An implicit surface tension model for the analysis of droplet dynamics

Alex Jarauta [a], Pavel Ryzhakov [b], Jordi Pons-Prats [b], Marc Secanell [a],*

[a] *Energy Systems Design Laboratory (ESDLab), University of Alberta, Edmonton, Canada*
[b] *Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE), Gran Capitán s/n, 08034 Barcelona, Spain*

A R T I C L E   I N F O

A B S T R A C T

A Lagrangian incompressible fluid flow model is extended by including an implicit surface tension term in order to analyze droplet dynamics. The Lagrangian framework is adopted to model the fluid and track its boundary, and the implicit surface tension term is used to introduce the appropriate forces at the domain boundary. The introduction of the tangent matrix corresponding to the surface tension force term ensures enhanced stability of the derived model. Static, dynamic and sessile droplet examples are simulated to validate the model and evaluate its performance. Numerical results are capable of reproducing the pressure distribution in droplets, and the advancing and receding contact angles evolution for droplets in varying substrates and inclined planes. The model is stable even at time steps up to 20 times larger than previously reported in literature and achieves first and second order convergence in time and space, respectively. The present implicit surface tension implementation is applicable to any model where the interface is represented by a moving boundary mesh.

© 2018 Elsevier Inc. All rights reserved.

1. Introduction

Two-phase flows with strong surface tension effects are found both in nature and engineering. Surface tension effects are dominant in fluids where Capillary, Bond and Weber numbers are small. In this context, one can distinguish two relevant cases: interaction of small liquid drops with air and motion of gas bubbles in liquid. These are important in a wide range of applications such as cavitation in pumps [1], icing in aircraft wings and wind turbines [2], condensation in heat exchangers [3], inkjet printing [4], and droplet shedding in fuel cell channels [5].

From a modeling perspective, all the above-mentioned phenomena pose two key challenges: i) the detection of the interface between the different phases, which (in the transient state) is continuously changing; and ii) the correct representation of flow variable discontinuities across the interface. Fluids with different viscosity result in velocity discontinuities in the direction tangential to the interface. A discontinuous pressure gradient is obtained when there is a density jump across the interface. Non-negligible surface tension leads to a discontinuity in the pressure field. Ultimately, modeling the surface tension force itself is a complex task as the surface tension depends on (and, at the same time, affects) the interface shape, leading to a strongly coupled non-linear problem. Interface tracking and surface tension modeling approaches vary depending on the kinematic framework used to model the problem.

---

* Corresponding author.
  E-mail address: secanell@ualberta.ca (M. Secanell).

Most commonly, two-phase problems are modeled using an Eulerian framework on a fixed mesh. In this case, the interface is usually identified over the fixed grid by a scalar function, i.e., a distance function in level set (LS) methods [6] or a volume fraction in volume of fluid (VOF) methods [7]. The interface, defined by the zero value of the scalar function, cuts the fixed grid elements at arbitrary positions. Piecewise linear interface calculation (PLIC) techniques [8,9] are most commonly used for reconstructing the interface with the VOF method and have been included in commercial codes [10]. In order to represent the discontinuity in the flow variables across the cut elements, shape function enrichment must be introduced [11].

Once the interface is identified, the surface tension is modeled in the vast majority of existing commercial and academic codes, such as ANSYS Fluent [12], GERRIS Flow solver (GFS) [13] or STAR-CCM [10,14], using the so-called continuum surface force (CSF) model [15], where surface tension is evaluated at the historical time step of the transient problem. Such explicit implementation of the surface tension term is simple: the force term evaluated on the basis of the known interface configuration at the previous time step is added to the residual of the linear momentum equation of the problem. Nearly all previous work in literature stated that explicit formulations lead to a capillary time step constraint due to the presence of surface tension effects [15,16]. However, Denner and van Wachem [17] recently suggested that the time step restriction is related to the spatiotemporal sampling of capillary waves.

Few implicit models on fixed grids have also been developed. An implicit formulation for surface tension dominated problems solved with the level set method was developed by Hysing [18]. The formulation was implicit in space, but semi-implicit in time. Using a fixed-grid approach, interface capturing can be particularly challenging, especially in three dimensional examples and with high order schemes. However, the mentioned model allowed using larger time steps than those reported in previous work. An equation for the maximum time step size depending on fluids' properties and element size used was described by Sussman and Ohta [16]. In their study, they relaxed this time step restriction via a volume preserving mean curvature flow for the computation of the surface tension boundary conditions. In general, a model can be fully implicit only if it is able to accurately track the interface. The above-mentioned fixed mesh models can only be semi-implicit in time, and therefore they cannot be as efficient as fully-implicit models for two-phase problems with surface tension effects. For details regarding time step restrictions of semi-implicit schemes the reader is referred to [18].

Mesh-moving Lagrangian and arbitrary Lagrangian Eulerian (ALE) two-phase models have also been proposed. These methods have the advantage of intrinsically tracking the interface as its position is defined by the mesh boundary. The interface is represented by a set of elemental edges (2D) or faces (3D). This allows for a precise representation of the interfacial discontinuities. In order to represent the variable discontinuities at the interface nodes, degrees of freedom are typically duplicated [19].

Several studies in literature proposed semi-implicit and implicit surface tension models using moving meshes. Slikkerveer et al. [20] introduced a two-dimensional finite element (FEM)-based model with an implicit treatment of the surface tension term. In this study, the authors observed that the maximum time step used still had to be reduced as the element size decreased. An implicit variational formulation for the surface tension term was proposed by Saksono and Perić [21]. However, the formulation relied on the assumption of revolution symmetry of the problem. Thus, the developed methodology could not be generalized. An enhanced version of the formulation to solve dynamic problems was presented in [22]. Due to accurate interface tracking, pure Lagrangian formulations can be advantageous for implicit models. However, domain movement may severely distort the interface mesh and therefore special techniques must be developed in order to maintain its correct topology. A novel semi-implicit approach was presented by Schroeder et al. [23] where the flow was solved in a fixed mesh, and the interface was represented by a moving surface mesh. The method was stable with a time step three times bigger than that for an explicit scheme. Zheng et al. [24] recently extended the previous work with an implicit model that used Lagrangian particles for the interface coupled with a fixed mesh (i.e., Marker-And-Cell (MAC) grid). Results showed the stability of the method even for a large time step of the order of 0.25 s.

Recently, a technique combining Lagrangian and Eulerian governing equations for the liquid and the gas phases, respectively, was proposed in references [25] and [26]. The method was extended later for surface tension-dominated problems in [5]. This approach, falling into the category of embedded or immersed boundary methods, allows for accurate tracking of the material interface and natural treatment of the interfacial discontinuities. An Eulerian formulation for the gas phase is the most natural choice for describing vessel-filling fluids (such as gas) with fixed external boundaries. On the other hand, representing the water domain using a moving mesh allows for tracking the air–water interface exactly, without experiencing numerical diffusion and eliminating the use of interface reconstruction methods [27]. The cost of re-meshing the Lagrangian domain is reduced when this domain represents a small fraction of the total computational domain, which is the case of droplet dynamics simulations. Thus, an embedded Eulerian–Lagrangian formulation for surface tension-driven problems is particularly advantageous.

In the present paper, an implicit surface tension model for moving meshes is derived. The implementation of the proposed model in the existing Lagrangian code is simple as it requires to add only an extra term to the elemental tangent matrix (also known as Jacobian, which is the matrix of first-order partial derivatives). The derivation originates from the idea of Hysing's work [18], adapting it to the moving-grid model and leading to a fully-implicit model. The surface tension term is linearized leading to an interfacial Laplacian operator in the tangent matrix of the governing system of equations ensuring stability of the model, even when the time step is several orders of magnitude greater than the one identified as critical for explicit schemes. The present model can be naturally integrated into the embedded two-phase framework previously presented in [5,25,28]. The main difference between the current model and state-of-the-art implicit models, such
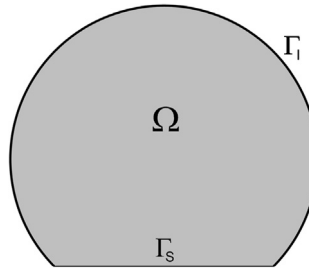
Fig. 1. Schematic representation of the considered single-phase system.

as the model proposed by Zheng et al. [24], is that the solution in the Lagrangian domain does not rely on a background fixed mesh, but on the Lagrangian mesh itself.

In the following sections, the governing equations for the liquid with surface tension effects are described and a finite element model is derived in the residual form paying particular attention to the surface tension term. Then, a mesh-based curvature computation algorithm is presented, the linearization of the surface tension term is derived and corresponding tangent matrix specified. An overall algorithm for the iterative solution of the non-linear problem is outlined. Finally, the model is used to solve several benchmark problems in order to demonstrate the validity of the formulation. The convergence rate of the method is estimated.

2. Numerical model

Let us consider a spatial domain liquid, $\Omega_L \subset \mathbb{R}^n$, where $n \in \{2, 3\}$ (see Fig. 1). The outer boundary of the fluid is designated as Γ_I. In case that the liquid domain is in contact with a solid substrate (e.g., sessile droplet), the part of the boundary in contact with the substrate is denoted by Γ_S.

2.1. Governing equations

The governing equations for the fluid are the momentum and mass conservation equations [29,30]. Since the current work focuses on water droplets, and considering water as an incompressible viscous Newtonian fluid, the governing equations read [10,31]:

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \left(\mathbf{c} \cdot \nabla \right) \mathbf{v} - \nabla \cdot (2\mu D\mathbf{v}) + \nabla p = \rho \mathbf{g} \qquad \text{on } \Omega \tag{1}$$

$$\nabla \cdot \mathbf{v} = 0 \qquad \text{on } \Omega \tag{2}$$

where ρ is the fluid density, $\mathbf{v}$ is velocity, t is time, μ is the fluid dynamic viscosity, $D\mathbf{v} = (\nabla \mathbf{v} + \nabla^T \mathbf{v})/2$ is the strain rate tensor [31], p is pressure, and g is the gravitational acceleration. The convective velocity $\mathbf{c} = \mathbf{v} - \mathbf{v}_m$, where $\mathbf{v}_m$ is the mesh velocity, is the relative velocity between the material and the mesh. Selecting $\mathbf{v} = \mathbf{v}_m$ corresponds to the Lagrangian description (i.e., $\mathbf{c} = 0$) [10,31]:

$$\rho \frac{\partial \mathbf{v}}{\partial t} - \nabla \cdot (2\mu D\mathbf{v}) + \nabla p = \rho \mathbf{g} \qquad \text{on } \Omega \tag{3}$$

$$\nabla \cdot \mathbf{v} = 0 \qquad \text{on } \Omega \tag{4}$$

2.2. Boundary conditions

In order to solve the problem at hand, governing equations (3) and (4) must be complemented with boundary conditions. If not mentioned otherwise we shall consider homogeneous Dirichlet boundary condition at Γ_S:

$$\mathbf{v} = 0 \qquad \text{at } \Gamma_S \tag{5}$$

At boundary Γ_I, the following Neumann condition corresponding to the surface tension is prescribed:

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \gamma \kappa \mathbf{n} \qquad \text{at } \Gamma_I \tag{6}$$

where $\mathbf{n}$ is the unit normal to Γ_I, γ is the surface tension coefficient and κ is the boundary curvature.

Let us consider that the fluid Ω is surrounded by an external fluid in static conditions with relative pressure $p_{\text{ext}} = 0$. Eq. (6) reads that the normal stress across Γ_I is balanced by surface tension. For an incompressible Newtonian fluid, the Cauchy stress tensor is given by:

$$\boldsymbol{\sigma} = -p\mathbf{I} + \mu\left(\nabla\mathbf{v} + \nabla^T\mathbf{v}\right) \tag{7}$$

where $\mathbf{I}$ is the identity tensor. Projecting Eq. (6) onto normal and tangential directions leads to the following conditions:

$$\mathbf{n} \cdot (\boldsymbol{\sigma} \cdot \mathbf{n}) = \gamma\kappa \qquad \text{at } \Gamma_I \tag{8}$$

$$\mathbf{t} \cdot (\boldsymbol{\sigma} \cdot \mathbf{n}) = 0 \qquad \text{at } \Gamma_I \tag{9}$$

Using (7) in the previous equation yields[1]:

$$p - \mu\mathbf{n} \cdot \left(\left[\nabla\mathbf{v} + \nabla^T\mathbf{v}\right] \cdot \mathbf{n}\right) = \gamma\kappa \qquad \text{at } \Gamma_I \tag{10}$$

For static drops, velocities are zero and the projection of viscous stresses onto the normal direction can be neglected. Eq. (10) therefore becomes the Laplace–Young equation:

$$p = \gamma\kappa \qquad \text{at } \Gamma_I \tag{11}$$

The boundary conditions at the contact line (i.e., $\partial\Gamma = \Gamma_S \cap \Gamma_I$) are discussed in Section 2.4.4.

2.3. FEM discretization

Governing equations (3) and (4) are discretized in space using standard mixed FEM with linear interpolation functions for velocity and pressure over 3-noded triangles or 4-noded tetrahedra. Time discretization is performed using the second-order Newmark–Bossak scheme [27]. For the sake of simplicity, the discretized equations are written using Backward Euler method. Neglecting the transposed velocity gradient term leads to a componentwise Laplacian matrix $\mathbf{L}$. This assumption is used to simplify the governing equations. Limache et al. [32] however showed that it is an acceptable assumption for low viscosity fluids, such as water.

