 modified solver the results are problematic as well especially for the tail as presented in Fig. 14(a,d,g). By
 557 using high sharpening value Fig. 13 (d,g) at low grid resolution to counter balance the numerical diffusion,
 558 tail snap-off at the spiral formation has been observed. Fragmentation or tail snapping off is evident in all
 559 figures.

560 Moving to a finer grid (200x200) the behaviour of the two solvers becomes similar although some differ-
 561 ences can be noticed. The standard solver with no compression Fig. 13(b) suffers from high diffusion as seen

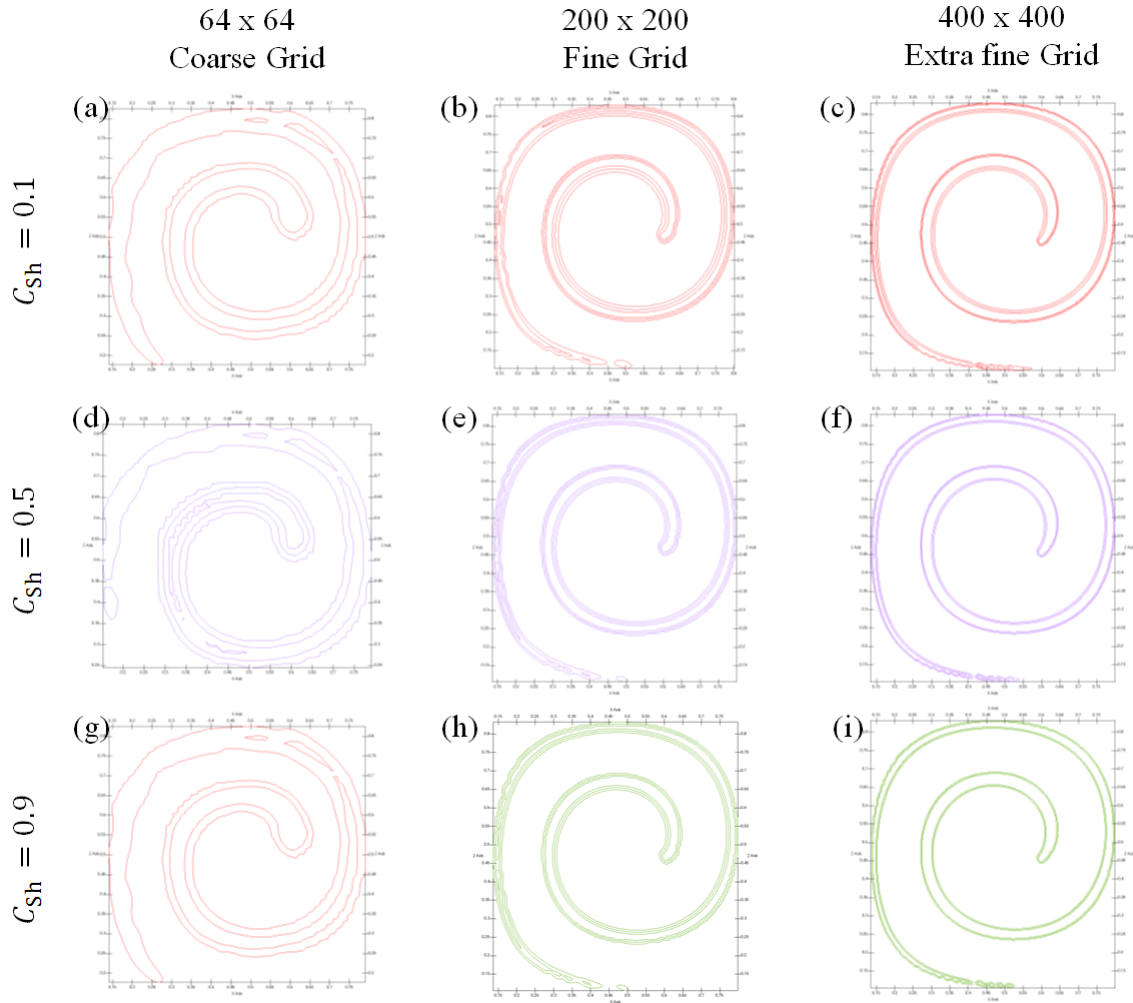


Figure 14: Circle in a vortex field after one revolution. Iso-contours of indicator function alpha sharp ($\alpha_{sh} = 0.1, 0.5$ and 0.9) is plotted for the adaptive modified solver using different sharpening coefficients, together with the reference shape.

