A numerical model of two-phase flow at the micro-scale using the volume-of-fluid method

Mosayeb Shams *, Ali Q. Raeini, Martin J. Blunt, Branko Bijeljic

Department of Earth Science and Engineering, Imperial College London, London SW7 2AZ, UK

Article history:
Received 3 May 2017
Received in revised form 17 December 2017
Accepted 19 December 2017
Available online 21 December 2017

Keywords:
Two-phase flow
Micro-scale modelling
Volume-of-fluid method
Spurious currents

Abstract

This study presents a simple and robust numerical scheme to model two-phase flow in porous media where capillary forces dominate over viscous effects. The volume-of-fluid method is employed to capture the fluid-fluid interface whose dynamics is explicitly described based on a finite volume discretization of the Navier–Stokes equations. Interfacial forces are calculated directly on reconstructed interface elements such that the total curvature is preserved. The computed interfacial forces are explicitly added to the Navier–Stokes equations using a sharp formulation which effectively eliminates spurious currents. The stability and accuracy of the implemented scheme is validated on several two- and three-dimensional test cases, which indicate the capability of the method to model two-phase flow processes at the micro-scale. In particular we show how the co-current flow of two viscous fluids leads to greatly enhanced flow conductance for the wetting phase in corners of the pore space, compared to a case where the non-wetting phase is an inviscid gas.

© 2017 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Understanding multiphase flow through porous media is of great importance in a variety of environmental, industrial and engineering applications such as contaminant cleanup [35], fluid separation in fuel cells [48], enhanced oil recovery [29] and carbon dioxide storage in geological porous media [32]. However, accurate modelling and quantification of such flows at the pore level is challenging when interfacial tension effects become dominant. Inaccurate numerical modelling of the interfacial force may upset the balance of forces in the momentum conservation equation and lead to instabilities and unphysical results [54,16,17].

Popular approaches to simulate multiphase pore-scale processes in porous media include pore-network models [13,7,6,53], lattice Boltzmann models [47,40,2], mesh-free Lagrangian particle approaches such as smoothed particle hydrodynamics [36,49], and grid-based methods such as finite difference, finite volume or finite element. The grid-based methods are used in conjunction with various algorithms to model fluid–fluid interfaces, such as front-tracking [18,41], volume-of-fluid [22,42], level-set [39,1] and phase-field [4,3] models. Pore-network models are computationally efficient, but they need to simplify the geometry and physics of the problem. Lattice Boltzmann methods have the advantage of not requiring explicit interface tracking or contact angle models, but numerical instabilities may arise when simulating multiphase flows with large density and viscosity ratios [10]. Lagrangian mesh-free methods are robust, simple to implement and able to handle com-

* Corresponding author.
E-mail address: m.shams14@imperial.ac.uk (M. Shams).

https://doi.org/10.1016/j.jcp.2017.12.027
0021-9991/© 2017 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).
plex interface motions. However, they are not always computationally efficient when dealing with large three-dimensional systems [56].

Grid-based methods are conventionally considered as the preferred approach for most applications, due to their computational efficiency and ability to simulate fluid flow with large density and viscosity ratios [34]. One major challenge in modelling flows with moving interfaces in grid-based methods, however, is the need for an algorithm to model the interface including its location, shape and evolution with time. In front-tracking approaches, such as the height function method [23], the interface is determined explicitly as a sharp boundary and its motion is tracked on the computational domain. The main advantage offered by front-tracking methods lies in their capability to resolve the position of the interface to a high accuracy. However, explicit tracking of the motion of interfaces that undergo large distortions can be a challenging problem. Another class of algorithms represent the interface implicitly by marking the fluids on both sides of the interface, using either virtual particles (marker-and-cell methods) [21,33], or an indicator function such as a volume fraction (volume-of-fluid methods), a level-set function (level-set methods) or an order parameter (phase-field methods). One key advantage of these methods as opposed to front tracking is that they do not require complicated interface tracking algorithms. This is important for modelling two-phase flows through complex geometries such as three-dimensional micro-CT images of porous media in which we usually have large interface motions and interactions. Phase-field models flexibly capture complex interface motions and related physical phenomena by describing the interface as a thin transition region based on thermodynamic arguments. However, resolving the diffusive interface while convecting it in a mass-conservative fashion can be numerically challenging and computationally expensive [31]. Volume-of-fluid methods, unlike level-set methods, are intrinsically mass conservative. In addition, they are more computationally efficient compared to marker-and-cell methods [34]. These advantages, in addition to flexibility for treating complex interface configurations, have made volume-of-fluid a popular and powerful tool for the direct numerical simulation of immiscible two-phase flow [37,5,52,44,14]. In this work, we use a volume-of-fluid approach to capture interfaces.

Different methods have been employed to compute interfacial forces using implicit data from interface capturing approaches such as the volume-of-fluid method on a fixed grid. Continuum models such as the continuous surface force (CSF) method [9] and the continuous surface stress (CSS) method [19] are among the most widely used techniques that treat the singular interfacial force as a body force and concentrate it in the region of the interface using a delta function. The main problem with these models, however, is that they suffer from spurious currents near the interface especially when it comes to simulating flows in which the capillary force is dominant over viscous and gravity effects [28,26]. Another fundamental drawback is that, due to the lack of a sharp boundary between different fluids, these methods may not be able to ensure a zero net capillary force on closed interfaces [50]. The adverse effects of failing to impose this force conservation property have been reported in modelling of free rising bubbles in quiescent liquids and moving droplets in uniform velocity fields [45,42].