After discretizing the governing equations in space and time, the problem can be stated as follows: Given $\bar{\mathbf{v}}_n$ and $\bar{p}_n$ at t_n, find $\bar{\mathbf{v}}_{n+1}$ and $\bar{p}_{n+1}$ at t_{n+1} as the solution of:

$$\mathbf{M}\frac{\bar{\mathbf{v}}_{n+1} - \bar{\mathbf{v}}_n}{\Delta t} + \mu\mathbf{L}\bar{\mathbf{v}}_{n+1} + \mathbf{G}\bar{p}_{n+1} = \bar{\mathbf{F}} + \bar{\mathbf{F}}_{st} \tag{12}$$

$$\mathbf{D}\bar{\mathbf{v}}_{n+1} = 0 \tag{13}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{L}$ is the Laplacian matrix, $\mathbf{G}$ is the gradient matrix, $\mathbf{D}$ is the divergence matrix, $\bar{\mathbf{v}}$ and $\bar{p}$ are the velocity and pressure respectively, $\bar{\mathbf{F}}$ is the body force vector and $\bar{\mathbf{F}}_{st}$ is the surface tension force vector. The matrices are assembled from the elemental contributions defined as:

$$M^{ab} = \rho \int_{\Omega_\mathbf{X}} N^a N^b \, d\Omega_\mathbf{X} = \rho \int_\Omega N^a N^b J(\mathbf{X}) \, d\Omega \tag{14}$$

$$L^{ab} = \int_{\Omega_\mathbf{X}} \frac{\partial N^a}{\partial X_i}\frac{\partial N^b}{\partial X_i} \Omega_\mathbf{X} = \int_\Omega \frac{\partial N^a}{\partial x_i}\frac{\partial N^b}{\partial x_i} J(\mathbf{X}) \, d\Omega \tag{15}$$

$$G_i^{ab} = -\int_{\Omega_\mathbf{X}} \frac{\partial N^a}{\partial X_i} N^b \, d\Omega_\mathbf{X} = -\int_\Omega \frac{\partial N^a}{\partial x_i} N^b J(\mathbf{X}) \, d\Omega \tag{16}$$

$$D_i^{ab} = \int_{\Omega_\mathbf{X}} N^a \frac{\partial N^b}{\partial X_i} \, d\Omega_\mathbf{X} = \int_\Omega N^a \frac{\partial N^b}{\partial x_i} J(\mathbf{X}) \, d\Omega \tag{17}$$

$$f_i^a = \rho \int_{\Omega_\mathbf{X}} N^a g_i \, d\Omega_\mathbf{X} = \rho \int_\Omega N^a g_i J(\mathbf{X}) \, d\Omega \tag{18}$$

$$f_{st,i}^a = -\int_{\Gamma_{I,\mathbf{X}}} \gamma\kappa N^a n_i \, d\Gamma_\mathbf{X} = -\int_{\Gamma_I} \gamma\kappa N^a n_i J_\Gamma(\mathbf{X}) \, d\Gamma \tag{19}$$

where N^a stands for the standard linear FE shape function at node a, and index i refers to spatial components. $\Omega_\mathbf{X}$ is the element integration domain corresponding to the updated configuration, Ω is the element integration domain corresponding

---

[1] In systems where there are surfactants or temperature changes, the surface tension coefficient is variable. However, surface tension gradients are not considered in the present work.
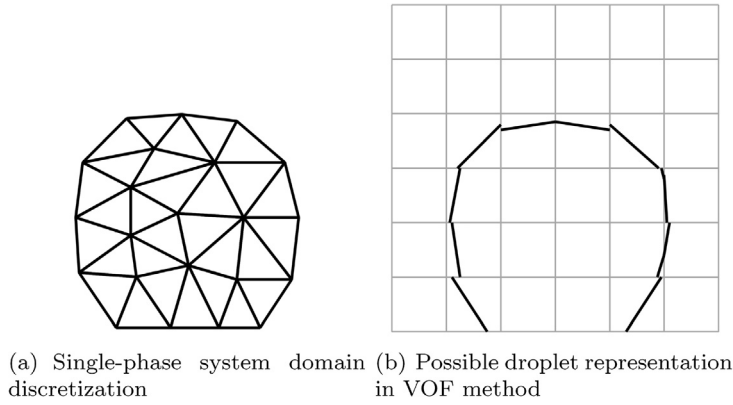
(a) Single-phase system domain discretization   (b) Possible droplet representation in VOF method

Fig. 2. (a) Discretized domain using a moving mesh and (b) Possible discretization using a fixed mesh.

to the reference configuration, and $J(\mathbf{X}) = \mathrm{d}\Omega_{\mathbf{X}}/\mathrm{d}\Omega$ and $J_{\Gamma}(\mathbf{X}) = \mathrm{d}\Gamma_{\mathbf{X}}/\mathrm{d}\Gamma$ are the Jacobians of the transformation between reference and updated configurations. Note that due to the use of an updated Lagrangian framework for the domain, the elemental integration domains in Eqs. (14)–(19) must be recomputed according to the changing mesh configuration.

Discretized computational domain is shown schematically on Fig. 2(a) where a moving mesh is used to represent the liquid domain. If one uses a fixed mesh, an additional technique is required to track or reconstruct the outer boundary Γ_I. A possible representation of the droplet using a fixed mesh is displayed in Fig. 2(b).

The resulting system of equations can be solved using a monolithic scheme, i.e., pressure and velocity are solved simultaneously [33], although any technique not requiring inconsistent pressure boundary condition can be applied [25,34].

The absence of pressure in Eq. (13) is the source of instabilities in the solution for velocity–pressure interpolation pairs that do not fulfill the inf–sup condition [31]. Linear velocity–pressure elements used here do not fulfill this condition and thus must be stabilized. One option relies in relaxing the incompressible condition by adding an extra term that depends on pressure, as shown in [27]. Another option is to add stabilization terms depending on mesh size and time step. Different methods have been presented in literature, such as Galerkin/least squares (GLS) [35], algebraic sub-grid scales (ASGS) [36], orthogonal sub-scales (OSS) [37] and finite increment calculus (FIC) [38]. In this work, the ASGS stabilization technique is implemented because it is a symmetric stabilization. The stabilized governing equations read:

$$\left(\mathbf{M}\frac{1}{\Delta t} + \mu\mathbf{L} + \mathbf{S}_K\right)\bar{\mathbf{v}}_{n+1} + \mathbf{G}\bar{p}_{n+1} = \bar{\mathbf{F}} + \bar{\mathbf{F}}_{\mathrm{st}} + \mathbf{M}\frac{\bar{\mathbf{v}}_n}{\Delta t} \tag{20}$$

$$(\mathbf{D} + \mathbf{S}_D)\,\bar{\mathbf{v}}_{n+1} + \mathbf{S}_L\bar{p}_{n+1} = \bar{\mathbf{F}}_q \tag{21}$$

where stabilization matrices are:

$$S_K^{ab} = \int\limits_{\Omega} \tau_2 \frac{\partial N^a}{\partial x_i}\frac{\partial N^b}{\partial x_i} J(\mathbf{X})\,\mathrm{d}\Omega \tag{22}$$

$$S_D^{ab} = \int\limits_{\Omega} \frac{\rho}{\Delta t}\tau_1 \frac{\partial N^a}{\partial x_i} N^b J(\mathbf{X})\,\mathrm{d}\Omega \tag{23}$$

$$S_L^{ab} = \int\limits_{\Omega} \tau_1 \frac{\partial N^a}{\partial x_i}\frac{\partial N^b}{\partial x_i} J(\mathbf{X})\,\mathrm{d}\Omega \tag{24}$$

$$f_q^a = \int\limits_{\Omega} \rho g_i \frac{\partial N^a}{\partial x_i}\left(\frac{\rho}{\Delta t}N^a + N^a\right) J(\mathbf{X})\,\mathrm{d}\Omega \tag{25}$$

Note that matrix $\mathbf{S}_L$ can be interpreted as a Laplacian term. The presence of this term is particularly important for the computational efficiency of the method [30].

Although the nonlinear convective term is absent in Eq. (20), the system of governing equations is still nonlinear. This is caused by the definition of these equations in terms of the unknown configuration $\mathbf{X}_{n+1}$, defined as:

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \Delta t\,\mathbf{v}_{n+1} \tag{26}$$

Therefore, the discrete operators defined by Eqs. (14)–(19), and (22)–(25) depend on the unknown nodal position. An option often encountered in numerical studies is to solve the nonlinear system of equations using a Newton–Raphson

method since it provides second-order convergence of the nonlinear iterative procedure. This can be achieved by expressing the governing equations in residual form first:

$$\bar{\mathbf{r}}_m = \bar{\mathbf{F}} + \bar{\mathbf{F}}_{st} - \mathbf{M}\frac{\bar{\mathbf{v}}_{n+1} - \bar{\mathbf{v}}_n}{\Delta t} - (\mu \mathbf{L} + \mathbf{S}_K)\bar{\mathbf{v}}_{n+1} - \mathbf{G}\bar{p}_{n+1} = 0 \tag{27}$$

$$\bar{\mathbf{r}}_c = \bar{\mathbf{F}}_q - (\mathbf{D} + \mathbf{S}_D)\bar{\mathbf{v}}_{n+1} - \mathbf{S}_L \bar{p}_{n+1} = 0 \tag{28}$$

Then, expanding the residual using a Taylor series and neglecting terms of second order and higher, the linearized form of the equation to solve is:

$$-\begin{pmatrix} \frac{\partial \bar{\mathbf{r}}_m}{\partial \bar{\mathbf{v}}^k} & \frac{\partial \bar{\mathbf{r}}_m}{\partial \bar{p}^k} \\ \frac{\partial \bar{\mathbf{r}}_c}{\partial \bar{\mathbf{v}}^k} & \frac{\partial \bar{\mathbf{r}}_c}{\partial \bar{p}^k} \end{pmatrix} \begin{pmatrix} d\bar{\mathbf{v}} \\ d\bar{p} \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{r}}_m \\ \bar{\mathbf{r}}_c \end{pmatrix} \tag{29}$$

where $d\bar{\mathbf{v}} = \bar{\mathbf{v}}_{n+1}^{k+1} - \bar{\mathbf{v}}_{n+1}^k$ and $d\bar{p} = \bar{p}_{n+1}^{k+1} - \bar{p}_{n+1}^k$. Index k denotes the nonlinear iteration. Similarly to Eq. (26), the unknown domain configuration at each nonlinear iteration k is computed as $\mathbf{X}_{n+1}^{k+1} = \mathbf{X}_{n+1}^k + \Delta t \, d\mathbf{v}$. Derivatives of the residuals with respect to velocity and pressure are easily obtained, and the system to solve reads:

$$\begin{pmatrix} \mathbf{M}\frac{1}{\Delta t} + \mu \mathbf{L} + \mathbf{S}_K + \mathbf{H}_{ST} & \mathbf{G} \\ \mathbf{D} + \mathbf{S}_D & \mathbf{S}_L \end{pmatrix} \begin{pmatrix} d\bar{\mathbf{v}} \\ d\bar{p} \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{r}}_m\left(\bar{\mathbf{v}}^k, \bar{p}^k\right) \\ \bar{\mathbf{r}}_c\left(\bar{\mathbf{v}}^k, \bar{p}^k\right) \end{pmatrix} \tag{30}$$

The derivatives of the matrix operators with respect to the solution at each nonlinear iteration are assumed to be negligible. Therefore, the current method can be classified as quasi-Newton. Matrix $\mathbf{H}_{ST}$ is the result of linearizing the surface tension term with respect to velocity. Including this term is necessary to overcome time step restrictions that appear in problems where surface tension effects are present[2] [5,15,16,18]. The derivation of this term is carried out next.

2.4. Surface tension term

The force term $\bar{\mathbf{F}}_{st}$ in Eq. (12) is the surface tension force, corresponding to the projection of the Cauchy stress tensor onto the normal direction at the domain boundary Γ_I:

$$\bar{\mathbf{F}}_{st} = -\int_{\Gamma} \gamma \kappa \mathbf{n} \cdot \mathbf{w} \, d\Gamma \tag{31}$$

where γ is the surface tension coefficient, κ is the curvature of the boundary and $\mathbf{n}$ is the unit normal vector to the boundary Γ. The negative sign in Eq. (31) means that the surface tension force is a vector pointing inwards Ω when Γ_I is convex, and it points outwards Ω when the boundary is concave.

2.4.1. Curvature

To compute the surface tension, one must evaluate the curvature. In two dimensions, the curvature can be computed by a simple difference scheme (see [5]), whereas in three dimensions, Meyer's model [39] has been adopted here since it is known to provide most accurate approximation in comparison with the other available approaches [40]. According to Meyer's model, the mean curvature value, κ, of a given node is obtained using the following expression:

$$\kappa = \frac{1}{2}\left\|\mathbf{K}(\mathbf{x}^a)\right\| \tag{32}$$

where $\mathbf{K}(\mathbf{x}^a)$ is the mean curvature normal operator at node a:

$$\mathbf{K}(\mathbf{x}^a) = \frac{1}{2A_M}\sum_{b \in N_1(a)}\left(\cot\alpha^{ab} + \cot\beta^{ab}\right)\left(\mathbf{x}^a - \mathbf{x}^b\right) \tag{33}$$

where α^{ab} and β^{ab} are the two angles opposite to edge $\overline{\mathbf{x}^a\mathbf{x}^b}$ (Fig. 3), $N_1(a)$ denotes the 1-ring neighborhood of node $\mathbf{x}^a$ and A_M is the Voronoi area associated to this node [39]. The normal vector $\mathbf{n}$ is simply obtained by normalizing the vector $\mathbf{K}(\mathbf{x}^a)$ in Eq. (33). The sign of κ is determined by the direction of vector $\mathbf{n}$, being positive if the normal points outwards, and negative if it points inwards.

---

[2] Considering $\mathbf{H}_{ST} = 0$ corresponds to methods where surface tension is integrated explicitly.
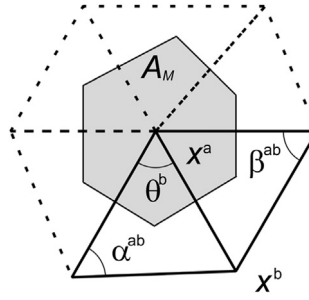
Fig. 3. 1-ring neighborhood of node $\mathbf{x}^a$ with angles opposite to an edge, and Voronoi area A_M associated to this node.