562 in the previous test cases where the (0.1) iso-contour disappears. As the compression value increases (see
 563 Fig. 13(e,h)) the standard solver shows early fragmentation at the tail or non-smooth interface. In contrast,
 564 the adaptive solver agrees with the expected spiral shape using different sharpening coefficients. Neverthe-
 565 less, with low sharpening value as shown in Fig. 14(b) early fragmentation with the 0.1 iso-contours lines
 566 loss has been observed. Increasing α_{sh} to values greater than 0.5 (see Fig. 14(e,h)) provides an accurate
 567 spiral shape with minimum phase snapping at the tail. Good agreement using adaptive compression has
 568 been achieved in balancing the swirling tails compared to the wiggly interface appeared using the standard
 569 solver. One can notice that the smallest fragmentation at the spiral tail seems to be unavoidable by using any

570 applied sharpening algorithm, as also discussed by Sato and Ničeno [55] and Malgarinos et al. [26], espe-
571 cially at regions where the liquid body becomes very thin. Fragmentation happens when the local interface
572 curvature becomes comparable to the cell size. At this point, the iso-contours are not able to represent the
573 significant interface curvature inside the cell any more. Iso-contours based on volume fraction advection,
574 leads to errors in the estimate of the fragmented droplet motion similar to those reported by Černe et al.
575 [56] and Roenby et al. [34]. As a final sensitivity test the grid size has been doubled (400x400), to examine
576 the influence of the mesh size on the adaptive solver. Both solvers perform better with this high resolution
577 grid, yet differences have been noticed as with the previous cases. As seen in Fig. 13(c) the standard (in-
578 terFoam) using zero compression coefficient gives a better interface representation with less diffusion and
579 stable tail. By introducing compression (see Fig. 13(f,i)) the spiral shape is maintained, although wiggly
580 shapes emerge near the outer interface. Using the adaptive compression no significant change is noticed; by
581 varying the sharpening value (C_{sh}): as seen in Fig. 14(c,f,i), the results do not change. The results indicate
582 that the balance between sharpening and compression is well achieved. Combining the developed solver
583 with fine grid proves the proposed methodology independent of tuning parameters which is a very desirable
584 feature within multiphase flows. Finally, it had been concluded that even by using medium quality mesh
585 (i.e. 200x200), the adaptive solver can provide satisfying results for a wide range of sharpening coefficients.

586 4.5. Bubble formation at T-junction

587 The previous benchmark cases tested the suitability of the developed model to a range of idealised
588 conditions. No significant topological changes occur and wettability effect is not present. Thus, further
589 validation against experimental data for the case of formation of bubbles in a T-junction has been performed.
590 This is a test case that involves wetting conditions at the wall as well as complex fluid interface topological
591 changes through the breakup and generation of bubbles. The focus is to test the accuracy of our adaptive
592 model in estimating the correct bubble shape and frequency as presented in the experiment of Arias et al.
593 [57]. Full wetting conditions ($\theta = 0^\circ$) at the main tube are used. Moreover, the contact angle imposed on
594 the injection tube (see Fig. 16) has been taken from the corresponding flow images. A constant contact
595 angle of $\theta = 25^\circ$ for the left wall and $\theta = 45^\circ$ for the right wall has been chosen to match the experiments.
596 The connection between the two channels as well as the flow directions and geometrical representation are
597 shown in Fig. 15.

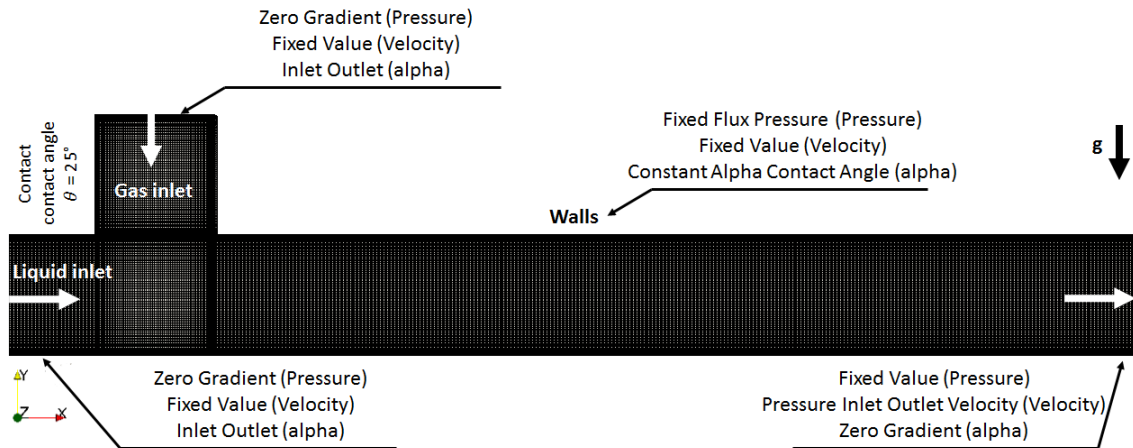


Figure 15: Geometrical model boundaries and overall dimensions

598 Two different operating conditions, summarised in Table 4, have been selected for presentation. The
 599 velocities selected for comparison with our numerical simulations are also shown in table 4. The conditions
 600 used are carefully selected to simulate low capillary number and to show two different bubble size formation
 601 with fluid properties listed in Table .5.