These issues have been addressed in many studies. Tryggvason et al. [50] employed a front-tracking method to reconstruct a sharp interface from volume fractions to compute a more accurate interfacial force on fixed grids. Scardovelli and Zaleski [46] used an interface tracking algorithm based on the piecewise linear interface construction method (PLIC) and a Lagrangian advection scheme to balance interfacial forces more accurately and reduce spurious currents to machine accuracy for two-dimensional calculations. Renardy and Renardy [44] developed a parabolic reconstruction of surface tension (PROST) algorithm for three-dimensional cases to model the force due to interfacial tension, which effectively reduced spurious currents in comparison to CSF and CSS. Jamet et al. [26] implemented an energy conserving discretization scheme, similar to the phase-field implementation of Jacqmin [25], to achieve a physical equilibrium state without spurious currents. He argued that conserving energy to round-off errors makes it possible to model fluid interfaces accurately, eliminating non-physical currents to machine accuracy, even if momentum is not strictly conserved.

In this paper, we use a face-based reconstruction algorithm to define the fluid–fluid interface as a sharp iso-contour surface to eliminate spurious currents effectively. Moreover, interfacial forces are computed directly on the reconstructed interface elements providing a force conservative interfacial tension model. We use interFoam, a volume-of-fluid based solver, in OpenFOAM [38] as the basis for implementing our numerical method. A modified fully sharpened indicator function is used to present the fluid–fluid interface as a sharp discontinuity [42]. A simple and effective algorithm is devised to reconstruct the interface from volume fractions only at the cell faces for which the gradient of the sharpened indicator function is non-zero. The reconstructed interface is then employed to compute the interfacial force at the centre of cell faces to give a balanced-force algorithm [16]. A primary advantage of this method is its ability to suppress spurious currents effectively and improve the force conservation property of the interfacial tension model even for low mesh resolutions. Herein, the term contour-level surface force (CLSF) method is used to refer to our approach for the treatment of the interfacial force.

In section 2, we present the governing mathematical equations. In section 3, we present the numerical implementation of the governing transport equations, the interface reconstruction algorithm and the interfacial force model in detail. The last section is dedicated to verification studies to ensure the correctness of the proposed numerical approach. The method is validated using several test cases. The process of equilibration of a static droplet in two and three dimensions is simulated to demonstrate the reduction of spurious currents (Section 4.1.1). The accuracy of the numerical method when liquid and solid phases come into contact is validated by modelling a wetting layer resting in the corner of a two-dimensional pore geometry with different corner and equilibrium contact angles (Section 4.1.2). The ability of the model to respect zero net capillary force on closed interfaces is investigated by simulating a micro-scale moving droplet in a uniform velocity field in two and three dimensions (Section 4.2.1). The dynamics of three-phase contact lines is examined by modelling the steady movement
of a contact line through two- and three-dimensional micro-channels (Section 4.2.2). Finally, the convergence behaviour of the method with spatial refinement is tested for corner flow of a wetting layer in a square capillary tube (Section 4.3). Here we demonstrate how co-current flow can lead to a greatly enhanced flow conductance when the non-wetting phase has a finite viscosity, as compared to the case where the non-wetting phase is an inviscid gas.

2. Mathematical model

The equations of motion for an isothermal, incompressible two-phase flow of Newtonian fluids, can be written using a single-fluid continuum approach as follows:

\[ \nabla \cdot \mathbf{u} = 0, \]
\[ \frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{uu}) = -\nabla p + \rho \mathbf{g} + \nabla \cdot \left( \mu \left( \nabla \mathbf{u} + \nabla^T \mathbf{u} \right) \right) + \mathbf{f}_{\sigma v}, \]

where \( \mathbf{u} \) is the velocity vector, \( p \) is pressure, \( \mathbf{g} \) is the gravitational acceleration, \( \mathbf{f}_{\sigma v} \) is the interfacial force concentrated on the interface, and \( \rho \) and \( \mu \) are the fluid density and viscosity, respectively, defined as

\[ \rho = \alpha \rho_1 + (1 - \alpha) \rho_2, \]
\[ \mu = \alpha \mu_1 + (1 - \alpha) \mu_2. \]

Here \( \alpha \) is an indicator function (physically representing the volume fraction of one phase) that has a value of one in the first fluid (subscript 1) and zero in the second fluid (subscript 2). In the volume-of-fluid method, \( \alpha \) represents volume fractions of one of the two fluids and varies smoothly between 0 and 1. The interface can be defined as the surface where the value of \( \alpha \) is equal to 0.5. The indicator function is evolved using the following advection equation:

\[ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) = 0. \]