2.4.2. Implicit treatment of the surface tension term

Knowing the domain configuration at $t = t_n$, and seeking the solution at t_{n+1}, an explicit treatment of the surface tension term implies that Eq. (31) is obtained using the known domain configuration $\mathbf{X}_n$.[3] In this article, to avoid excessive time step restrictions faced by explicit schemes [5], the surface tension term is treated implicitly by expressing it as a function of the unknown configuration $\mathbf{X}_{n+1}$:

$$\left(\bar{\mathbf{F}}_{\text{st}}\right)_{n+1} = -\int_{\Gamma_{n+1}} \gamma \kappa_{n+1} \mathbf{n}_{n+1} \cdot \mathbf{w} \, d\Gamma \tag{34}$$

where κ_{n+1} and $\mathbf{n}_{n+1}$ denote the curvature and normal vector obtained using the unknown domain configuration $\mathbf{X}_{n+1}$.

Let us define the identity map on an arbitrary surface Γ:

$$\text{id}_\Gamma (\mathbf{x}) = \mathbf{x} \qquad \forall \mathbf{x} \in \Gamma \tag{35}$$

Then, the following properties can be defined [18,41,42]:

$$\nabla_s \text{id}_\Gamma = \mathbf{I} - \mathbf{n} \otimes \mathbf{n} \tag{36}$$

$$\Delta_s \text{id}_\Gamma = \nabla_s \cdot (\nabla_s \text{id}_\Gamma) = -\nabla_s \cdot (\mathbf{n} \otimes \mathbf{n}) = \kappa \mathbf{n} \tag{37}$$

where ∇_s is the surface gradient operator (i.e., the conventional gradient without the component normal to the surface), and Δ_s is the Laplace–Beltrami operator.

Using Eq. (37) in (34), yields:

$$\left(\bar{\mathbf{F}}_{\text{st}}\right)_{n+1} = -\int_{\Gamma_{n+1}} \gamma \kappa_{n+1} \mathbf{n}_{n+1} \cdot \mathbf{w} \, d\Gamma = -\int_{\Gamma_{n+1}} \gamma \left(\Delta_s \text{id}_\Gamma\right)_{n+1} \cdot \mathbf{w} \, d\Gamma \tag{38}$$

Integrating by parts and applying the surface divergence theorem [18,20–22,43], Eq. (38) reads:

$$\left(\bar{\mathbf{F}}_{\text{st}}\right)_{n+1} = -\int_{\Gamma_{n+1}} \gamma \left(\Delta_s \text{id}_\Gamma\right)_{n+1} \cdot \mathbf{w} \, d\Gamma$$

$$= -\int_{\partial\Gamma_{n+1}} \gamma \mathbf{m}_{n+1} \cdot \mathbf{w} \, d\left(\partial\Gamma\right) + \int_{\Gamma_{n+1}} \gamma \left(\nabla_s \text{id}_\Gamma\right)_{n+1} \cdot \nabla\mathbf{w} \, d\Gamma \tag{39}$$

where $\partial\Gamma$ is the boundary of Γ, and $\mathbf{m}$ is the normal of the boundary $\partial\Gamma$, perpendicular to $\mathbf{n}$ and $\mathbf{ds}$ (unit vector tangent to $\partial\Gamma$), as shown in Fig. 4.

Note that Eq. (39) is evaluated over boundary Γ_{n+1}. A major drawback of Eulerian formulations is that boundary Γ has to be explicitly found. This task can be rather complex for high order time integration schemes, specially in three dimensions, leading to significant errors [18]. The Lagrangian formulation does not have this disadvantage since the position of the boundary is defined by the deforming mesh.

Considering that the identity map is equal to the unknown configuration $\mathbf{X}_{n+1}^k$, it can be updated in every non-linear iteration (Eq. (29)) as follows:

$$(\text{id}_\Gamma)_{n+1}^{k+1} = (\text{id}_\Gamma)_{n+1}^k + \Delta t \, d\bar{\mathbf{v}} \tag{40}$$

---

[3] For sake of clarity, the Jacobians of the transformations between $\mathbf{x}$ and $\mathbf{X}$ are omitted in this section. However, this transformation is performed as shown in Eqs. (14)–(19).
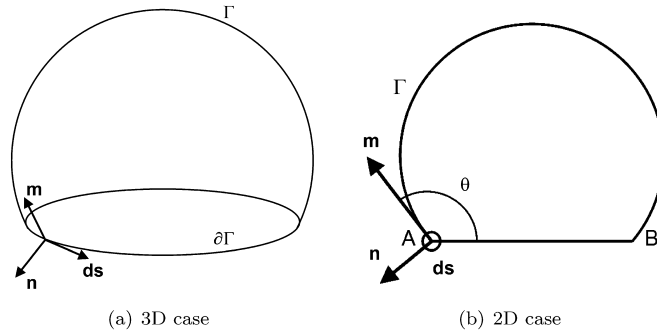
(a) 3D case   (b) 2D case

Fig. 4. Interface domain, boundary and normal vector to the boundary.

This term depends on the variable of interest and can therefore be substituted in Eq. (39):

$$\int_{\Gamma_{n+1}^{k+1}} \gamma \, (\nabla_s \mathrm{id}_\Gamma)_{n+1}^{k+1} \cdot \nabla \mathbf{w} \, \mathrm{d}\Gamma = \int_{\Gamma_{n+1}^{k}} \gamma \, (\nabla_s \mathrm{id}_\Gamma)_{n+1}^{k} \cdot \nabla \mathbf{w} \, \mathrm{d}\Gamma + \Delta t \int_{\Gamma_{n+1}^{k}} \gamma \nabla_s \, \mathrm{d}\bar{\mathbf{v}} \cdot \nabla \mathbf{w} \, \mathrm{d}\Gamma \tag{41}$$

The second term on the right-hand side in Eq. (41) is a velocity Laplacian that may be interpreted as a diffusion term added to the interfacial nodes in the tangential direction. This term is responsible for stabilizing the surface tension effects because it adds viscous dissipation at the interface. Note also that this term is multiplied by the time step, and therefore it increases with step size. Other surface tension models in literature, such as those of Slikkerveer et al. [20], Hysing [18], Saksono and Peric [21,22], Raessi et al. [44], and Sussman and Ohta [16], also overcome the time step restriction using this term, as recently shown by Denner et al. [45]. In this work, the Laplacian term is precisely a consequence of the linearization of the surface tension term in the implicit setting.

Since vector $\mathbf{m}_{n+1}$ also depends on the unknown configuration, the same procedure is applied to the first term in Eq. (39) for its linearization, given that:

$$\mathbf{m}_{n+1}^{k+1} \approx \mathbf{m}_{n+1}^{k} + \Delta t \, d\bar{\mathbf{v}} \tag{42}$$

Note that Eqs. (40) and (42) are based on the same assumptions and therefore they are used to update the identity map and vector $\mathbf{m}$ at each nonlinear iteration [18]. Thus, the first term in Eq. (39) yields:

$$\int_{\partial\Gamma_{n+1}^{k+1}} \gamma \mathbf{m}_{n+1}^{k+1} \cdot \mathbf{w} \, \mathrm{d}\,(\partial\Gamma) = \int_{\partial\Gamma_{n+1}^{k}} \gamma \mathbf{m}_{n+1}^{k} \cdot \mathbf{w} \, \mathrm{d}\,(\partial\Gamma) + \Delta t \int_{\partial\Gamma_{n+1}^{k}} \gamma \, \mathrm{d}\bar{\mathbf{v}} \cdot \mathbf{w} \, \mathrm{d}\,(\partial\Gamma) \tag{43}$$

Thus, the surface tension contribution that must be added to the left-hand side (H_{ST}^{ab}) of the momentum equation (Eq. (29)) is given by the second terms in Eqs. (41) and (43), respectively [42]:

$$H_{ST}^{ab} = -\Delta t \int_{\partial\Gamma_{n+1}} \gamma N^a N^b \, \mathrm{d}\,(\partial\Gamma) + \Delta t \int_{\Gamma_{n+1}} \gamma \frac{\partial N^a}{\partial x_i} \left(\delta_{ij} - n_i n_j \right) \frac{\partial N^b}{\partial x_j} \, \mathrm{d}\Gamma \tag{44}$$
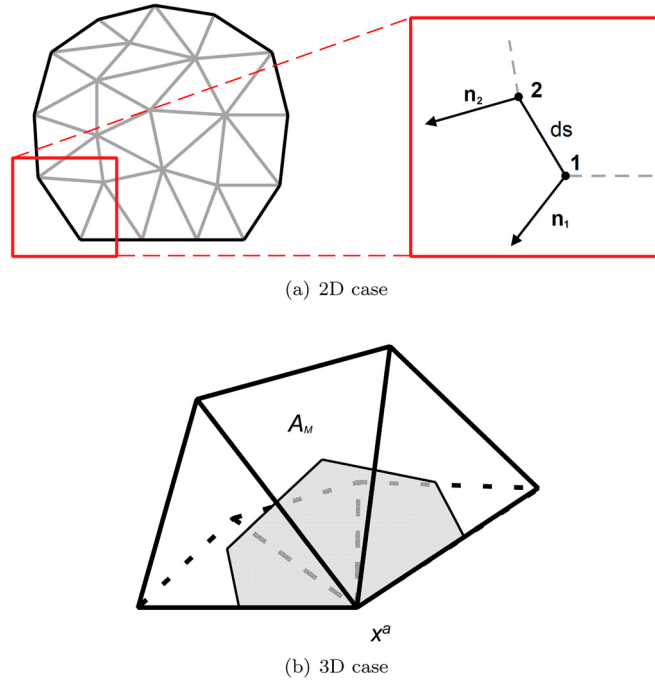
The surface tension contribution in the right-hand side term ($f_{st,i}^a$) is obtained from the first term in Eqs. (41) and (43):

$$\left(\bar{\mathbf{F}}_{st} \right)_{n+1}^{k} = - \int_{\partial\Gamma_{n+1}^{k}} \gamma \mathbf{m}_{n+1}^{k} \cdot \mathbf{w} \, \mathrm{d}\,(\partial\Gamma) + \int_{\Gamma_{n+1}^{k}} \gamma \, (\nabla_s \mathrm{id}_\Gamma)_{n+1}^{k} \cdot \nabla \mathbf{w} \, \mathrm{d}\Gamma \tag{45}$$

Using Eqs. (38) and (39), the right-hand term can be rewritten as follows:

$$\begin{aligned}
\left(\bar{\mathbf{F}}_{st} \right)_{n+1}^{k} &= - \int_{\partial\Gamma_{n+1}^{k}} \gamma \mathbf{m}_{n+1}^{k} \cdot \mathbf{w} \, \mathrm{d}\,(\partial\Gamma) + \int_{\Gamma_{n+1}^{k}} \gamma \, (\nabla_s \mathrm{id}_\Gamma)_{n+1}^{k} \cdot \nabla \mathbf{w} \, \mathrm{d}\Gamma \\
&= - \int_{\Gamma_{n+1}^{k}} \gamma \, (\Delta_s \mathrm{id}_\Gamma)_{n+1}^{k} \cdot \mathbf{w} \, \mathrm{d}\Gamma \\
&= - \int_{\Gamma_{n+1}^{k}} \gamma \kappa_{n+1}^{k} \mathbf{n}_{n+1}^{k} \cdot \mathbf{w} \, \mathrm{d}\Gamma
\end{aligned} \tag{46}$$

(a) 2D case

(b) 3D case

Fig. 5. (a) Normal vector at contact line node and its nearest neighbor to obtain the curvature at contact line in 2D and (b) Voronoi area for a contact line node in 3D.

or alternatively in index notation:

$$f^a_{st,i} = - \int_{\Gamma_{n+1}} \gamma \kappa_{n+1} N^a n_i \, d\Gamma \tag{47}$$

Eq. (47) is the surface tension contribution term added on the right-hand side of the momentum equation (Eq. (29)) in the present model.

2.4.3. Semi-implicit variant of the method

The current method allows for the semi-implicit treatment of surface tension by simply removing the velocity Laplacian $\mathbf{H}_{ST}$ from the tangent matrix in Eq. (29). Therefore, the only surface tension term added to the right-hand side vector in Eq. (29) is the expression written in Eq. (47). In this case, all terms depending on the unknown configuration (i.e., curvature, κ, and normal vector, $\mathbf{n}$) are only updated at the end of each nonlinear iteration.

2.4.4. Curvature at contact line and dynamic contact angle condition

According to Buscaglia and Ausas [42], the interfacial energy at Γ_S (Fig. 1) due to solid/fluid interaction is constant and therefore the surface tension term in this region vanishes. The surface tension term along the contact line, $\partial\Gamma$ (Fig. 4), however is non-zero and requires special treatment. At the contact line, the surface tension term must be applied using the normal vector corresponding to the static equilibrium configuration, $\mathbf{n}_{eq}$, instead of the actual normal vector $\mathbf{n}$ [5], because at steady state both normals must coincide. The curvature at the nodes that represent the contact line is computed using the average normal vector[4] at the contact line ($\mathbf{n}_1$ in Fig. 5(a)) and the normal vector at its nearest neighbor node from Γ_I (node 2 in Fig. 5(a)). The curvature at the contact line (in 2D) is:

$$\kappa_1 = \left\| \frac{d\mathbf{n}}{ds} \right\| = \frac{\|\mathbf{n}_1 - \mathbf{n}_2\|}{ds} \tag{48}$$

---

[4] The normal vector at the contact line is computed as an average of its neighbor nodes:

$$\mathbf{n} = \frac{\sum_i \mathbf{n}_i}{N}.$$

where ds is the distance between nodes 1 and 2. The same idea is applied for the computation of the curvature in 3D. The computation of the curvature at the nodes that belong to the contact line is obtained using Eq. (32). However, the nodes that are in contact with the substrate and do not belong to the contact line are excluded from the 1-ring neighborhood (see Fig. 5(b)).