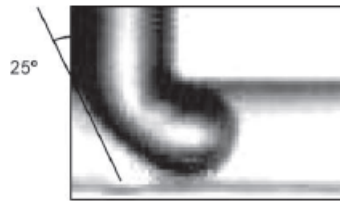


Figure 16: Contact angle at injection tube measured from experimental images

Table 4: Inlet velocities for liquid and gas, dimensionless numbers and regime expected

Case	$U_g(m/s)$	$U_l(m/s)$	$MaxRe$	$MaxWe$	$Exp.Regime$
Case 1	0.242	0.318	32	1.4	<i>Slug</i>
Case 2	0.068	0.531	53	3.92	<i>Bubble</i>

602 For this test case the appearance of spurious numerical currents would create instability during the
 603 bubble formation process. These currents induce unphysical vortices at the interface, destabilising the
 604 simulations and strongly distorting the interface movement. Gravity acceleration constant was $9.8 m/s^2$,
 605 while the values of maximum Weber number ($\frac{\rho DU^2}{\sigma}$) and the maximum Reynolds number ($\frac{\rho DU}{\mu}$) were the

Table 5: Fluid physical properties

	$\rho(Kg/m^3)$	$\nu(m^2/s)$	$\sigma(N/m)$
Water properties at 25°C	1000	1.004×10^{-6}	0.07
Air properties at 25°C	1.2	8.333×10^{-6}	0.07

606 same as in the experiments and shown in table 4.

607 Comparison of the results from the modified solver and the standard solver (interFoam) using different
 608 compression values against the experiments are shown in Figs. 17 and 18. Depending on the inlet velocity
 609 imposed, one should expect to reproduce different bubbles formation.

610 Figure 17 presents the first bubble generation sequence as mentioned in case 1 Table 4. Using the
 611 standard solver, the slug formation is achieved only when adjusting the compression coefficient to the value
 612 of two as seen in Fig. 17d. Even in this case though the detached ligaments of the fluid appear to be more
 613 spherical than what the experiments indicate. Using the comparison value of one the standard solver failed
 614 to predict the interface snap-off as seen in Fig. 17c. In contrast looking at Fig. 17b it is noticed that the
 615 results obtained by the new adaptive model agree very well with the experiments in **terms of both slug**
 616 formation and snap-off time as seen in Fig. 17a. The adaptive framework predicts the interface snap-off
 617 correctly and minimises the overall parasitic currents. Moreover, the standard solver shows a considerable
 618 increase in parasitic velocity near the interface that may reaches eight times the magnitude of the flow
 619 velocity. The new solver achieved low parasitic currents during the snap-off events while maintaining an
 620 accurate sharp interface.

621 Figure 18 presents bubble flow patterns obtained by imposing higher liquid velocity but lower gas
 622 velocity as in case 2 Table 4 in comparison to the previous case. Good agreement in terms of shape and
 623 patterns between experiments and all numerical simulations can be observed regardless of the solver used.
 624 It is worth mentioning though that looking at Figs. 18c, 18d when the standard interFoam solver is used,
 625 bubbles are generated at different frequencies based on the compression coefficient value. By comparing
 626 the two figures to the experimental Fig. 18a one can also notice that the snap-off time is delayed compared
 627 to the experimental results, while in Fig. 18b one can observe that using the developed adaptive solver,
 628 the snap-off time and the bubble generation frequency is matching well with the experiences. According
 629 to the experimental observations, bubble generation results from the breakup of a gas thread that develops