The interfacial force per unit area, \( \mathbf{f}_\sigma \), at any point on the interface, assuming the interfacial tension, \( \sigma \), is constant, can be obtained as [51]

\[ \mathbf{f}_\sigma = \sigma \kappa \mathbf{n}, \]

where

\[ \kappa \mathbf{n} = \lim_{dS \to 0} \frac{1}{dS} \oint_C \mathbf{m} d\ell, \]

for an infinitesimal volume element \( dV \) enclosing an interface \( dS \) bounded by a closed contour \( C \) (Fig. 1). Here \( \kappa \) is the local interface curvature, \( \mathbf{n} \) is the local unit vector normal to the interface, \( d\ell \) is the length of an increment along the perimeter of the contour line, \( C \), and the vector \( \mathbf{m} = \mathbf{t} \times \mathbf{n} \) lies on the interface and is perpendicular to both the normal vector, \( \mathbf{n} \), and the unit vector \( \mathbf{t} \) tangent to the interface along \( C \). Finally, the interfacial force per unit volume, \( \mathbf{f}_{\sigma v} \), can be computed as

\[ \mathbf{f}_{\sigma v} = \mathbf{f}_\sigma \delta_s \]

where \( \delta_s \) is a delta function concentrated on the interface, i.e. is non-zero only at the interface points.

In the single-fluid approach, the fluid–fluid interface is treated as a discontinuity in fluid properties rather than an explicit boundary. Therefore we do not need to prescribe kinematic and dynamic boundary conditions explicitly at the fluid–fluid interface [15]. In our study, apart from bulk phase viscosity, \( \mu \), Eq. (3), we do not consider any additional hydrodynamic resistance to flow at the fluid–fluid interface, i.e., zero interfacial shear viscosity [20].
In contact line regions, where the fluid–fluid interface joins a solid boundary, wetting effects must be taken into account. This is achieved by modifying unit vectors normal to the interface at the solid boundary in accordance with a prescribed contact angle, $\theta$ (see Section 3.2). The contact angle is an input parameter and, in our numerical tests, it is set equal to the equilibrium contact angle.

3. Numerical method

The finite volume method, with a collocated arrangement of variables at centres of discrete non-overlapping control volumes [55], is used to numerically solve the governing equations of flow, Eqs. (1), (2) and (4). The flow algorithm in conjunction with a volume-of-fluid based interface tracking method, is taken from the interFoam solver of OpenFOAM [38, 27, 45]. In this section, first the algorithm employed to solve and couple the transport equations is briefly described for the sake of completeness, see Deshpande et al. [11] and references therein for more details. Then, we present a new numerical algorithm to reconstruct the location of the interface. Finally, the computation of the interfacial force on the reconstructed interface and its transfer to the underlying fixed mesh are discussed.

3.1. Numerical formulation of transport equations

The linearized semi-discrete form of the momentum equation can be written as

$$A_d \mathbf{u} = \mathbf{H} (\mathbf{u}) - \nabla p + \mathbf{F}, \quad \tag{8}$$

where $A_d$ is a scalar field containing the diagonal elements of the coefficient matrix, corresponding to a system of linear algebraic equations resulting from the discretization of the momentum equation, Eq. (2), the term $\mathbf{H} (\mathbf{u})$ accounts for off-diagonal elements of the coefficient matrix and also all source terms apart from pressure gradients and body forces, and $\mathbf{F}$ is any body force (in our case, only interfacial forces). A first-order, forward Euler method is used for the temporal discretization, while second-order finite volume schemes, based on Gauss’s theorem, are used for the spatial discretizations. The bounded self-filtered central differencing (SFCD) scheme is used to interpolate velocity and the indicator function from cell centres to face centres in the convective terms. Central differencing is used to compute required gradients normal to the cell faces.

By rearranging Eq. (8), we have

$$\mathbf{u} = \frac{\mathbf{H} (\mathbf{u}) + \mathbf{F}}{A_d} - \frac{1}{A_d} \nabla p. \quad \tag{9}$$

Imposing the continuity condition, Eq. (1), upon this velocity field, after discretization, leads to the following equation for the pressure:

$$\sum_f \frac{S_f}{(A_d)_f} \nabla^\perp_f p = \sum_f \tilde{\phi}_f, \quad \tag{10}$$

where

$$\tilde{\phi}_f = \langle \frac{\mathbf{H} (\mathbf{u})}{A_d} \rangle_f \cdot \mathbf{S}_f + \frac{\phi_{F,f}}{(A_d)_f}. \quad \tag{11}$$

Here, $\tilde{\phi}_f$ can be considered as an intermediate face-centred flux field that lacks the pressure gradient effect, $\mathbf{S}_f$ and $S_f$ are, respectively, the outward-pointing area vector of face $f$ and its magnitude, $\langle . \rangle_f$ denotes the face-centred field interpolated from the corresponding cell-centred field, $\nabla^\perp_f$ denotes face normal gradient evaluated as a scalar field directly at the face centres, and $\phi_{F,f} = F_f \cdot S_f$ is the interfacial force flux computed directly at the face $f$, which is the focus of discussion in Subsection 3.3.

Once the pressure field is obtained, a corrected face-centred flux field, $\phi_f$, is computed using the following equation:

$$\phi_f = \tilde{\phi}_f - \frac{S_f}{(A)_f} \nabla^\perp_f p. \quad \tag{12}$$

Finally, the cell-centred velocity vector for each cell $c$, $\mathbf{u}_c$, can be reconstructed in a conserved way from its face fluxes, $\phi_f$, as follows:

$$\mathbf{u}_c = S^{-1} \cdot \sum_{f \in c} \phi_f \mathbf{S}_f, \quad \tag{13}$$

where $S^{-1}$ is the inverse of the following second rank symmetric tensor:
\[ S = \sum_{f \in c} S_f S_f^T. \]  
(14)

Here \( \sum_{f \in c} \) is a sum operator over all faces belonging to the cell \( c \).