For the special case of rough surfaces, the concept of *static contact angle* cannot be used [46]. On rough surfaces, the contact line pins and the contact angle changes from one equilibrium configuration to another. The present work uses two threshold values, $\theta_{\min}$ and $\theta_{\max}$, as contact angle conditions. The contact line is fixed only within the range $\theta \in [\theta_{\min}, \theta_{\max}]$. These maximum and minimum values represent the measured contact angles for incipient motion when the droplet is placed on a tilted plane of a given material [5]. Note also that the contact angle might not correspond to that of the material due to surface roughness as described by Wenzel and Cassie–Baxter models [47]. When the dynamic contact angle is not fulfilled (i.e., $\theta < \theta_{\min}$ or $\theta > \theta_{\max}$), the contact line is allowed to move. In this case, instead of a no-slip boundary condition (Eq. (5)), a slip boundary condition is applied at the contact line nodes:

$$\mathbf{v} \cdot \mathbf{n} = 0 \tag{49}$$

where $\mathbf{n}$ is the unit normal vector of the substrate. More details on the dynamic contact angle condition for rough surfaces can be found in [5].

2.5. Solution algorithm

The solution of the problem solved in the Lagrangian domain is summarized next. Given the solution at t_n, $\bar{\mathbf{v}}_n$ and $\bar{p}_n$, at the known configuration $\mathbf{X}_n$, the procedure to find the solution at time t_{n+1} is implemented in Algorithm 1.

Algorithm 1: Simulation of droplet dynamics using a Lagrangian formulation.

1   **for** $t = t_{n+1}$ **do**
2   Identify the boundary of the Lagrangian mesh;
3   Current configuration is the known configuration: $\mathbf{X}_{n+1}^k = \mathbf{X}_n$;
4   **for** <u>nonlinear iteration k</u> **do**
5   Obtain curvature at $\mathbf{X}_{n+1}^k$;
6   Update discrete operators in Eqs. (14)–(18) and (22)–(25);
7   Compute $f_{\mathrm{st},i}^a$ using Eq. (47) and add to RHS in Eq. (29);
8   Compute H_{ST}^{ab} using Eq. (44) and add to LHS in Eq. (29);
9   Solve Eq. (29) for $d\bar{\mathbf{v}}$ and $d\bar{p}$;
10   Update velocity $\bar{\mathbf{v}}_{n+1}^{k+1} = \bar{\mathbf{v}}_{n+1}^k + d\bar{\mathbf{v}}$ and pressure $\bar{p}_{n+1}^{k+1} = \bar{p}_{n+1}^k + d\bar{p}$;
11   Update configuration: $\mathbf{X}_{n+1}^{k+1} = \mathbf{X}_{n+1}^k + \Delta t \cdot d\bar{\mathbf{v}}$;
12   Remesh;
13   **end**
14      $\mathbf{X}_{n+1} = \mathbf{X}_n + \Delta t \cdot \bar{\mathbf{v}}_{n+1}$;
15 **end**

The process of remeshing may need to be performed since interior nodes move according to equation $\mathbf{X}_{n+1}^{k+1} = \mathbf{X}_{n+1}^k + \Delta t \cdot d\bar{\mathbf{v}}$. A boundary-preserving remeshing is implemented, which means that nodal connectivities are generally not changed at the boundary. In case of boundary mesh stretching, local refinement is applied. When considering the effects of the surrounding gas phase is essential, the solution procedure is discussed in reference [25].

2.6. Implementation

The method was implemented within *Kratos Multi-Physics*, a C++ object oriented FE framework [48]. A quasi-Newton method is used to solve Eqs. (27) and (28). The resulting system of equations is solved using the bi-conjugate gradient stabilized method (BICGSTAB). This method is known for its ability to solve non-symmetric matrices with improved rate of convergence and low computational cost when compared to other iterative methods, such as conjugate gradient squared (CGS), biconjugate gradient (Bi-CG) or generalized minimum residual (GMRES) [49].

2.7. Input parameters

Table 1 shows the solver parameters used in the examples below, such as absolute and relative tolerances for pressure and velocity and maximum number of iterations.

Results were obtained using an Ubuntu 14.04 box with an Intel® Core™ i7 CPU 4750HQ @ 2.0 GHz with 8 processors.

Table 1
Solver parameters used in the velocity and pressure solvers.

Parameter	Value
Newton solver max number of iterations	500
Velocity relative tolerance $\left(\frac{\|d\tilde{\mathbf{v}}\|}{\|\tilde{\mathbf{v}}_{n+1}\|}\right)$	10^{-4}
Velocity absolute tolerance (Eq. (52))	10^{-6}
Pressure relative tolerance $\left(\frac{\|d\tilde{p}\|}{\|\tilde{p}_{n+1}\|}\right)$	10^{-4}
Pressure absolute tolerance (Eq. (53))	10^{-6}
Linear solver iterative tolerance	10^{-8}
Linear solver max number of iterations	5000

Table 2
Analytical solution for pressure at steady state (Eq. (50)), pressure at the interior of the sphere and error between these two values.

Example	$2\gamma/R$ [Pa]	p [Pa]	error [%]
Sphere	8	8.03	0.44
Cube	6.46	6.45	−0.15
Step	7.14	7.18	0.56

3. Results and discussion

3.1. Static droplet

The first example studies a static droplet in order to check that at steady state: i) the pressure in the interior of the droplet is identical to the one provided by the analytical solution, and ii) the steady configuration of the domain is a sphere regardless of the initial domain shape. The analytical solution for pressure is given by the Laplace–Young equation:

$$p_L - p_g = \gamma \left(\frac{1}{R_1} + \frac{1}{R_2} \right) = \gamma \left(\frac{1}{R} + \frac{1}{R} \right) = \frac{2\gamma}{R} \tag{50}$$

where p_L is the liquid pressure, i.e., pressure inside the droplet, p_g is the gas pressure outside the droplet (zero in this case), R_1 and R_2 are the principal radii of curvature of the surface, and $R = R_1 = R_2$ is the radius of the sphere. Different initial configurations are used: a) a spherical droplet with radius $R = 0.25$ m, b) a cubic domain of 0.5 m per side, where the boundary has either zero or positive values for the curvature, and c) a step generated by removing a quarter of the cubic domain is studied. The latter domain includes the challenging case of having a corner where the curvature is negative.

The surface tension force is the only acting force and gravity is neglected ($g = 0$). Fluid density, viscosity and surface tension coefficient are set to unity ($\rho = 1$ kg m^{-3}, $\mu = 1$ kg m^{-1} s^{-1}, $\gamma = 1$ N m^{-1}). Initial pressure in the liquid is set to $p_0 = 0$ Pa. The domain is meshed using triangular elements of size $h = 1/25$ m and a time step size of 0.01 s is used. The predicted drop geometry at several times during the simulation is shown in Fig. 6.

In all configurations, the steady state solution is achieved in less than 1 s and the final geometry is a sphere. Computational time was less than 2 minutes. Table 2 shows the value of $2\gamma/R$, where R is the radius of the sphere obtained at steady state, the average value of pressure in the domain and the relative error between these two values. It can be observed that in all cases, the error observed is less than 1%.

Volume conservation of the method has also been analyzed for the cube geometry. Taking the initial volume, V_0, as a reference, the evolution of relative volume conservation error has been calculated as follows [50]:

$$E_{V_i} = \frac{|V_i - V_0|}{V_0} \tag{51}$$

where V_i is the volume measured at time t_i. The evolution of this magnitude during the simulation is displayed in Fig. 7(d), showing values of relative conservation volume error of the order of 10^{-5}. Previous models, such as that of Denner and van Wachem [50], and Zheng et al. [24], reported values of E_{V_i} between one and two orders of magnitude higher, indicating that the present model has excellent volume conservation properties.

3.1.1. Convergence analysis

In order to study the effects of the surface tension term on the convergence, the step geometry above is also solved for two additional cases:

- **Case 1:** Surface tension with semi-implicit formulation: the above example but setting term $\mathbf{H}_{ST}$ in Eq. (29) to zero
- **Case 2:** Gravity dominated problem: the above example with gravity $\mathbf{g} = 9.81$ m s^{-2} and $\gamma = 0$, fixing the velocity of the lower boundary to 0, as well as the normal component of the velocity of the lateral boundaries
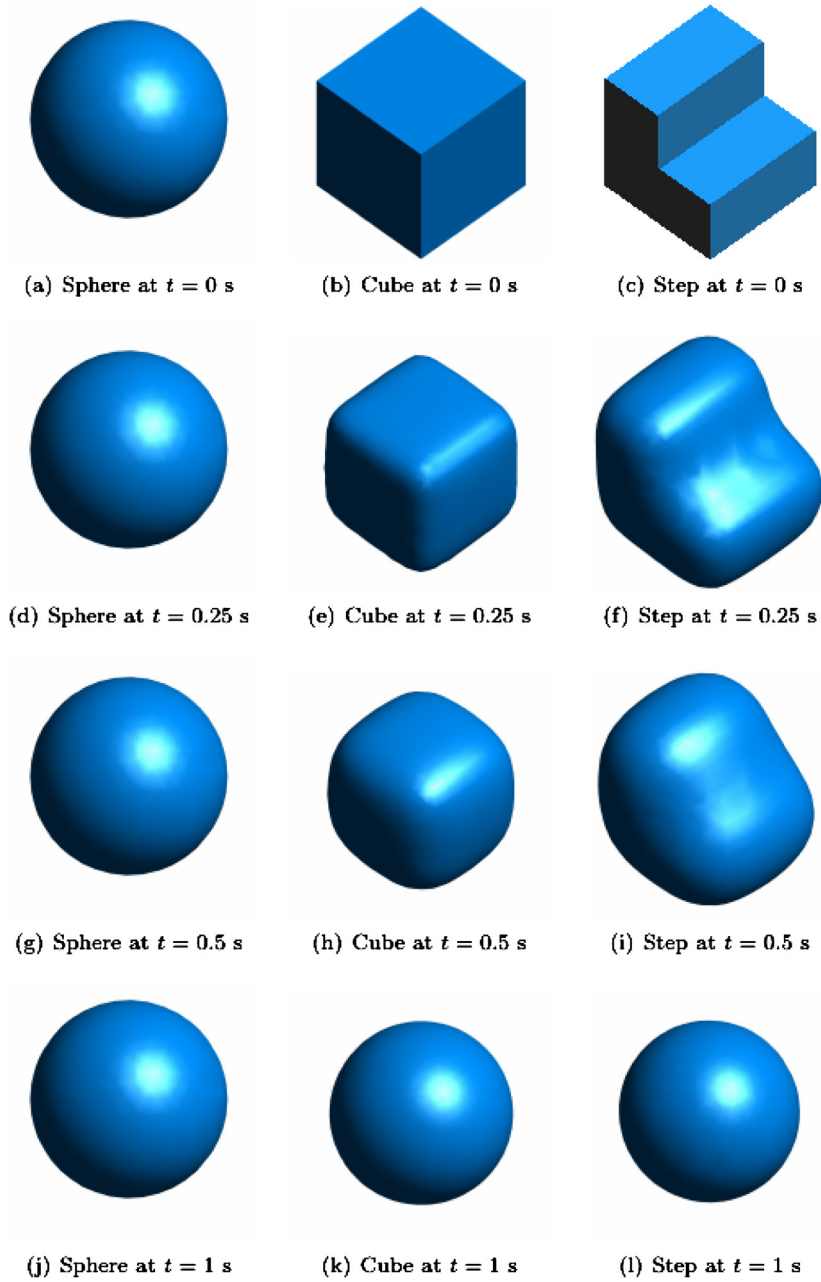
(a) Sphere at $t = 0$ s   **(b)** Cube at $t = 0$ s   **(c)** Step at $t = 0$ s

(d) Sphere at $t = 0.25$ s   **(e)** Cube at $t = 0.25$ s   **(f)** Step at $t = 0.25$ s

(g) Sphere at $t = 0.5$ s   **(h)** Cube at $t = 0.5$ s   **(i)** Step at $t = 0.5$ s

(j) Sphere at $t = 1$ s   **(k)** Cube at $t = 1$ s   **(l)** Step at $t = 1$ s

Fig. 6. Domain evolution for the sphere (left column), cube (center column) and step (right column) configurations.