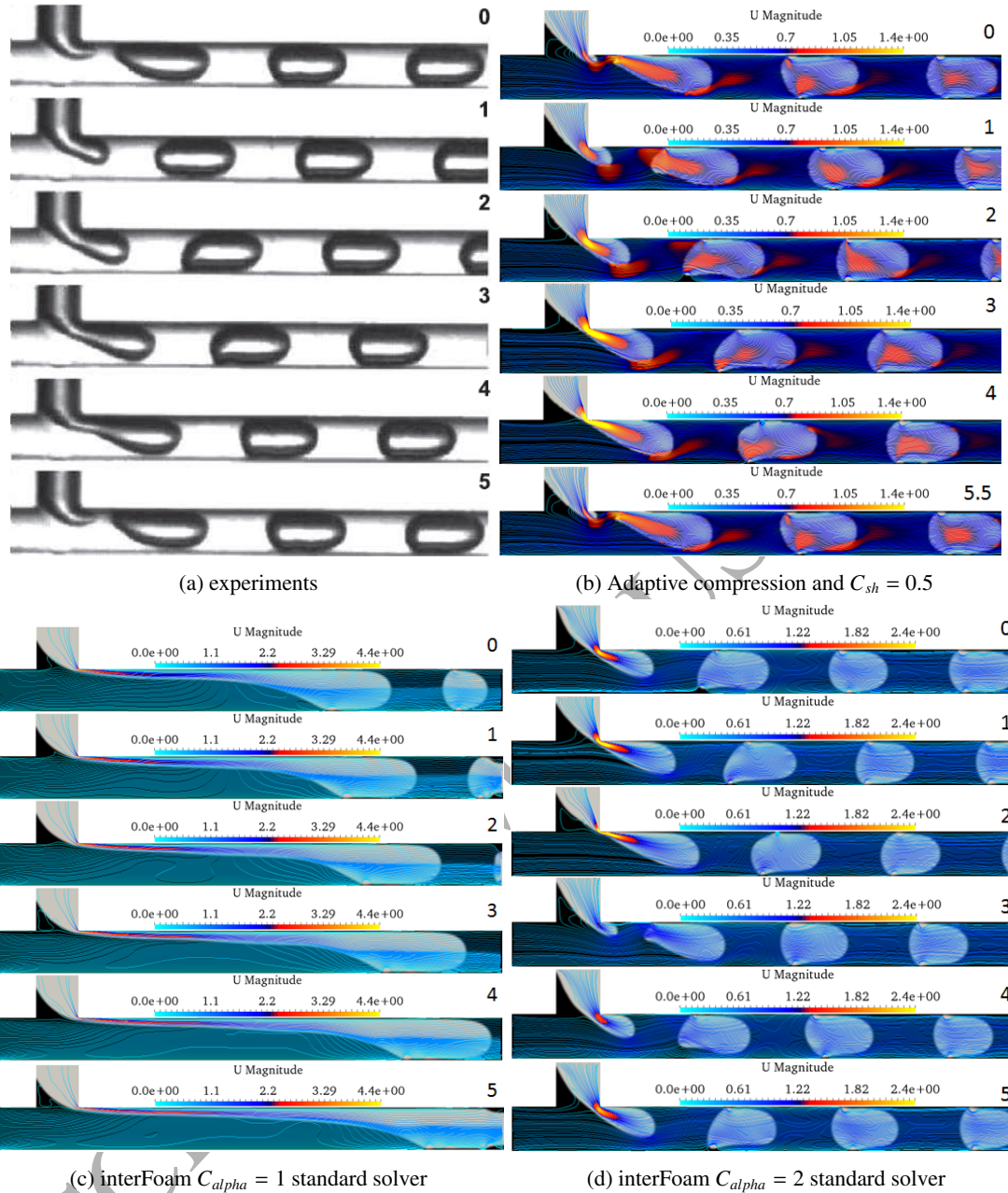


Figure 17: Slug flow, (a) experiments and (b,c,d) numerical simulations. $U_L = 0.318$ m/s and $U_G = 0.242$ m/s. Time (ms) is indicated in the upper right corner. Stream lines are coloured with velocity magnitude in all the figures.

630 after the T- junction. The explanation for the breakup is supported by the Plateau-Rayleigh instability as
 631 discussed by Ménétrier-Deremble and Tabeling [58] or by the effects of the flowing liquid from the tip of
 632 the thread to the neck where pinch-off occurs as presented by van Steijn et al. [59]. The surface tension has
 633 a stabilising effect and opposes any deformation of the interface tending to create a bubble. The snapping