The major problem with advecting the indicator function using Eq. (4), is numerical diffusion, which tends to smear the interface sharpness through time. To alleviate this problem, an additional artificial convection term is added to the advection equation:

\[ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}) + C_u \nabla \cdot [\alpha (1 - \alpha) \mathbf{u}_c] = 0, \]  
(15)

where \( C_u \) is a compression factor, set to 1 in our simulations, and \( \mathbf{u}_c \) is the relative velocity vector [45,11]. The fluid–fluid interface is advected by solving the system of linear algebra arising from the discrete form of Eq. (15) for each cell \( c \):

\[ \bar{\partial}_t \alpha + \frac{1}{v_c} \sum_{f \in c} \left[ (\alpha_f \phi_f + C_u \langle \alpha_f \rangle_f (1 - \alpha_f) \phi_{f,c} \right] = 0. \]  
(16)

Here \( \bar{\partial}_t \) represents the first-order Euler method for the temporal discretization, \( v_c \) is the volume of cell \( c \), \( \phi_f \) is the volumetric flux at face \( f \) computed using Eq. (12), and \( \phi_{f,c} \) is a relative volumetric flux at face \( f \) defined as

\[ \phi_{f,c} = |\phi_f| (\mathbf{n}_f) \cdot \mathbf{n}_f, \]  
(17)

where \( \mathbf{n}_f = S_f/S_f \) is the unit normal to the face \( f \), and \( \mathbf{n}_c \) is the unit normal to the interface defined as

\[ \mathbf{n}_c = \frac{\nabla \alpha}{|\nabla \alpha|}. \]  
(18)

Eqs. (10), (12) and (16) are solved iteratively in two loops: (a) an outer-corrector loop to semi-implicitly couple the interface evolution and consequently the interfacial forces with the velocity field, and (b) an inner-PISO loop, to couple pressure and velocity fields, based on the pressure implicit with splitting of operator (PISO) method of Issa [24]. The solution algorithm from time step \( n \) to time step \( n + 1 \) can be summarized as follows:

1. Outer-corrector loop (\( n_{outerLoop} \) iterations):
   a. Advect the indicator function to update interface profile at time \( n + 1 \) (Eq. (16)), using the fluxes at time \( n \) for the first iteration.
   b. Update fluid properties, viscosity and density (Eq. (3)).
   c. Reconstruct the interface by linear approximation of the \( \alpha = 0.5 \) contour, and compute interfacial forces at face centres (details in Sections 3.2 and 3.3).
   d. Inner-PISO loop (\( n_{PISO} \) iterations):
      i. Compute the intermediate volumetric flux field, \( \tilde{\phi}_f \) (Eq. (11)).
      ii. Solve the pressure equation to update the pressure field (Eq. (10)).
      iii. Correct the volumetric flux field (Eq. (12)).
      iv. Reconstruct the cell-centred velocity field (Eq. (13)).
   e. End of inner-PISO loop.
2. End of outer-corrector loop.

We use two iterations in both the outer-corrector and the inner-PISO loops, \( n_{outerLoop} = n_{PISO} = 2 \), in our simulations.

3.2. Reconstructing the interface

Our motivation in reconstructing the interface is to reduce spurious velocities and calculate the pressure jump across the fluid–fluid interface more accurately within a finite-volume volume-of-fluid framework. The proposed algorithm is face-based, robust and easy to implement. We define the interface as face-based surface elements that consist of connected points where the value of \( \alpha \) is equal to 0.5. For the purpose of constructing a sharp interface, we first introduce a new indicator function by clipping \( \alpha \) and rescaling it between 0 and 1 as follows [42]:

\[ \alpha_c = \frac{1}{1 - C_c} \left[ \min \left( \max \left( \alpha, \frac{C_c}{2} \right), 1 - \frac{C_c}{2} \right) - \frac{C_c}{2} \right]. \]  
(19)

where the clipping factor, \( C_c \), controls the sharpness of \( \alpha_c \) in the vicinity of \( \alpha = 0.5 \). In this study, this factor is set to 0.98. Then, we declare a cell face to be a mixed-face, \( f_{\text{mix}} \), when for that face the following condition is satisfied (Fig. 2):

\[ |\nabla f \alpha_c| > \frac{\epsilon}{h}, \]  
(20)
where $\epsilon$ is a small number set to $10^{-8}$ and $h$ is the face normal distance between the centroid of the two cells that share the face $f$.

This technique converts the smooth profile of the indicator function, $\alpha$, into a sharp discontinuity, $\alpha_c$, without distorting the location of $\alpha = 0.5$. The sharpness of the discontinuity can be controlled using the clipping factor, $C_c$. As the clipping factor approaches 1, the discontinuity becomes sharper and the profile of $\alpha_c$ becomes closer to a stair-stepped approximation. Employing the face normal gradient of $\alpha_c$, instead of $\alpha$, to mark mixed-faces, i.e. the only faces on which the interfacial force is non-zero, increases the interface locality. Consequently, this improves the balance of the interfacial forces with the pressure gradient which can effectively eliminate spurious currents, as shown for static test cases in Section 4.