The nonlinear procedure shown in Algorithm 1 is considered to converge when both the norms of velocity, $d\bar{\mathbf{v}}$, and pressure increments, $d\bar{p}$, are sufficiently small. In order to check the convergence rate of the nonlinear iteration procedure, the following norms are computed:

$$v_{\text{norm}} = \frac{\|d\bar{\mathbf{v}}\|}{N_v} \leq \varepsilon_{\text{vel}} \tag{52}$$

$$p_{\text{norm}} = \frac{\|d\bar{p}\|}{N_p} \leq \varepsilon_{\text{press}} \tag{53}$$

where $d\bar{\mathbf{v}}$ and $d\bar{p}$ are the unknowns in the quasi-Newton algorithm (Eq. (29)), N_v and N_p are the number of degrees of freedom for velocity and pressure, and ϵ_{vel} and ϵ_{press} are the tolerances for velocity and pressure, respectively. Figs. 7(a)
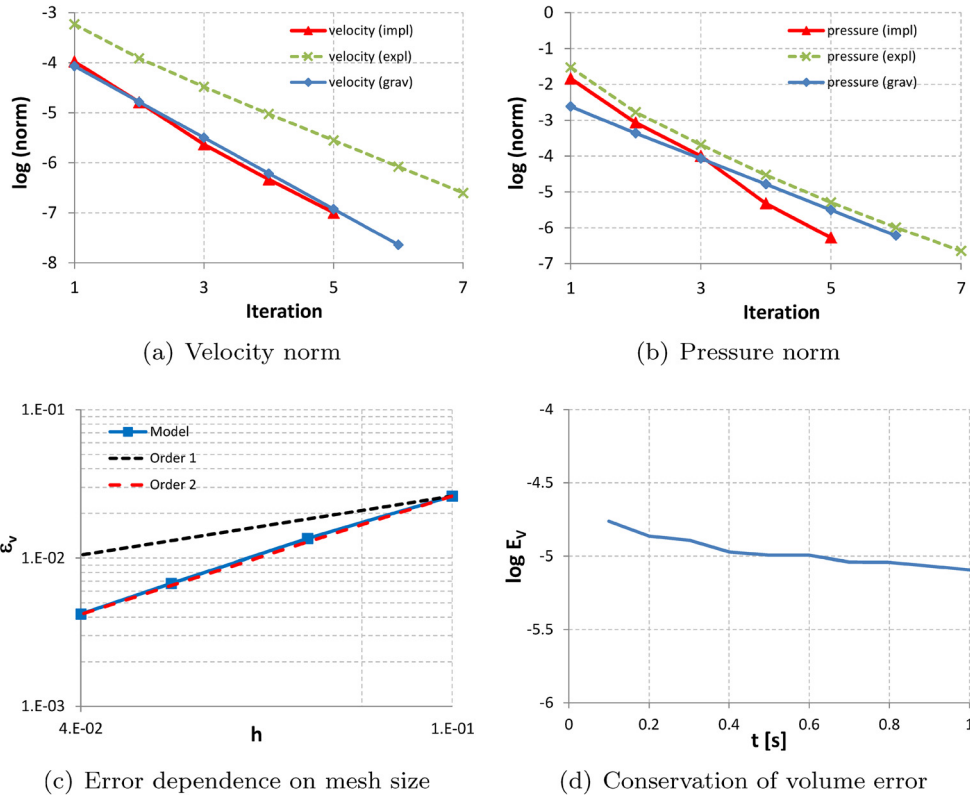
Fig. 7. (a) Velocity norm (Eq. (52)) and (b) pressure norm (Eq. (53)) evolution in the nonlinear procedure, for case 1 (expl), case 2 (grav), and the proposed implicit model (impl). (c) Velocity error dependence on mesh size for the implicit model (Eq. (54)), and (d) Evolution of the conservation of volume error (Eq. (51)) for the step example.

and 7(b) show the convergence rate for velocity and pressure for the three considered cases at $t = 0.05$ s, using a time step size $\Delta t = 0.01$ s.

Results show that both velocity and pressure converge quickly to the solution (*velocity (impl)* and *pressure (impl)*, respectively) for the implicit surface tension case, with velocity and pressure reaching the desired tolerance of 10^{-6} in 5 iterations. The rate of convergence obtained is linear for both variables.

The explicit formulation (*velocity (expl)* and *pressure (expl)* in Fig. 7) also shows a linear order of convergence, albeit with a smaller slope, for velocity and pressure. The problem needs 7 iterations to converge. This indicates the importance of including the tangent matrix corresponding to the surface tension term for obtaining a correct linearization. Results in Figs. 7(a) and 7(b) therefore show that the implicit variant of the method is more efficient than the semi-implicit one.

The order of convergence for the gravity dominated problem remains linear. Fig. 7(a) shows that the convergence of the velocity is very similar to that from the implicit case. The pressure norm, however, converges at a slower rate (Fig. 7(b)). Therefore, the order of convergence of the method is not affected by the presence of surface tension.

The dependence of the accuracy of the method on the mesh size is also measured. The fully implicit model is again considered, and the step geometry is discretized with different mesh sizes, ranging from $h = 0.1$ to 0.04 m. A 1 s simulation is performed, and the velocity of one node at the interface is compared to the analytical solution (i.e., $\mathbf{v} = 0$). The error in velocity is measured as the $L2$-norm of the difference between the numerical solution, $\mathbf{v}_h$ and the analytical one:

$$\varepsilon_v = \|\mathbf{v}_h - \mathbf{v}\|_{L2} \tag{54}$$

The error measured for each of the considered meshes is displayed in Fig. 7(c). The lines corresponding to a linear and quadratic relationship between error and mesh size are also plotted for reference. It can be observed that the error in velocity has a quadratic relationship with the mesh size. Moreover, the fact that the velocity error decreases with mesh refinement confirms the observations of Raessi et al. [44] and Denner et al. [45] that the extra viscous term at the interface is proportional to the surface tension coefficient and the time step, but inversely proportional to the mesh size. In the present method, this viscous term is the above-mentioned velocity Laplacian term from Eq. (41), also included in previous surface tension models [16,18,20,44].
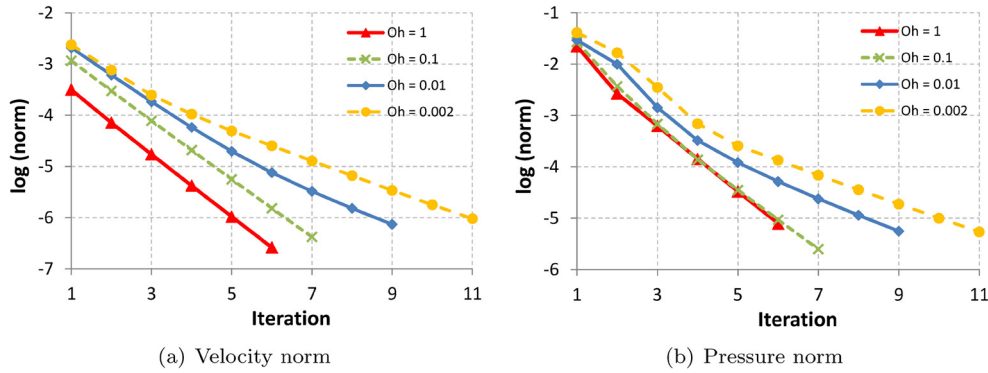
(a) Velocity norm   (b) Pressure norm

Fig. 8. (a) Velocity norm and (b) Pressure norm evolution in the nonlinear procedure for different values of the Ohnesorge number.

3.1.2. Effect of viscosity on stability

A parameter characterizing the ratio between viscous effects, and surface tension and the inertial effects is the Ohnesorge number, defined as:

$$Oh = \frac{\mu}{\sqrt{2\rho\gamma R}} \tag{55}$$

where μ is the dynamic viscosity, ρ is the density, γ is the surface tension coefficient and R is a characteristic length, taken as the radius of the droplet in this example. Settings chosen for the example in the previous sub-section correspond to $Oh \approx 1.414$. Therefore, viscous effects dominated over surface tension and inertia. As the viscous effects are reduced, numerical stability is more challenging. It is known that $Oh < 0.01$ define a challenging setting for models accounting for surface tension typically leading to the spurious numerical instabilities at the interface.

In this sub-section, the step geometry case in the previous section is solved with different viscosity values in order to check the stability of the method. Fig. 8 shows the evolution of velocity and pressure norms in the nonlinear procedure for different of Oh values.

The lowest Oh that can be simulated while maintaining the stability of the solution is $Oh = 0.002$, which means that the method is stable even when surface tension and inertial effects are dominant over viscous ones.

3.1.3. Time step analysis

A critical time step for solving surface tension dominated problems using a fully explicit scheme was estimated by Sussman and Ohta [16]:

$$\Delta t_{\mathrm{crit}} = \sqrt{\frac{(\rho_L + \rho_G)\, h^3}{\gamma\,(2\pi)^3}} \tag{56}$$

where ρ_L is the density of the considered fluid, ρ_G is the density of the surrounding fluid, γ is the surface tension coefficient between both fluids and h is the mesh size. Using Eq. (56) with the properties described in Section 3.1, the critical time step for the above-mentioned problem is 5×10^{-4} s. Although other models consider viscosity effects on the time step restriction [17,24,51], the current model focuses on the stability regarding the capillary time step constraint.

The droplet example with the initial step geometry above is again reproduced in order to: i) check the effect of time step size on convergence of the nonlinear procedure, and ii) estimate the critical time step. The problem is solved for several time steps, i.e., $\Delta t = 0.1$, 0.01 and 0.001 s, and the maximum time step is identified as the largest step in the set above that leads to convergence of the quasi-Newton problem at $t = 0.5$ s using the set tolerance for both velocity and pressure norms in Table 1.

Effects of increasing Δt on the performance of the nonlinear procedure are shown in Figs. 9(a) and 9(b). For the three different time steps chosen, the rate of convergence is similar. Results indicate that increasing the time step two orders of magnitude (i.e., from 0.001 to 0.1 s) leads to one (for velocity norm) and two (pressure norm) extra iterations for the convergence of the nonlinear procedure. Therefore, such increase of the time step does not impoverish the overall performance of the method.

For the maximum time step size (i.e., 0.1 s), the nonlinear procedure converges to the desired tolerance after 8 iterations. For $\Delta t = 0.001$ s, the solution is reached after 6 iterations, one more than the $\Delta t = 0.01$ case. Although this may seem counter-intuitive, the term $\mathbf{H}_{ST}$ in the tangent matrix, corresponding to linearization of the surface tension term (Eq. (44)), is proportional to the time step size, and therefore reducing the time step size leads to a decrease in this term. Thus, an optimal time step size exists such that the initial solution to the linear solver is appropriate and the step size is not too small.

(a) Velocity norm   (b) Pressure norm

(c) Error dependence on time step size
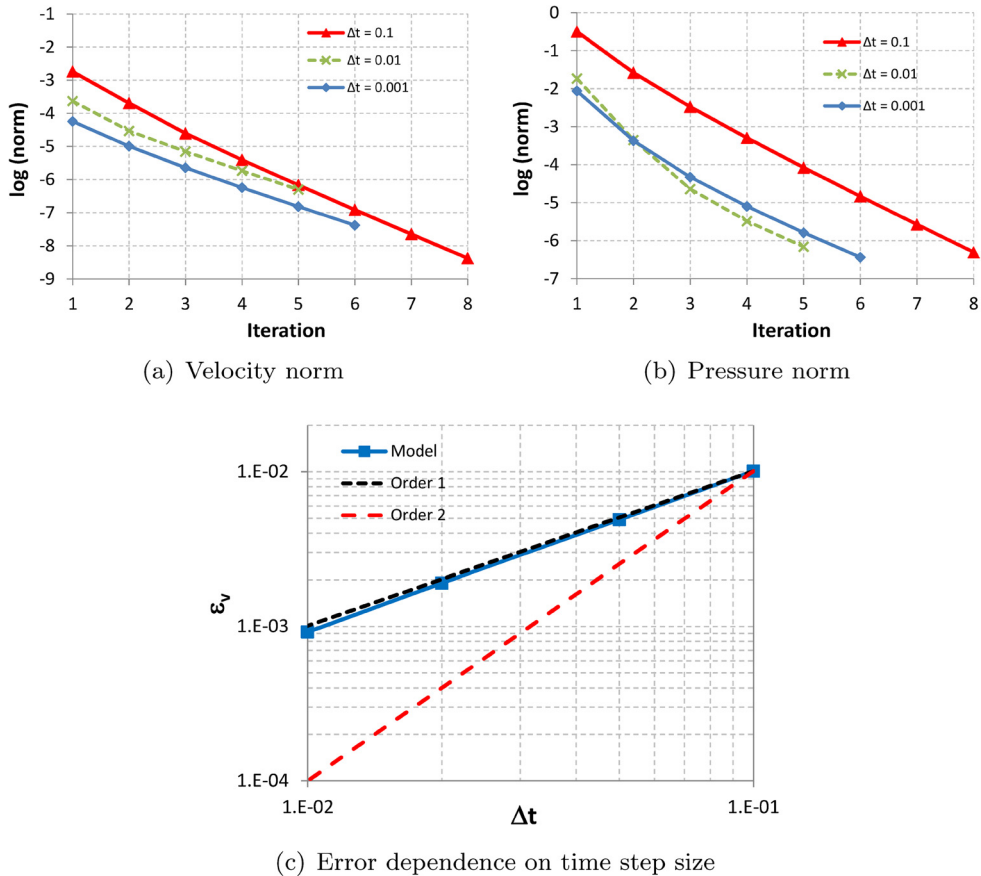
Fig. 9. (a) Velocity norm and (b) pressure norm evolution in the nonlinear procedure for several time steps, and (c) velocity error dependence on time step size.

The dependence of the accuracy of the method on the time step size has also been measured. Different time step sizes, ranging from $\Delta t = 0.1$ to 0.01 s have been considered. A 1 s simulation has been performed, and the velocity of one node at the interface is compared to the analytical solution (i.e., $\mathbf{v} = 0$). The error in velocity is measured as the $L2$-norm of the difference between the numerical solution, $\mathbf{v}_h$ and the analytical one (Eq. (54)). The error measured for each of the considered meshes is displayed in Fig. 9(c). The lines corresponding to a linear and quadratic relationship between error and mesh size have also been plotted for reference. It can be observed that the error in velocity has a linear relationship with the time step size.

Maximum time step has also been studied for step size $= \{0.01, 0.02, ..., 0.1\}$ for the step case considering both implicit and semi-implicit formulations[5] (i.e., with and without the linearized surface tension term $\mathbf{H}_{ST}$). Table 3 shows a comparison between the critical time step for the problem at hand for the implicit and semi-implicit formulations. In both examples, the maximum time step for the implicit formulation was 10 times higher than that for the semi-implicit one. The critical time step for the latter case (i.e., $\Delta t_{max}^{semi} = 0.01$ s) is 20 times higher than the critical time step resulting from Eq. (56) because the formulation is semi-implicit.