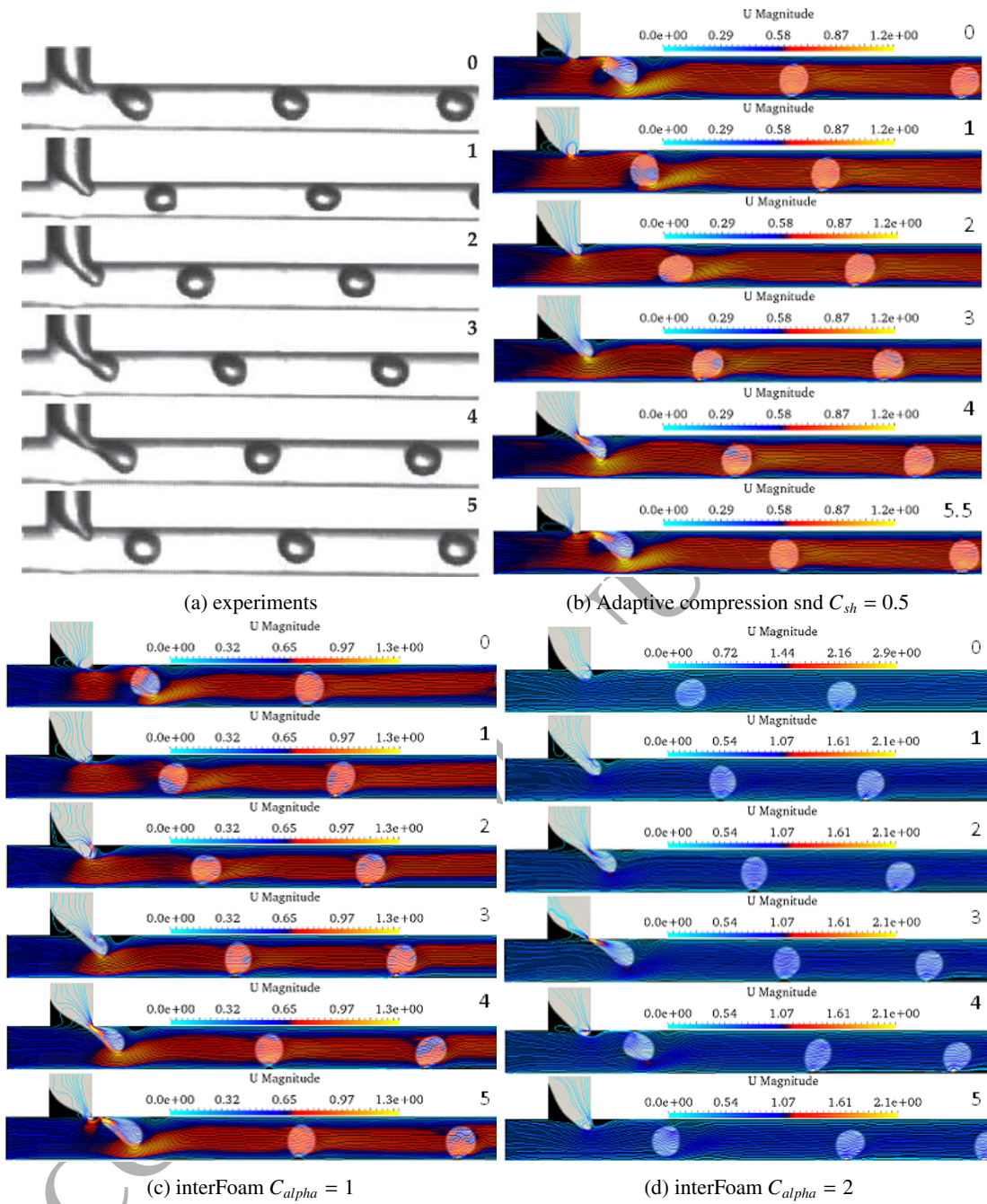


Figure 18: Bubble flow, (a) experiments and (b,c,d) numerical simulations. $U_L = 0.531$ m/s and $U_G = 0.068$ m/s. Time (ms) is indicated in the upper right corner. Stream lines are coloured with velocity magnitude in all the figures.

634 events discussed by the previous literature are in agreement with the simulations presented here, since no
 635 unnatural pinch-off has been observed using the modified solver. On the other hand, a long thread of gas
 636 generated using (interFoam) is clearly seen in Fig. 17c.

Table 6: Error in Bubble generation frequency

	<i>Sim.frequency(Hz)</i>	<i>Error_f</i>
Case 1 (Modified solver)	190.47	4.7 %
Case 1 (interFoam $C_{alpha} = 1$)	210.53	5.2 %
Case 1 (interFoam $C_{alpha} = 2$)	No Bubble generation	100 %
Case 2 (Modified solver)	200.00	1.9 %
Case 2 (interFoam $C_{alpha} = 1$)	184.00	9.8 %
Case 2 (interFoam $C_{alpha} = 2$)	179.21	12.15 %

637 In the previous section a qualitative comparison has been demonstrated using the standard solver and
638 the developed solver against different variation of the control parameters. The validation has been extended
639 to quantitatively compare the bubble generation frequency with experiments. To ensure regularity in the
640 formation of bubbles, a train of bubbles is generated containing at least four of them. The generation
641 frequency was estimated by measuring the time required to create the bubbles. The first bubble of each
642 train, which was strongly dependent on the initial geometry was not considered. We quantify the accuracy
643 of the bubble generation frequency using the following equation:

$$Error_f = \frac{Sim.freq - Exp.freq}{Exp.freq} \quad (36)$$

644 where the *Sim. freq* is the time calculated from the simulations in order to generate one bubble and *Exp.*
645 *freq* is the time needed to produce one bubble in the actual experiment. Table 6 shows the error in the bubble
646 frequency generation compared to the experimental data. For Case (1) although the qualitative results are
647 very close between Fig. 17b and Fig. 17d, one can notice that the developed solver can achieve better
648 accuracy in the in bubble generation frequency. In case (2) the simulation data are qualitatively similar to
649 the experimental results.

650 5. Conclusions

651 A multiphase flow solver for interface capturing at low capillary number flows has been developed
652 and evaluated against well established benchmark cases. Wide range of control parameters of the VoF
653 methodology have been tested, aiming to shed light to their effect on physical properties of micro-scale
654 flows as well as how they interlink. Five different test cases, chosen specifically to highlight the strengths
655 and sensitivity of each model are presented; the best results obtained are summarized in Tables (7,8). The

656 present work was intended to overcome a natural tendency to evaluate numerical methods using only test
657 cases close to the specific application for which they were designed in the first place. In our study a wide
658 range of conditions have been tested, starting from static interfaces (static droplet), and moving to interface
659 smearing (Zalesaks disk, circle in a vortex field) and bubble generation using experimental (T-junction).
660 As it has been demonstrated, although for all the test cases there is a unique optimum set of parameters
661 relevant to sharpening and smoothing part of the method ($U_f\% = 0.05, n = 10, C_{sh} = 0.5$), this is not
662 the case for the C_{compr} term. The results presented here as well as in previous literature studies, indicate
663 that this term is the most versatile coefficient depending on the physical characteristics of the case under
664 consideration as well as the grid size. With the inclusion of adaptive compression this difficulty is waved
665 and an a-priori selection of a value is not required. Even more importantly, it seems that the adaptive nature
666 of the coefficient that controls the interface thickness counter balances the need for very fine grids. The
667 combination of an adaptive compression VoF algorithm and a smoothing technique for the computation
668 of the surface tension has been shown to give accurate results and satisfactory convergence. Advection
669 tests in which interfaces are transported by an assumed external velocity field have been considered while
670 a quantitative comparison with previous literature has been also made. In addition, bubble formation in
671 a liquid flow was simulated by solving the Navier–Stokes equations coupled to the volume fraction field
672 equation in a T-junction configuration for which experimental data are available. From the advection test
673 cases, where the volume fraction equation is solved, the compression method as implemented in the solver
674 interFoam failed to predict the results qualitatively. In contrast, the results obtained with the adaptive
675 modified solver, adhere closely to literature. The used adaptive compression method proved to be mass
676 conserving. In the future work, the proposed method will be used to model multiphase flow using real
677 porous rocks produced from micro-CT images to characterize the effect of wettability on droplet impacting
678 porous media.