The smooth profile of $\alpha$ may get distorted due to the artificial compression of the interface during the advection process, Eq. (16), especially for dynamic test cases. This can adversely affect the accuracy of the interface normal vector computations which require a smooth change in the gradient of the indicator function. Therefore, we define a smoothed indicator function, $\alpha_s$, by interpolating $\alpha$ from cells to faces and back to faces, recursively, for four iterations ($i = 0–3$):

$$\alpha_{s,i+1} = C_s(\alpha_{s,i} f)_c + (1 - C_s)\alpha_{s,i}, \quad \alpha_{s,0} = \alpha,$$

where $(\cdot)_c$ denotes the cell-centred field interpolated from the corresponding face-centred field, and $C_s$ is the smoothing relaxation factor, which is set to 0.6 in our simulations. This smoothed indicator function is then used to obtain the interface normal vectors. This step may also include an optional prior clipping of $\alpha$ to exclude localized unphysical variations in $\alpha$ close to 0 or 1 in the vicinity of the interface zone,

$$\alpha_{s,0} = \min(\max(C_{c,0}(\alpha - 0.5) + 0.5, 0), 1),$$

where $C_{c,0}$ is a constant controlling the length of the clipping that is set to 1.02.

A second order least-squares method is used to estimate the gradient of $\alpha_s$ at mesh points, $p$, which form the mixed-faces:

$$(\nabla \alpha_s)_p = \sum_{i=1}^{N_f} \omega_i^2 \mathcal{G}_p^{-1} \cdot \mathbf{d}_i (\Delta \alpha_s)_i,$$

where $\mathcal{G}_p^{-1}$ is the inverse of the following symmetric tensor:

$$\mathcal{G}_p = \sum_{i=1}^{N_f} \omega_i^2 \mathbf{d}_i \mathbf{d}_i^T,$$

and

$$(\Delta \alpha_s)_i = \alpha_{s,i} - \alpha_{s,p}.$$

Here $N_f$ is the number of faces that share the point $p$, $\mathbf{d}_i$ is the distance vector between the centroid of face $i$ and the location of point $p$, $\omega_i$ is the inverse distance weighting factor calculated as $\omega_i = 1/|\mathbf{d}_i|$, $\alpha_{s,i}$ is the interpolated value of the smoothed indicator function, $\alpha_s$, at face $i$, and $\alpha_{s,p}$ is the interpolated value of $\alpha_s$ at the point $p$, computed using the following inverse-distance-weighting interpolation method:

$$\alpha_{s,p} = \frac{\sum_{i=1}^{N_f} \omega_i \alpha_{s,i}}{\sum_{i=1}^{N_f} \omega_i}.$$
Using a least-squares method to estimate \((\nabla \alpha_s)_p\) is preferable because it only uses the values of immediate neighbour cells of point \(p\), hence preserving its locality to a greater extent than the wide stencil finite difference approximations at cell centres followed by interpolation. Localized estimations of \((\nabla \alpha_s)_p\) are crucial to compute the interface location, \(\alpha = 0.5\), accurately.

Finally, the interface unit normal for each corner point \(p\) is defined as follows:

\[
\textbf{n}_p = \frac{(\nabla \alpha_s)_p}{|\nabla \alpha_s)_p|}. \tag{27}
\]

For the points on solid boundaries, fluid–fluid–solid contact points, however, the direction of \(\textbf{n}_p\) must be modified in accordance with the contact angle, \(\theta\), to obtain interface unit normal vectors at the contact line, \(\textbf{n}_p|_w\) (Fig. 3). The interface normals at the contact line, \(\textbf{n}_p|_w\), are obtained as a linear combination of \(\textbf{n}_p\) and the unit normal vector to the solid wall, \(\textbf{n}_w\),

\[
\textbf{n}_p|_w = (\cos \theta - \frac{\sin \theta \cos \theta'}{\sin \theta'})\textbf{n}_w + \frac{\sin \theta}{\sin \theta'}\textbf{n}_p, \tag{28}
\]

where

\[
\theta' = \cos^{-1}(\textbf{n}_w \cdot \textbf{n}_p). \tag{29}
\]

Here \(\textbf{n}_w\) is obtained by interpolating face unit normal vectors of the solid boundary to the mesh points. In the case of stair-like solid boundaries, used in Section 4.1.2, \(\textbf{n}_w\) is smoothed by recursively interpolating from the mesh points to their adjacent solid boundary faces and back to the mesh points [42], four times in this work.

For every mixed-face, an interface element is considered so that a one-to-one correspondence is established between the points of the mixed-face and the interface points. The location of the interface points, where \(\alpha = 0.5\), is calculated from a simple linear extrapolation (see Fig. 2(c))

\[
x_{\text{int.}p} = x_p + \frac{0.5 - \alpha_{s,p}}{|\nabla \alpha_s)_p|}\textbf{n}_p, \tag{30}
\]

where \(x_{\text{int.}p}\) is the position vector of the interface points, and \(x_p\) is the position of point \(p\) on the mixed-face.