Raessi et al. [44] reported that the time step restriction in their model could be exceeded by a factor of 5 while maintaining the stability of the solution. In the current formulation, this factor can be as high as 200. However, this value is lower than the factor of almost 1000 reported by Zheng et al. [24].

3.1.4. Comparison with VOF: pressure accuracy and parasitic currents

A simplified version of the static drop example in two dimensions is considered here. This example has been performed by many authors in literature due to its simplicity. Gerlach et al. [52] performed a comparison between three different VOF methods. For each of them, the average pressure, $\langle P \rangle$, defined as the sum of nodal pressures divided by the total number of nodes, as well as L_1 and L_2 error norms were computed. The error norms were defined as:

---

[5] The sphere example does not have a limiting time step since the initial configuration of the domain coincides with the steady state one. Thus, regardless the time step used for the simulation, the solution always converges to the desired tolerance in a few iterations.

Table 3
Maximum time step (Δt_max) in the simulations using the implicit *(impl)* and semi-implicit *(semi)* formulations, and ratio between them (Δt_max *(impl)*/Δt_crit *(semi)*).

Example	Δt_max (semi) [s]	Δt_max (impl) [s]	Factor
Cube	0.01	0.1	10
Step (impl)	0.01	0.1	10

Table 4
Comparison of the best VOF model in [52] and the present model.

Method	$\langle P \rangle / P_\text{exact}$	L_2	L_1	$|u_{\text{max},1}|$	$|u_{\text{max},50}|$
PROST	1.0007	3.02×10^{-3}	7.18×10^{-4}	1.14×10^{-7}	5.69×10^{-6}
This work	1.0003	3.57×10^{-4}	3.12×10^{-4}	4.38×10^{-8}	1.22×10^{-6}

Table 5
Comparison of the best VOF model in [52] and the present model.

h [m]	$\langle P \rangle / P_\text{exact}$	L_2	L_1	$|u_{\text{max},1}|$	$|u_{\text{max},50}|$
PROST					
0.2	1.0048	4.83×10^{-3}	4.81×10^{-3}	7.82×10^{-8}	3.91×10^{-6}
0.1	1.0009	9.79×10^{-4}	9.48×10^{-4}	1.70×10^{-7}	8.53×10^{-6}
0.05	1.00007	5.25×10^{-4}	7.04×10^{-5}	4.34×10^{-7}	2.17×10^{-5}
This work					
0.2	1.0012	1.25×10^{-3}	1.25×10^{-4}	6.08×10^{-9}	4.09×10^{-7}
0.1	1.0003	3.57×10^{-4}	3.12×10^{-4}	4.38×10^{-8}	1.22×10^{-6}
0.05	1.00007	3.65×10^{-4}	7.85×10^{-5}	4.26×10^{-7}	3.55×10^{-6}

$$L_1 = \left| \frac{\sum_i^N P_i - P_\text{exact}}{N P_\text{exact}} \right| \qquad L_2 = \left[\frac{(\sum_i^N P_i - P_\text{exact})^2}{N P_\text{exact}^2} \right]^{0.5} \tag{57}$$

where P_i was the nodal pressure at node i, P_exact was the analytical solution for pressure (Eq. (11)), and N was the number of nodes.

A circular droplet of radius $R = 0.02$ m is considered, with the following properties: $\rho = 1000$ kg m^{-3}, $\mu = 10^{-5}$ kg m^{-1} s^{-1}, $\gamma = 0.02361$ N m^{-1}. The droplet is surrounded by an external fluid, represented by a square domain of $L = 0.06$ m per side, and $\rho_G = 500$ kg m^{-3}. Both domains are discretized with triangular elements of size $h = 0.001$ m. The time step size is 10^{-5} s, which is the same value used in [52]. Average pressure and pressure errors between the best VOF model reported by Gerlach et al. [52] and the model presented in this work have been compared, and results are shown in Table 4. The ratio between mean pressure and the exact value is closer to 1.0 for the present model, and the L_1 and L_2 errors obtained are smaller as well.

The static drop example neglects all external forces, and therefore surface tension should balance a constant pressure within the droplet at the interface. It has been observed that spurious velocities, i.e., "parasitic currents", appear as a numerical artifact [23,45,52,53]. Gerlach et al. [52] obtained the maximum velocity norm in the domain at the first time step, $|u_{\text{max},1}|$, and after 50 time steps, $|u_{\text{max},50}|$. These values are compared as well in Table 4, showing that parasitic currents are smaller for the present model. Fig. 10(a) shows the velocity field within the droplet, where parasitic currents appear close to the interface. This observation has been reported by other state-of-the-art models, such as the ones presented by Popinet [53], and Schroeder et al. [23].

Gerlach et al. [52] also performed a mesh-sensitivity analysis for the present example, the only difference being the density of the surrounding fluid, i.e., $\rho_G = 1$ kg m^{-3}. Mean pressure, pressure errors and maximum velocities were measured for three different mesh sizes: $h = 0.2$, 0.1, and 0.05 m. Comparison between the two models is shown in Table 5. Errors measured for the current model are lower than those obtained by Gerlach et al. [52].

3.2. Dynamic drop

The present model can be used to simulate the oscillations of a free droplet. This phenomenon has been studied in literature by Lamb [54] and Prosperetti [55], among others. Prosperetti [55] proposed a model for the temporal evolution of the droplet interface. If a perturbation was introduced at the interface with an initial amplitude A_{n0}, it was predicted that the amplitude of this perturbation decayed exponentially:

$$A_n = A_{n0} \exp\left(-\frac{t}{\tau_n} \right) \tag{58}$$

where τ_n was a time constant for the decay that depended on the fluid properties [55]:

(a) Static droplet

(b) $\theta_S = 70$ deg

(c) $\theta_S = 90$ deg

(d) $\theta_S = 135$ deg
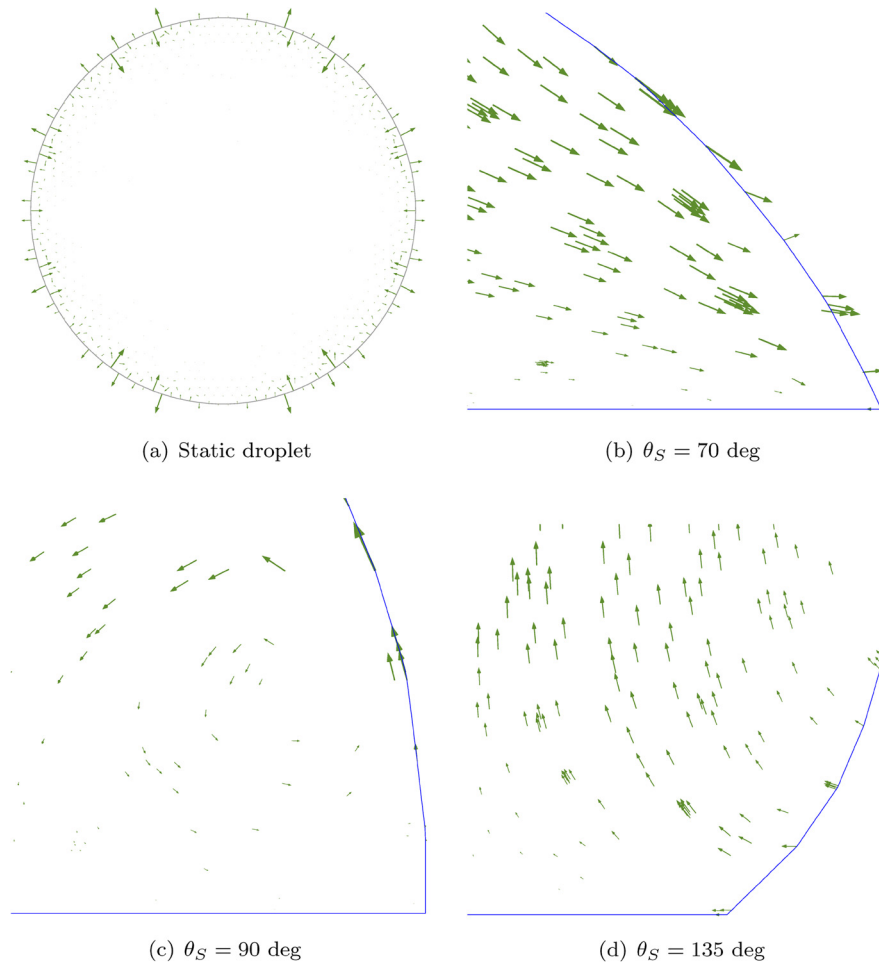
Fig. 10. Velocity fields for (a) the static droplet example at $t = 50 \times 10^{-5}$ s, and (b)–(d) for the sessile droplet examples.

$$\tau_n = \frac{\rho R^2}{\mu (n-1)(2n+1)} \tag{59}$$

where R was the radius of the initially unperturbed free droplet, and n is the oscillation mode.

A spherical droplet with radius $R = 1$ m, $p_0 = 0$ Pa, $\gamma = 1$ N m^{-1}, $\rho = 1$ kg m^{-3}, and $\mu = 0.01$ kg m^{-1} s^{-1} exposed to an initial perturbation on the interface of amplitude $A_{n0} = 0.01$ m is simulated to compare numerical and analytical results. The domain is meshed using triangular elements of size $h = 1/25$ m and a time step size of 0.01 s is used. The temporal evolution of the amplitude corresponding to mode 2 of oscillation is displayed as a black solid line in Fig. 11(a). The solution is compared to the model proposed by Prosperetti in reference [55], shown in Fig. 11(a) as a black dashed line. Figs. 11(b)–11(d) show the evolution of the domain shape through the simulation.

Two additional values for the viscosity have been considered: $\mu = 0.1$ and $\mu = 0.001$ kg m^{-1} s^{-1}. Results for the simulations and analytical solutions are shown in green and blue lines in Fig. 11(a), respectively. The predicted solution in all three cases has good agreement with the analytical one.

3.3. Capillary waves

Many surface tension models in literature have included the dispersion of capillary waves benchmark study [45,52,56,57], therefore this study is also analyzed here. Two fluids, initially at rest, are considered with the following properties[6]: $\rho_1 = 0.1$ kg m^{-3}, $\rho_2 = 1$ kg m^{-3}, $\mu_1 = 1.6394 \times 10^{-4}$ kg m^{-1} s^{-1}, $\mu_2 = 1.6394 \times 10^{-3}$ kg m^{-1} s^{-1}, and $\gamma = 0.25 \, \pi^{-3}$ N m^{-1}.

---

[6] Although the case with density ratios $\rho_2/\rho_1 = 10$ is also analyzed in Prosperetti's study [58], most authors in literature choose the ratio $\rho_2/\rho_1 = 1$. The reason why a ratio of 10 is chosen in this work is because for density ratios close to 1, the numerical solution becomes unstable. This has been reported in literature as "Added-mass effect" [59,60]. This is the first example using the embedded formulation [25] where the effect has been manifested. Why it appears in this case and not in other examples, and how to alleviate it will be the subject of future studies.
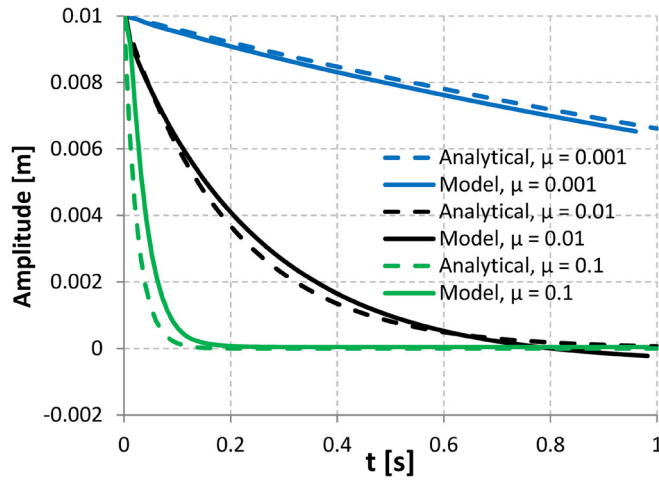
(a) Temporal evolution of A_n
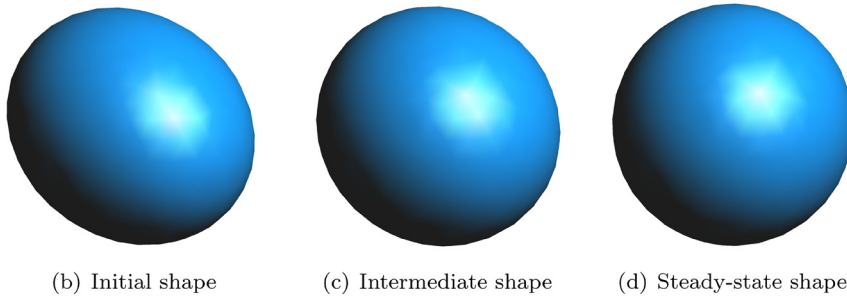
(b) Initial shape   (c) Intermediate shape   (d) Steady-state shape

Fig. 11. (a) Temporal evolution of amplitude of oscillation of mode 2 according to the present model (solid lines) and the analytical solution (dashed line) presented in [55], for different viscosity values, and (b)–(d) shape evolution of the droplet during the simulation. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)
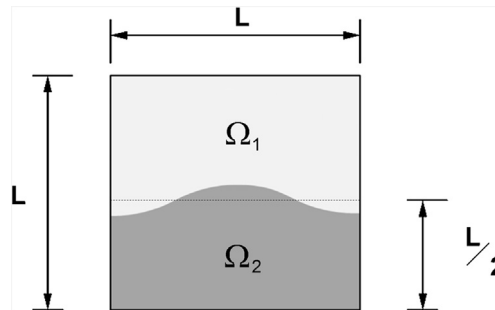

Fig. 12. Computational domain for the capillary waves example. Both fluids are initially at rest, with their interface shaped following a sinusoidal function with amplitude 0.01 m and wavelength 1 m.