Table 7: Benchmark summary highlighting the best set-up for static droplet test cases, along advantages and disadvantages

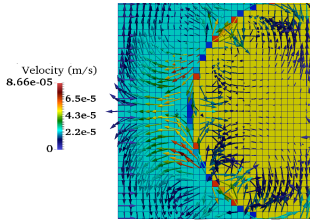
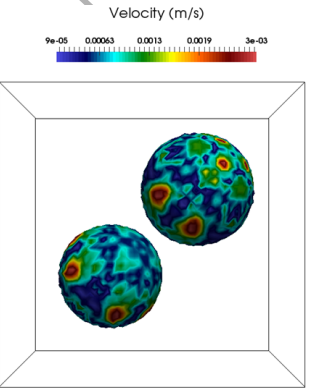
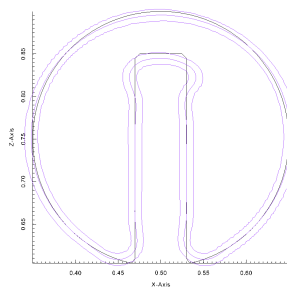
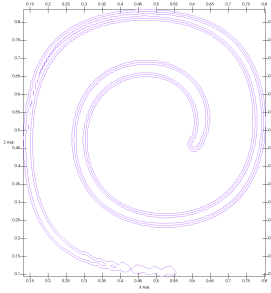
Benchmark	Control parameters	Effective results	Comments
Static Droplet	Case SF1 <ul style="list-style-type: none"> • $U_f\% = 0.05$ • $n = 10$ • $C_{sh} = 0.5$ • $C_{compr.} = 0.5$ 		Advantage <ul style="list-style-type: none"> • Interface presented in one grid cell Disadvantage <ul style="list-style-type: none"> • Sensitive to compression coefficient value (C_{comp} tested 0.5,1,2,3) • Adaptive compression not used
Interacting Parasitic Currents of two relaxing droplets	Case SF1 <ul style="list-style-type: none"> • $U_f\% = 0.05$ • $n = 10$ • $C_{sh} = 0.5$ • $C_{compr.} = 0.5$ 		Advantage <ul style="list-style-type: none"> • Interface presented in one grid cell • Droplets do not merge Disadvantage <ul style="list-style-type: none"> • Sensitive to compression coefficient value (C_{comp} tested 0.5,1,2,3) • Higher parasitic current than one droplet test • Adaptive compression not used

Table 8: Benchmark summary highlighting the best set-up for a typical advection test cases, along advantages and disadvantages

Benchmark	Control parameters	Effective results	Comments
Zalesaks Disk	Fine Grid (200 x 200) <ul style="list-style-type: none"> • $U_f\% = 0.05$ • $n = 10$ • $C_{sh} = 0.5$ • $C_{compr.}$ <i>Adptive</i> =		Advantage <ul style="list-style-type: none"> • Not sensitive to grid size after the 200x200 • Not sensitive to compression value using the adaptive solver Disadvantage <ul style="list-style-type: none"> • By increasing C_{sh}, interface becomes sharper yet not stable for low parasitic current.
Circle in a vortex field	Fine Grid (200 x 200) <ul style="list-style-type: none"> • $U_f\% = 0.05$ • $n = 10$ • $C_{sh} = 0.5$ • $C_{compr.}$ <i>Adptive</i> =		Advantage <ul style="list-style-type: none"> • Increase in accuracy regardless of compression Disadvantage <ul style="list-style-type: none"> • Snapping at tail non avoidable due to grid size effect.

679 6. Acknowledgements

680 The authors would also like to acknowledge the contribution of The Lloyds Register Foundation. Lloyds
681 Register Foundation helps to protect life and property by supporting engineering-related education, public
682 engagement and the application of research. Dr Vogiatzaki would like to acknowledge UKs Engineering
683 and Physical Science Research Council support through the grant EP/P012744/1. Prof. Marengo and Dr.
684 Georgoulas would like to acknowledge UKs Engineering and Physical Science Research Council support
685 through the grant EP/P013112/1.

686 References

- 687 [1] G. M. Whitesides, The origins and the future of microfluidics, *Nature* 442 (2006) 368–373.
- 688 [2] M. Wörner, Numerical modeling of multiphase flows in microfluidics and micro process engineering: a review of methods
689 and applications, *Microfluidics and nanofluidics* 12 (6) (2012) 841–886.
- 690 [3] R. A. Mahdi, H. Mohammed, K. Munisamy, N. Saeid, Review of convection heat transfer and fluid flow in porous media
691 with nanofluid, *Renewable and Sustainable Energy Reviews* 41 (2015) 715–734.
- 692 [4] C. N. Baroud, F. Gallaire, R. Dangla, Dynamics of microfluidic droplets, *Lab on a Chip* 10 (16) (2010) 2032–2045.
- 693 [5] A. K. Yadav, J. C. de la Cal, M. J. Barandiaran, Feasibility of tubular microreactors for emulsion polymerization, *Macro-*
694 *molecular Reaction Engineering* 5 (1) (2011) 69–77.
- 695 [6] M. Andrew, H. Menke, M. J. Blunt, B. Bijeljic, The imaging of dynamic multiphase fluid flow using synchrotron-based x-ray
696 microtomography at reservoir conditions, *Transport in Porous Media* 110 (1) (2015) 1–24.
- 697 [7] S. Osher, J. A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formula-
698 tions, *Journal of computational physics* 79 (1) (1988) 12–49.
- 699 [8] J. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, *Journal of Computational Physics* 100
700 (1992) 335–354.
- 701 [9] M. Sussman, E. Fatemi, P. Smereka, S. Osher, An improved level set method for incompressible two-phase flows, *Computers*
702 *& Fluids* 27 (5) (1998) 663–680.
- 703 [10] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *Journal*
704 *of Computational physics* 114 (1) (1994) 146–159.
- 705 [11] T. Pringuey, R. S. Cant, Robust Conservative Level Set Method for 3D Mixed-Element Meshes Application to LES of Primary
706 Liquid-Sheet Breakup, *Communications in Computational Physics* 16 (02) (2014) 403–439.
- 707 [12] T. Pringuey, R. S. Cant, High order schemes on three-dimensional general polyhedral meshes Application to the level set
708 method, *Communications in Computational Physics* 12 (01) (2012) 1–41.
- 709 [13] D. L. Chopp, Computing minimal surfaces via level set curvature flow, *Journal of Computational Physics* 106 (1) (1993)
710 77–91.