Once the locations of the interface points are computed, the area vector of each interface element, \(\textbf{S}_{\text{int.f.}}\), is calculated using the following sum over its points:

\[
\textbf{S}_{\text{int.f.}} = \frac{1}{2} \sum_{i=1}^{N_p} (x_{\text{int.}p_{i+1}} - x_{\text{int.}p_i}) \times (x_{\text{int.}c} - x_{\text{int.}p_i}), \tag{31}
\]

where \(N_p\) is the total number of the corner points that form the interface element, and \(x_{\text{int.c}} = \frac{\sum_{i=1}^{N_p} x_{\text{int.}p_i}}{N_p}\) is the centroid of the interface element. The interface element points are ordered in a way that the computed area vector always points towards the same fluid (fluid 1).

Employing the least-squares method to estimate the interface normal at corner points alongside the one-to-one correspondence between corner points of interface elements and mixed-faces results in a simple and robust face-based methodology for reconstructing the interface. All that is required is a set of discrete marked faces described by their corner points. We stress that the clipped and smoothed indicator functions, \(\alpha_c\) and \(\alpha_s\), are only used to reconstruct the interface.

**Fig. 3.** A schematic representation of (a) unit normal vectors, \(\textbf{n}_p\), Eq. (27), to a fluid–fluid interface (dotted arrow) and to its contact point (solid arrow), on a solid surface, and (b) the modified interface unit normal vector at the wall, \(\textbf{n}_p|_w\), Eq. (28), which is in accordance with the imposed contact angle, \(\theta\).
elements, without affecting the original indicator function, $\alpha$. Moreover, the indicator function, $\alpha$, is advected in a conservative control-volume framework that guarantees mass conservation to round-off in all test cases; see the test case in Subsection 4.2.2, for an example.

3.3. Computing the interfacial force

In this section, we introduce our algorithm for computation of interfacial forces, the contour-level surface force (CLSF) model. Then, we briefly describe the previously-published continuous surface force (CSF) [9] and sharp surface force (SSF) [42] models. In Section 4, we show a comparison of the accuracy of the CLSF, CSF and SSF algorithms.

3.3.1. Contour-level surface force (CLSF) model

In this work, the interface is reconstructed and explicitly represented as discrete elements. The interfacial force for each interface element, $f_{\text{int}, \sigma}$, can be written as an integral of $f_{\sigma}$, Eq. (5), over the surface of the element and converted to a line integral using the Stokes theorem [50]:

$$ f_{\sigma, \text{int}} = \int_{S_{\text{int}}} f_{\sigma} ds = \int_{S_{\text{int}}} \kappa n ds = \int_{C_{\text{int}}} m d\ell, $$

where $C_{\text{int}}$ is the closed boundary along the edges of the interface elements. The interfacial force for each element is computed using this line integral over the edges of the interface element. This offers the unique advantage of not needing to calculate the interface curvature at cell centres and then interpolating them to faces, which improves the balance of forces in the momentum equation [50].

On the discrete level, the interfacial force on an interface element corresponding to face $f$, $f_{\sigma, \text{int}, f}$, is obtained by estimating Eq. (32) using the following sum over the corner points of the element:

$$ f_{\sigma, \text{int}, f} = \sum_{i=1}^{N_p} \left( x_{\text{int}, p_i+1} - x_{\text{int}, p_i} \right) \times \left( \frac{n_{p_j} + n_{p_j+1}}{2} \right), $$

(33)

The computed interfacial forces are then smoothed by looping over corresponding mixed-faces to equally distribute $f_{\sigma, \text{int}, f}$ between the two grid cells that share the mixed-face, and then interpolate the cell values back to the mixed-face based on an area-weighting method,

$$ f_{\sigma, \text{int}, f} = \sum_{i=1}^{2} \omega_i f_{\sigma, i}, $$

(34)

where

$$ \omega_i = \frac{s_{\text{int}, f}}{\sum_{j=1}^{N_{\text{max}}} s_{\text{int}, f_j}}, $$

(35)

and

$$ f_{\sigma, i} = \frac{1}{2} \sum_{j=1}^{N_{\text{max}}} |\Delta \alpha_c| f_{\sigma, \text{int}, f_j}, $$

(36)

Here, $\omega_i$ is the area-based interpolation weighting for cell $i$, $N_{\text{f, mix}}$ is the number of mixed faces belonging to cell $i$, and $|\Delta \alpha_c|_f$ is the magnitude of the difference of $\alpha$ between the two cells that share the mixed-face $f$. This smoothing step is essential to keep the conservation of the net capillary force on closed surfaces within an acceptable limit for our dynamic test cases (see Section 4.2), when applying the interfacial forces onto the underlying fixed grid.

Finally, the interfacial force flux term, $\phi_{f, f} = F_f \cdot S_f$, Eq. (11), is approximated on the fixed grid faces as follows:

$$ \phi_{f, f} = f_{\sigma, \text{int}, f} \cdot (n_f) \nabla_f^\top \alpha_c, $$

(37)

where $n_f$ is the interface normal computed using Eq. (18).