Gravity is neglected and surface tension is the only acting force. The domain is represented by a square of 1 m per side, where each fluid occupies half of the domain, as displayed in Fig. 12.

A capillary wave is introduced at the interface by giving it the shape of a sinusoidal curve with initial amplitude $a_0 = 0.01$ m, and wavelength $\lambda = 1$ m. The domain is discretized using three different meshes with elements of 1/16, 1/32, and 1/64 m in size, and the time step is $\Delta t = 0.001$ s. A slip boundary condition is applied to all boundaries of the domain, and the motion of the interface between the fluids is given by surface tension. The analytical solution for the temporal evolution of the amplitude of the capillary wave was given by Prosperetti [58], and it is displayed in Fig. 13 (black solid line).

The effects of mesh size on the numerical solution are assessed first. Results for the three different mesh sizes (i.e., $h = 1/16$, 1/32, and 1/64 m) considered for the present case of study are shown in Fig. 13(a). The mesh quality clearly affects the frequency and amplitude of the solution. Whereas the meshes with 16 and 32 elements show higher frequency
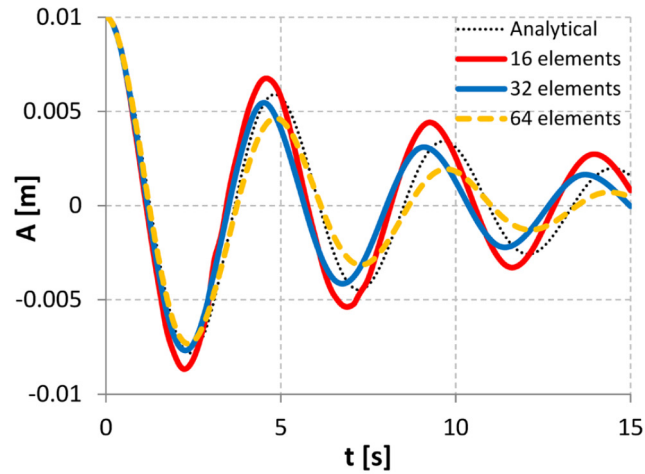
(a) Solution obtained with different mesh sizes
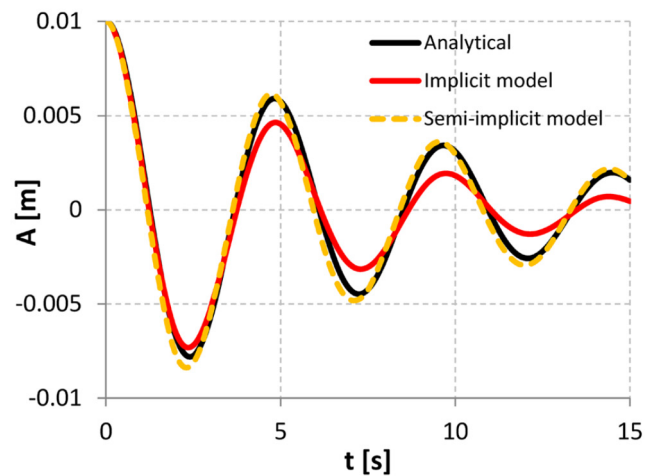
(b) Comparison between analytical solution, and implicit and semi-
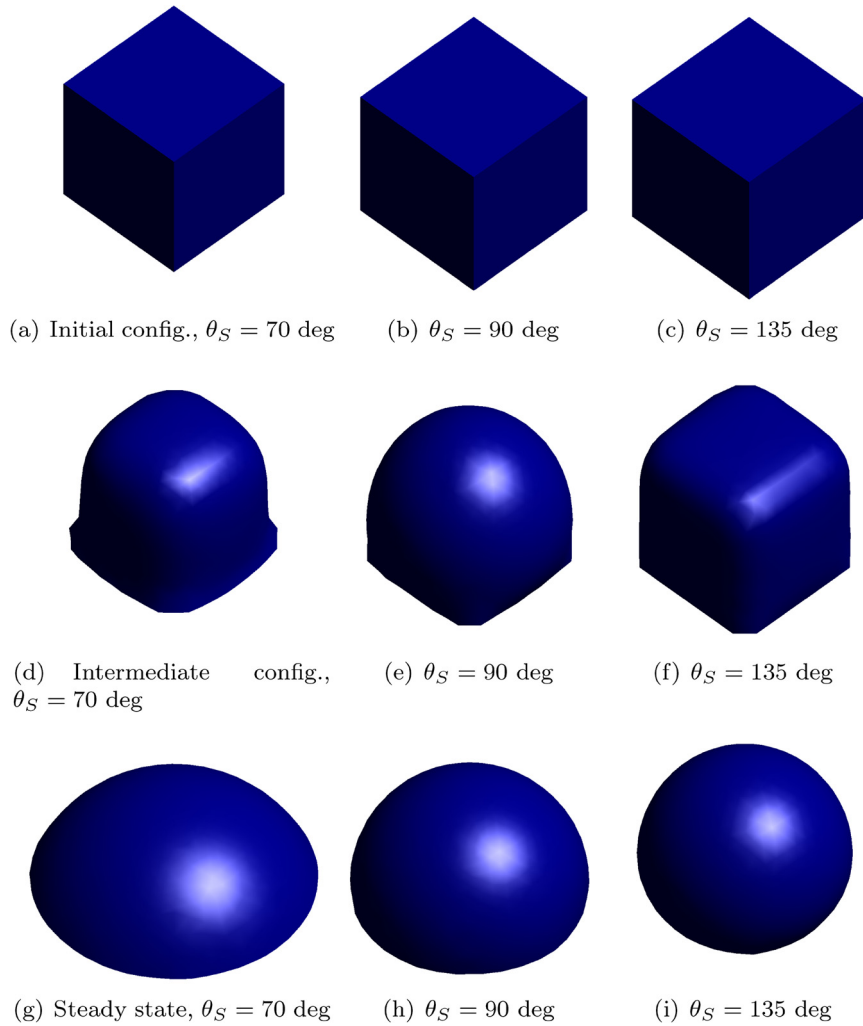implicit models

Fig. 13. Temporal evolution of the amplitude of the capillary wave (a) for different mesh sizes, and (b) the numerical solutions the implicit and semi-implicit variants of the present model compared to the analytical solution from Prosperetti [58].

than the analytical solution (black dotted line), the solution with the finest mesh (i.e., the one with $h = 1/64$ m) shows a frequency almost identical to the analytical one. However, this solution also has more damping than the other two. This effect is probably caused by the velocity Laplacian term. A mesh sensitivity study was performed and shown that a mesh with 64 elements per side provides a grid independent solution. Therefore, meshes with less elements result in an overprediction of the vibration frequency, while further refinement yields the same solution.

In order to check the effects of the velocity Laplacian on the solution, the same simulation has been performed using the semi-implicit variant of the method. In this case, only the finest mesh (i.e., $h = 1/64$ m) is considered. The corresponding result is depicted in Fig. 13(b) (orange dashed line). The solution obtained has almost identical frequency and amplitude than the analytical one, indicating that the semi-implicit variant of the current method is capable of reproducing the solution proposed by Prosperetti [58].

3.4. Sessile drop in different substrates

Sessile drop simulations are performed to validate the model capability of representing wetting on solid substrates. The initial configuration for the domain is a cube of 1 mm in each side. The domain is discretized using tetrahedral elements of $h = 0.1$ mm and a time step $\Delta t = 10^{-2}$ s. Three different substrates are considered, characterized by a static contact angle

(a) Initial config., $\theta_S = 70$ deg   (b) $\theta_S = 90$ deg   (c) $\theta_S = 135$ deg

(d) Intermediate config., $\theta_S = 70$ deg   (e) $\theta_S = 90$ deg   (f) $\theta_S = 135$ deg

(g) Steady state, $\theta_S = 70$ deg   (h) $\theta_S = 90$ deg   (i) $\theta_S = 135$ deg

Fig. 14. Evolution of droplet along time with (left column) $\theta_S = 70$ deg, (center column) $\theta_S = 90$ deg and (right column) $\theta_S = 135$ deg.

of $\theta_S = 70$, 90 and 135 deg, respectively. The only acting forces on the droplet are gravity and surface tension. However, for the considered droplet size, surface tension effects dominate over gravitational forces since the Bond number is less than 1.[7]

The droplet domain evolution on the three considered substrates is displayed in Fig. 14. At $t = 0$ s, the domain is not deformed. Immediately after starting the simulation, the interface Γ_I is deformed by the surface tension force and the curvature is minimized (i.e., the corners vanish). Then, the contact line starts to move until equilibrium is reached within 1 s. The computational time required for each example was approximately 1 h.

In order to measure the error in the simulations, the difference between the modeled contact angle and the prescribed value for each substrate is computed. The obtained contact angle and the difference between the theoretical static contact angle and the angle observed in the simulation once equilibrium is reached are shown in Table 6. The value of contact angle at steady state is in good agreement with the prescribed contact angle, giving a maximum relative error of 0.68%. Simulations were repeated for smaller mesh sizes (i.e. $h = 0.05$ and 0.01 mm), and results varied less than 3%.

For a similar simulation, Buscaglia and Ausas [42] used a time step of 2×10^{-7} s, which is five orders of magnitude smaller than the one used in the present study. Parasitic currents have also been observed in this example. Their distribution is shown in Figs. 10(b)–10(d) for the three different substrates. A similar distribution was obtained by Buscaglia and Ausas [42], although they did not report their magnitude. In this work, the maximum velocity obtained at steady state
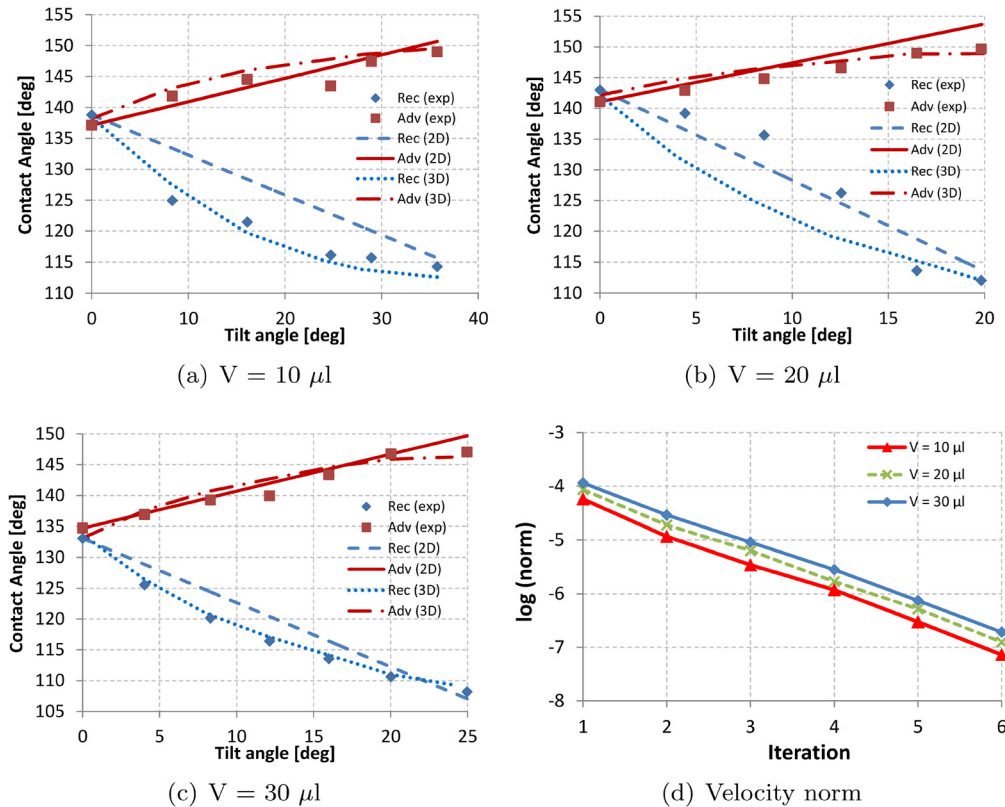

7

$$Bo = \frac{\rho g d^2}{\gamma} = \frac{1000 \cdot 9.81 \cdot \left(10^{-3}\right)^2}{0.072} \approx 0.14 < 1. \tag{60}$$

Table 6
Prescribed (θ_S) and obtained (θ_{obs}) contact angles, and relative error (ϵ_θ) between these variables.

	$\theta_S = 70$ deg	$\theta_S = 90$ deg	$\theta_S = 135$ deg
θ_{obs} [deg]	69.58	90.61	135.73
ϵ_θ [%]	−0.6	0.68	0.54

(a) $V = 10 \ \mu$l

(b) $V = 20 \ \mu$l

(c) $V = 30 \ \mu$l

(d) Velocity norm

Fig. 15. ((a)–(c)) Comparison between the measured contact angle from reference [5] (square and diamond markers) and the results obtained with the 2D and 3D model, and (d) velocity norm evolution during the iterative procedure at $t = 10$ s.

ranged between 10^{-6} and 10^{-5}, being higher for the case with $\theta_S = 70$ deg. This value is in agreement with the results shown in Table 5.

3.5. Sessile drop on an inclined plane

On a recent publication [5], a two-dimensional version of the proposed model was used to predict the deformation of a sessile droplet placed on a horizontal substrate (SIGRACET 24BC, GDL side) that is then tilted at a constant rate of 0.4 deg s^{-1}. Results were compared to the experimental observations using 3 different droplet volumes. In this example, the same problem is solved in three dimensions.