- 711 [14] G. Russo, P. Smereka, A remark on computing distance functions, *Journal of Computational Physics* 163 (1) (2000) 51–67.
- 712 [15] H. Rusche, Computational fluid dynamics of dispersed two-phase flows at high phase fractions, Ph.D. thesis, Imperial College
713 London (University of London), 2003.
- 714 [16] H. G. Weller, G. Tabor, H. Jasak, C. Fureby, A tensorial approach to computational continuum mechanics using object-
715 oriented techniques, *Journal of Computational Physics* 12 (1998) 620–631.
- 716 [17] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows
717 with SURFER, *Journal of Computational Physics* 113 (1) (1994) 134–147.
- 718 [18] O. Ubbink, R. Issa, A method for capturing sharp fluid interfaces on arbitrary meshes, *Journal of Computational Physics*
719 153 (1) (1999) 26–50.
- 720 [19] I. Park, K. Kim, J. Kim, S. Van, A volume-of-fluid method for incompressible free surface flows, *International Journal for*
721 *Numerical Methods in Fluids* 61 (12) (2009) 1331–1362.
- 722 [20] V. R. Gopala, B. G. van Wachem, Volume of fluid methods for immiscible-fluid and free-surface flows, *Chemical Engineering*
723 *Journal* 141 (1) (2008) 204–221.
- 724 [21] C. Wu, D. Young, H. Wu, Simulations of multidimensional interfacial flows by an improved volume-of-fluid method, *Inter-*
725 *national Journal of Heat and Mass Transfer* 60 (2013) 739–755.
- 726 [22] E. Aulisa, S. Manservigi, R. Scardovelli, S. Zaleski, Interface reconstruction with least-squares fit and split advection in
727 three-dimensional Cartesian geometry, *Journal of Computational Physics* 225 (2) (2007) 2301–2319.
- 728 [23] F. Denner, B. G. van Wachem, Compressive VOF method with skewness correction to capture sharp interfaces on arbitrary
729 meshes, *Journal of Computational Physics* 279 (2014) 127–144.
- 730 [24] M. Sussman, E. G. Puckett, A coupled level set and volume-of-fluid method for computing 3D and axisymmetric incom-
731 pressible two-phase flows, *Journal of Computational Physics* 162 (2) (2000) 301–337.
- 732 [25] G. Son, N. Hur, A coupled level set and volume-of-fluid method for the buoyancy-driven motion of fluid particles, *Numerical*
733 *Heat Transfer: Part B: Fundamentals* 42 (6) (2002) 523–542.
- 734 [26] I. Malgarinos, N. Nikolopoulos, M. Gavaises, Coupling a local adaptive grid refinement technique with an interface sharpen-
735 ing scheme for the simulation of two-phase flow and free-surface flows using VOF methodology, *Journal of Computational*
736 *Physics* 300 (2015) 732–753.
- 737 [27] Ž. Tuković, H. Jasak, A moving mesh finite volume interface tracking method for surface tension dominated interfacial fluid
738 flow, *Computers & fluids* 55 (2012) 70–84.
- 739 [28] J. Heyns, A. Malan, T. Harms, O. Oxtoby, Development of a compressive surface capturing formulation for modelling free-
740 surface flow by using the volume-of-fluid approach, *International Journal for Numerical Methods in Fluids* 71 (6) (2013)
741 788–804.
- 742 [29] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, *International Journal for*
743 *Numerical Methods in Fluids* 30 (6) (1999) 775–793.
- 744 [30] G. Tryggvason, R. Scardovelli, S. Zaleski, Direct numerical simulations of gas–liquid multiphase flows, Cambridge Univer-
745 sity Press, 2011.