The steps taken in the CLSF method to compute the face-based interfacial force at each time step, can be summarized as follows:

1. Compute a sharp indicator function, $\alpha_c$ (Eq. (19)).
2. Find all mixed-faces, i.e. faces at the boundary between the two fluids, for which $\nabla_f^\top \alpha_c$ is nonzero (Eq. (20)).
3. Compute a smoothed indicator function, $\alpha_s$ (Eq. (21)).
4. Compute interpolated values of $\alpha_s$ at the mesh points (Eq. (26)).
5. For each mixed-face, do the following:
   a. Compute the gradient of $\alpha_s$ at the face vertices using a second order least-squares method (Eq. (23)).
   b. Obtain the interface normals at the vertices using the computed gradients (Eq. (27)). Modify the normal vectors on solid boundaries to include wettability effects (Eq. (28)).
   c. Estimate the location of the fluid–fluid interface, where $\alpha = 0.5$, in the direction of the interface normal vectors (Eq. (30)).
   d. Form interfacial surface elements using the computed interface location coordinates by holding a one-to-one correspondence between the mixed-faces and their interface elements (Eq. (31)).
   e. Compute the interfacial force on the interface elements (Eq. (33)).
6. Smooth the computed interfacial forces (Eq. (34)).
7. Use the computed interfacial forces to estimate the interfacial force flux on the corresponding mixed-faces on the fixed grid (Eq. (37)).

3.3.2. Continuous surface force (CSF) model

In the CSF model [9], the interfacial force is estimated as

$$ f_{sv} = \sigma \kappa \mathbf{n}_s \delta_s, \quad (38) $$

where $\delta_s$ is a delta function concentrated on the interface, $\mathbf{n}_s$ is the unit normal interface, and $\kappa = -\nabla \cdot \mathbf{n}_s$ is the interfacial curvature.

On the discrete level, to have a balanced-forced flow algorithm similar to Francois et al. [16], Eq. (38) is used to approximate the interfacial force flux term, $\phi_{F\cdot f}$, on the faces of the fixed-grid, as follows:

$$ \phi_{F\cdot f} = \sigma \left( \langle \kappa \mathbf{n}_s \rangle_f \cdot \mathbf{S}_f \right) \delta_{sf}, \quad (39) $$

where interfacial unit normals, $\mathbf{n}_s$, are computed at cell centres using Eq (18), and $\delta_{sf}$ is approximated as

$$ \delta_{sf} \approx \nabla^\perp_f \alpha. \quad (40) $$

3.3.3. Sharp surface force (SSF) model

In the SSF model, the same approach is employed as the CSF model, Eq. (38), however, the delta function in Eq. (39), $\delta_{sf}$, is approximated using the normal gradient of the sharp indicator function, $\alpha_s$, Eq. (19), rather than $\alpha$ [42]:

$$ \delta_{sf} \approx \nabla^\perp_f \alpha_s. \quad (41) $$

This implies that the SSF method employs a more compact delta function in comparison to the CSF method, while the interfacial normal vectors, $\mathbf{n}_s$, and consequently the interfacial curvature, $\kappa$, are estimated with a wide stencil as in the CSF method. We show that even though this approach may help to reduce spurious currents in some static test cases, it has the same limitations as CSF: large errors in pressure jump and consequently an unbalanced (non-zero) interfacial force on closed surfaces [45].

4. Model validation

In this section we use several two- and three-dimensional test cases to validate our numerical method. We first consider examples where the two phases are in static equilibrium (Section 4.1). Then, the accuracy of the method is examined using dynamic test cases (Section 4.2). Finally, to explore applications in porous media, corner flow through a three-dimensional non-circular capillary tube is investigated (Section 4.3).

In all simulations presented herein, for liquid–liquid systems we use two liquids, both with viscosities of $10^{-3}$ Pa·s and densities of 1000 kg/m$^3$ and for gas–liquid systems a gas phase with a viscosity of $10^{-5}$ Pa·s and a density of 1 kg/m$^3$. The interfacial tension is constant and taken to be $\sigma = 0.03$ N/m. We compare the results of the CLSF model to analytical solutions and also the CSF and SSF models.

4.1. Static test cases

A common test to validate two-phase flow models with interfacial tension, is simulating fluid–fluid systems at static equilibrium conditions where the velocity field must remain zero [41,44]. The discrete balance of the pressure gradient with the interfacial force can be examined by measuring the value of the maximum velocity in the computational domain. In addition, the accuracy of the interfacial force model can be quantified by measuring the error in pressure jump across the interface.
4.1.1. Static droplet

First, we simulate a two-dimensional circular droplet in the absence of gravity as either (a) a gas–liquid system (gas is surrounded by liquid) or (b) a liquid–liquid system. A static droplet with a radius of \( R = 10 \mu m \) is centred in a rectangular domain having side lengths of \( L = 40 \mu m \). The computational domain is discretized using a uniform Cartesian grid with a resolution of \( \delta x = 8 \), where \( \delta x \) is the side length of uniform grid cells. Since the interfacial force is treated explicitly, small time steps, \( \delta t \), are taken to satisfy the time step constraint

\[
\delta t \leq \left[ \frac{\overline{\rho} \delta x^3}{2 \pi \sigma} \right]^{1/2},
\]

where \( \overline{\rho} = \frac{\rho_1 + \rho_2}{2} \) \cite{9}.