Fluid properties for air and water are the following: $\rho_a = 1.2$ kg m^{-3}, $\rho_w = 1000$ kg m^{-3}, $\mu_a = 1.98 \times 10^{-5}$ kg m^{-1} s^{-1}, $\mu_w = 10^{-3}$ kg m^{-1} s^{-1}, and $\gamma = 0.072$ N m^{-1}. These values are taken at a reference temperature $T = 298$ K and pressure $p = 1$ atm. Gravity is also included in this example, with $g = 9.81$ m s^{-2}. Mesh size in the three different cases depends on the droplet diameter, so the ratio $D/h = 20$ elements, where D is the droplet diameter and h is the mesh size, is used as the criteria to generate the mesh. Time step in all cases is $\Delta t = 0.01$ s.

Fig. 15 shows the comparison between the results obtained with the current three-dimensional model and experimental and numerical results reported in reference [5]. The 3D model is able to capture the experimentally observed nonlinear behavior whereas the 2D model showed a linear relationship between contact angle and plate tilt angle.

There is good agreement between the numerical and experimental data for the 10 and 30 µl-volume droplets (Fig. 15(a) and 15(c)). For the 20 µl-volume droplet (Fig. 15(b)), the receding contact angle does not show the same agreement. It can be observed that in this case, the receding contact angle has a different behavior compared to the other two cases. This could be due to non-uniform surface roughness or chemical heterogeneity of the substrate.

This example has the advantage that the droplet is initially at rest, and the configuration coincides with the steady-state one. Moreover, air is in quiescent conditions. Overall, the droplet does not undergo large deformations, which means that the model reaches convergence in a few iterations. This can be observed in Fig. 15(d), where the evolution of the velocity norm in the nonlinear procedure is shown for the three droplet volumes. The norm of the velocity is lower than 10^{-6} in just 4 iterations. Pressure norms had similar trends in the three cases, but with values one order of magnitude higher. Therefore, choosing an absolute tolerance of 10^{-6} for velocity and 10^{-5} for pressure is enough for this particular problem.

4. Summary and conclusions

In this paper, a finite element model for incompressible flows with surface tension effects has been presented and coupled to a Lagrangian fluid flow solver.

The use of a Lagrangian description for the liquid domain allows for the representation of the boundary via a boundary mesh. Not only it allows to track the evolving boundary without any additional technique, but it also facilitates the computation of the curvature necessary for the surface tension evaluation. Differential geometry has been used to derive the tangent matrix corresponding to the surface tension. This term alleviates the time restrictions characteristic of explicit surface tension models. The proposed model is relatively easy to implement within an existing moving-mesh framework since only an extra Laplacian-like term needs to be included in the tangent matrix in order to treat the surface tension implicitly.

Steady state droplet geometry studies show that present model predictions coincide with analytical ones for the examples considered. The maximum time step obtained for the proposed method is 20 times higher than that reported in reference [16]. Increasing step size is critical in order to perform simulations of processes that might take several minutes such as in fuel cell droplet shedding [28]. The method has second order accuracy in space and first order accuracy in time. The dynamic droplet example shows that predicted temporal decay of the perturbation matches the theoretical prediction shown in reference [55]. Volume conservation of the method has also been assessed, showing errors of the order of 10^{-5}, which is between one and two orders of magnitude lower than state-of-the-art models [24,50].

Pressure accuracy and parasitic current magnitude have also been compared to VOF-based models [52], obtaining better results with the present model. Parasitic currents have also been observed in the sessile droplet examples, where it has been concluded that their magnitude increases for lower contact angle values.

The validity of the model to study sessile droplets on different substrates and on a tilting plate has been demonstrated by comparison to experimental data. Results for droplets in a tilted plate show that the contact angle evolution is nonlinear with respect to the tilt angle. This observation is consistent with the experimental data obtained and can only be predicted by three-dimensional simulations.

Acknowledgements

References

[1] C.E. Brennen, Cavitation and Bubble Dynamics, Cambridge University Press, 2013.
[2] M. Kreder, J. Alvarenga, P. Kim, J. Aizenberg, Design of anti-icing surfaces: smooth, textured or slippery?, Nat. Rev. Mater. 1 (2016) 15003.
[3] Q. Li, G. Flamant, X. Yuan, P. Neveu, L. Luo, Compact heat exchangers: a review and future applications for a new generation of high temperature solar receivers, Renew. Sustain. Energy Rev. 15 (9) (2011) 4855–4875.
[4] S. Shukla, K. Domican, K. Karan, S. Bhattacharjee, M. Secanell, Analysis of low platinum loading thin polymer electrolyte fuel cell electrodes prepared by inkjet printing, Electrochim. Acta 156 (2015) 289–300.
[5] A. Jarauta, P.B. Ryzhakov, M. Secanell, P.R. Waghmare, J. Pons-Prats, Numerical study of droplet dynamics in a polymer electrolyte fuel cell gas channel using an embedded Eulerian–Lagrangian approach, J. Power Sources 323 (2016) 201–212.
[6] S. Osher, J. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[7] C. Hirt, B. Nichols, Volume of Fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.
[8] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion, in: Numerical Methods for Fluid Dynamics, vol. 24, 1982, pp. 273–285.
[9] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, S. Zaleski, Volume-of-fluid interface tracking with smoothed surface stress methods for three dimensional flows, J. Comput. Phys. 152 (1999) 423–456.
[10] A. Jarauta, P. Ryzhakov, Challenges in computational modeling of two-phase transport in polymer electrolyte fuel cells flow channels: a review, Arch. Comput. Methods Eng. (2017) 1–31.
[11] N. Sukumar, N. Moës, B. Moran, T. Belytschko, Extended finite element method for three-dimensional crack modelling, Int. J. Numer. Methods Eng. 48 (11) (2000) 1549–1570.
[12] ANSYS Fluent, http://www.ansys.com/Products/Fluids/ANSYS-Fluent.
[13] Gerris flow solver, http://gfs.sourceforge.net/wiki/index.php/Main_Page.
[14] STAR-CCM+, https://mdx.plm.automation.siemens.com/star-ccm-plus.
[15] J. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys. 100 (1992) 335–354.
[16] M. Sussman, M. Ohta, A stable and efficient method for treating surface tension in incompressible two-phase flow, SIAM J. Sci. Comput. 31 (2009) 2447–2471.

[17] F. Denner, B. van Wachem, Numerical time-step restrictions as a result of capillary waves, J. Comput. Phys. 285 (2015) 24–40.
[18] S. Hysing, A new implicit surface tension implementation for interfacial flows, Int. J. Numer. Methods Fluids 6 (51) (2006) 659–672.
[19] K. Kamran, R. Rossi, E. Onate, S. Idelsohn, A compressible Lagrangian framework for the simulation of the underwater implosion of large air bubbles, Comput. Methods Appl. Mech. Eng. 255 (2013) 210–225.
[20] P.J. Slikkerveer, E.P.V. Lohuizen, S.B.G. O'Brien, An implicit surface tension algorithm for Picard solvers of surface-tension-dominated free and moving boundary problems, Int. J. Numer. Methods Fluids 22 (1996) 851–865.
[21] P.H. Saksono, D. Perić, On finite element modelling of surface tension. Variational formulations and applications – Part I: quasistatic problems, Comput. Mech. 38 (2006) 265–281.
[22] P.H. Saksono, D. Perić, On finite element modelling of surface tension. Variational formulations and applications – Part II: dynamic problems, Comput. Mech. 38 (2006) 251–263.
[23] C. Schroeder, W. Zheng, R. Fedkiw, Semi-implicit surface tension formulation with a Lagrangian surface mesh on an Eulerian simulation grid, J. Comput. Phys. 231 (4) (2012) 2092–2115.
[24] W. Zheng, B. Zhu, B. Kim, R. Fedkiw, A new incompressibility discretization for a hybrid particle MAC grid representation with surface tension, J. Comput. Phys. 280 (2015) 96–142.
[25] P.B. Ryzhakov, A. Jarauta, An embedded approach for immiscible multi-fluid problems, Int. J. Numer. Methods Fluids 81 (2015) 357–376.
[26] J. Marti, P. Ryzhakov, S. Idelsohn, E. Oñate, Combined Eulerian–PFEM approach for analysis of polymers in fire situations, Int. J. Numer. Methods Eng. 92 (2012) 782–801.
[27] P. Ryzhakov, R. Rossi, S. Idelsohn, E. Oñate, A monolithic Lagrangian approach for fluid–structure interaction problems, Comput. Mech. 46 (2010) 883–899.
[28] P. Ryzhakov, A. Jarauta, M. Secanell, J. Pons-Prats, On the application of the PFEM to droplet dynamics modeling in fuel cells, Comput. Part. Mech. 4 (3) (2017) 285–295.
[29] R. Bird, W. Stewart, E. Lightfoot, Transport Phenomena, 2nd edition, John Wiley and Sons, 2002.
[30] F. White, Viscous Fluid Flow, 2nd edition, McGraw-Hill, 1991.
[31] J. Donea, A. Huerta, Finite Element Methods for Flow Problems, 1st edition, John Wiley & Sons, 2003.
[32] A. Limache, S. Idelsohn, R. Rossi, E. Oñate, The violation of objectivity in Laplace formulations of the Navier–Stokes equations, Int. J. Numer. Methods Fluids 54 (6–8) (2007) 639–664.
[33] P. Ryzhakov, J. Cotela, R. Rossi, E. Oñate, A two-step monolithic method for the efficient simulation of incompressible flows, Int. J. Numer. Methods Fluids 74 (12) (2014) 919–934.
[34] P. Ryzhakov, A modified fractional step method for fluid–structure interaction problems, Rev. Int. Métod. Numér. Cálc. Diseño Ing. 33 (1) (2017) 58–64.
[35] T.J.R. Hughes, L.P. Franca, G.M. Hulbert, A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/least-squares method for advective–diffusive equations, Comput. Methods Appl. Mech. Eng. 73 (2) (1989) 173–189.
[36] R. Codina, A stabilized finite element method for generalized stationary incompressible flows, Comput. Methods Appl. Mech. Eng. 190 (20–21) (2001) 2681–2706.
[37] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, Comput. Methods Appl. Mech. Eng. 190 (2000) 1579–1599.
[38] E. Oñate, A stabilized finite element method for incompressible viscous flows using a finite increment calculus formulation, Comput. Methods Appl. Mech. Eng. 182 (3) (2000) 355–370.
[39] M. Meyer, M. Desbrun, P. Schröder, A.H. Barr, Discrete differential-geometry operators for triangulated 2-manifolds, in: Visualization and Math. III, 2003, pp. 35–57.
[40] I.V.M. Tasso, S.S. Rodrigues, G.C. Buscaglia, Assessment of curvature approximation methods in the simulation of viscous biological membranes.
[41] E. Bänsch, Finite element discretization of the Navier–Stokes equations with a free capillary surface, Numer. Math. 88 (2) (2001) 203–235.
[42] G. Buscaglia, R. Ausas, Variational formulations for surface tension, capillarity and wetting, Comput. Methods Appl. Mech. Eng. 200 (45–46) (2011) 3011–3025.
[43] C. Weatherburn, Differential Geometry of Three Dimensions, 4th edition, Cambridge University Press, 1955.
[44] M. Raessi, M. Bussmann, J. Mostaghimi, A semi-implicit finite volume implementation of the CSF method for treating surface tension in interfacial flows, Int. J. Numer. Methods Fluids 59 (10) (2009) 1093–1110.
[45] F. Denner, F. Evrard, R. Serfaty, B. van Wachem, Artificial viscosity model to mitigate numerical artefacts at fluid interfaces with surface tension, Comput. Fluids 143 (2017) 59–72.
[46] A. Milne, A. Amirfazli, The Cassie equation: how it is meant to be used, Adv. Colloid Interface Sci. 170 (1) (2012) 48–55.
[47] A. Milne, A. Amirfazli, Drop shedding by shear flow or hydrophilic to superhydrophobic surfaces, Langmuir 25 (24) (2009) 14155–14164.
[48] P. Dadvand, R. Rossi, E. Oñate, An object-oriented environment for developing finite element codes for multi-disciplinary applications, Arch. Comput. Methods Eng. 17 (3) (2010) 253–297.
[49] C. Kelley, Iterative Methods for Linear and Nonlinear Equations, 1st edition, SIAM, 1995.
[50] F. Denner, B. van Wachem, Compressive VOF method with skewness correction to capture sharp interfaces on arbitrary meshes, J. Comput. Phys. 279 (2014) 127–144.
[51] C. Galusinski, P. Vigneaux, On stability condition for bifluid flows with surface tension: application to microfluidics, J. Comput. Phys. 227 (12) (2008) 6140–6164.
[52] D. Gerlach, G. Tomar, G. Biswas, F. Durst, Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows, Int. J. Heat Mass Transf. 49 (3–4) (2006) 740–754.
[53] S. Popinet, An accurate adaptive solver for surface-tension-driven interfacial flows, J. Comput. Phys. 228 (16) (2009) 5838–5866.
[54] H. Lamb, Hydrodynamics, 4th edition, Cambridge University Press, Cambridge, 1916, https://books.google.es/books?id=d_AoAAAAYAAJ.
[55] A. Prosperetti, Normal-mode analysis for the oscillations of a viscous-liquid drop in an immiscible liquid, J. Méc. 19 (1) (1980) 149–182.
[56] S. Popinet, S. Zaleski, A front-tracking algorithm for accurate representation of surface tension, Int. J. Numer. Methods Fluids 30 (6) (1999) 775–793.
[57] J. Kim, A continuous surface tension force formulation for diffuse-interface models, J. Comput. Phys. 204 (2) (2005) 784–804.
[58] A. Prosperetti, Motion of two superposed viscous fluids, Phys. Fluids 24 (7) (1981) 1217–1223.
[59] P. Causin, J. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid–structure problems, Comput. Methods Appl. Mech. Eng. 194 (42–44) (2005) 4506–4527.
[60] S. Idelsohn, F.D. Pin, R. Rossi, E. Oñate, Fluid–structure interaction problems with strong added-mass effect, Int. J. Numer. Methods Eng. 80 (10) (2009) 1261–1294.