- 746 [31] C. Bilger, M. Aboukhedr, K. Vogiatzaki, R. Cant, Evaluation of two-phase flow solvers using Level Set and Volume of Fluid
747 methods, *Journal of Computational Physics* .
- 748 [32] B. Van Wachem, A.-E. Almstedt, Methods for multiphase computational fluid dynamics, *Chemical Engineering Journal*
749 96 (1) (2003) 81–98.
- 750 [33] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, M. W. Williams, A balanced-force algorithm
751 for continuous and sharp interfacial surface tension models within a volume tracking framework, *Journal of Computational*
752 *Physics* 213 (1) (2006) 141–173.
- 753 [34] J. Roenby, H. Bredmose, H. Jasak, A Computational Method for Sharp Interface Advection, arXiv preprint arXiv:1601.05392
754 .
- 755 [35] S. Muzafferija, M. Peric, Computation of free-surface flows using interface-tracking and interface-capturing methods, *Ad-*
756 *vanced Fluid Mechanics* 24 (1999) 59–100.
- 757 [36] H. Jasak, H. G. Weller, Interface-tracking capabilities of the InterGamma differencing scheme, Tech. Rep., Imperial College
758 London, 1995.
- 759 [37] Y. Renardy, M. Renardy, PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method, *Journal of*
760 *Computational Physics* 183 (2) (2002) 400–421.
- 761 [38] B. Leonard, The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection, *Computer*
762 *methods in applied mechanics and engineering* 88 (1) (1991) 17–74.
- 763 [39] S. Popinet, An accurate adaptive solver for surface-tension-driven interfacial flows, *Journal of Computational Physics*
764 228 (16) (2009) 5838–5866.
- 765 [40] F. Denner, B. G. van Wachem, Numerical time-step restrictions as a result of capillary waves, *Journal of Computational*
766 *Physics* 285 (2015) 24–40.
- 767 [41] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, *International Journal for*
768 *Numerical Methods in Fluids* 30 (1999) 775–793.
- 769 [42] W. J. Rider, D. B. Kothe, Reconstructing volume tracking, *Journal of computational physics* 141 (2) (1998) 112–152.
- 770 [43] O. OpenCFD, The Open Source CFD Toolbox, User Guide, OpenCFD Ltd .
- 771 [44] R. Issa, Solution of the implicitly discretised fluid flow equations by Operator-Splitting, *Journal of Computational Physics* 62
772 (1985) 40–65.
- 773 [45] J. H. Ferziger, M. Peric, A. Leonard, *Computational methods for fluid dynamics*, 1997.
- 774 [46] A. Georgoulas, P. Koukouvinis, M. Gavaises, M. Marengo, Numerical investigation of quasi-static bubble growth and de-
775 tachment from submerged orifices in isothermal liquid pools: The effect of varying fluid properties and gravity levels, *Inter-*
776 *national Journal of Multiphase Flow* 74 (2015) 59 – 78, doi:http://dx.doi.org/10.1016/j.ijmultiphaseflow.2015.04.008.
- 777 [47] A. Q. Raeini, M. J. Blunt, B. Bijeljic, Modelling two-phase flow in porous media at the pore scale using the volume-of-fluid
778 method, *Journal of Computational Physics* 231 (17) (2012) 5653–5668.
- 779 [48] M. Aboukhedr, M. Gavaises, A. Georgoulas, M. Marengo, K. Vogiatzaki, Numerical Investigation of Droplet Spreading on
780 Porous and Non-porous surfaces, 27th Annual Conference on Liquid Atomization and Spray Systems .

- 781 [49] J. D. Anderson, J. Wendt, Computational fluid dynamics, vol. 206, Springer, 1995.
- 782 [50] H. Weller, Derivation, modelling and solution of the conditionally averaged two-phase flow equations, Nabla Ltd, No Tech-
783 nical Report TR/HGW/02 .
- 784 [51] S. T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, Journal of computational physics 31 (3)
785 (1979) 335–362.
- 786 [52] H. Jasak, H. Weller, Interface tracking capabilities of the inter-gamma differencing scheme, Department of Mechanical
787 Engineering, Imperial College of Science, Technology and Medicine .
- 788 [53] M. Williams, D. Kothe, E. Puckett, Accuracy and convergence of continuum surface tension models, Fluid Dynamics at
789 Interfaces, Cambridge University Press, Cambridge (1998) 294–305.
- 790 [54] R. Scardovelli, S. Zaleski, Title not needed for ILASS format, Annual Review of Fluid Mechanics 31 (1999) 567–603.
- 791 [55] Y. Sato, B. Ničeno, A conservative local interface sharpening scheme for the constrained interpolation profile method, Inter-
792 national Journal for Numerical Methods in Fluids 70 (4) (2012) 441–467.
- 793 [56] G. Černe, S. Petelin, I. Tiselj, Numerical errors of the volume-of-fluid interface tracking algorithm, International journal for
794 numerical methods in fluids 38 (4) (2002) 329–350.
- 795 [57] S. Arias, X. Ruiz, J. Casademunt, L. Ramírez-Piscina, R. González-Cinca, Experimental study of a microchannel bubble
796 injector for microgravity applications, Microgravity Science and Technology 21 (1-2) (2009) 107–111.
- 797 [58] L. Ménétrier-Deremble, P. Tabeling, Droplet breakup in microfluidic junctions of arbitrary angles, Physical Review E 74 (3)
798 (2006) 035303.
- 799 [59] V. van Steijn, C. R. Kleijn, M. T. Kreutzer, Flows around confined bubbles and their importance in triggering pinch-off,
800 Physical review letters 103 (21) (2009) 214501.