Fig. 4 shows comparisons of calculated dimensionless measures of the maximum magnitude of the spurious currents. The spurious current capillary number, \( Ca = |\mathbf{u}|_{\text{max}} \mu / \sigma \), for the gas–liquid and liquid–liquid systems is shown as a function of a dimensionless time scale,

\[
t_d = \frac{\mu t}{\rho D^2},
\]

where \( \mu \) and \( \rho \) are, respectively, viscosity and density of the liquid, and \( D \) is the diameter of the droplet.

It can be observed that \( Ca \) reduces to machine precision, \( 10^{-15} \), in both systems using the CLSF method. For the SSF method, the velocity field also dies out to machine precision but with a slower convergence rate. On the other hand, for the CSF method, the maximum dimensionless error in the velocity field remains of order \( 10^{-3} \) for the gas–liquid and the liquid–liquid systems, which obviously is unfavourable when using the numerical method to model multiphase flow at low capillary numbers.

Fig. 5 shows the velocity vectors and droplet shapes (\( \alpha = 0.5 \) contour-line) at \( t_d = 0.25 \) for the liquid–liquid system using the three formulations. The velocity vectors are scaled by different factors to make them visible. The magnitude of the error in maximum velocity, Eq. (44), is given in Table 1. One can see that the structure and location of spurious vortices for the CSF method are different from those of the SSF and CLSF methods. For the SSF and CLSF methods, the spurious currents spread over a smaller area of the domain, due to employing a more compact delta function approximation than the CSF method, and are insignificant away from the interface to such an extent that they are effectively not present. This clearly shows the importance of implementing the interfacial force with a sharp force model to eliminate spurious currents efficiently.

The error in velocity is measured using the following \( L_\infty \) error norm:

\[
L_\infty(\mathbf{u}) = |\mathbf{u}|_{\text{max}},
\]

while the relative error of the pressure jump is estimated as

\[
E(\Delta P) = \frac{|\Delta P_{\text{numerical}} - \Delta P_{\text{exact}}|}{\Delta P_{\text{exact}}},
\]

The pressure jump across the droplet, according to the Young–Laplace equation, is \( \Delta P_{\text{exact}} = \sigma / R = 3000 \) Pa.
Fig. 5. Velocity vector field and the indicator function contour (α = 0.5) at dimensionless time \( t_d = 0.25 \), Eq. (43), for a two-dimensional droplet in static equilibrium (the liquid–liquid system): (a) the CSF method, (b) the SSF method, and (c) the CLSF method. The mesh resolution is \( R/\delta x = 8 \). Vector plots are scaled differently to be visible.

Table 1

Effect of mesh resolution on the error in maximum velocity, \( L_\infty(u) \), Eq. (44), and pressure jump across the interface, \( E(\Delta P) \), Eq. (45), for the liquid–liquid two-dimensional static droplet at a dimensionless time \( t_d = 0.25 \), Eq. (43), using the CSF, SSF and CLSF methods.

<table>
<thead>
<tr>
<th>( R/\delta x )</th>
<th>CSF ( L_\infty(u) ) (m/s)</th>
<th>( E(\Delta P) )</th>
<th>SSF ( L_\infty(u) ) (m/s)</th>
<th>( E(\Delta P) )</th>
<th>CLSF ( L_\infty(u) ) (m/s)</th>
<th>( E(\Delta P) )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>( 1.28 \times 10^{-2} )</td>
<td>0.170</td>
<td>( 1.09 \times 10^{-2} )</td>
<td>0.160</td>
<td>( 5.14 \times 10^{-9} )</td>
<td>0.013</td>
</tr>
<tr>
<td>8</td>
<td>( 5.29 \times 10^{-3} )</td>
<td>0.170</td>
<td>( 1.12 \times 10^{-9} )</td>
<td>0.160</td>
<td>( 3.76 \times 10^{-10} )</td>
<td>0.012</td>
</tr>
<tr>
<td>16</td>
<td>( 4.05 \times 10^{-3} )</td>
<td>0.130</td>
<td>( 8.80 \times 10^{-11} )</td>
<td>0.054</td>
<td>( 2.49 \times 10^{-10} )</td>
<td>0.011</td>
</tr>
<tr>
<td>32</td>
<td>( 8.04 \times 10^{-3} )</td>
<td>0.102</td>
<td>( 4.98 \times 10^{-11} )</td>
<td>0.013</td>
<td>( 8.48 \times 10^{-11} )</td>
<td>0.009</td>
</tr>
</tbody>
</table>

In Table 1, we investigate the results for the error in maximum velocity, \( L_\infty(u) \), and pressure jump, \( E(\Delta P) \), Eqs. (44) and (45) respectively, as a function of mesh resolution for the two-dimensional static droplet (liquid–liquid system) at \( t_d = 0.25 \). These results show that the CLSF method can predict the pressure jump accurately, with about 1% error, even at low resolutions, \( R/\delta x = 4 \) and \( R/\delta x = 8 \), while for the CSF and SSF methods the error is about 16% for these cases. This accuracy at low resolution is essential for modelling large problems and especially for three-dimensional calculations where the resolution may be low due to constraints imposed by high simulation costs. Further refinements to \( R/\delta x = 16 \) and \( R/\delta x = 32 \) lead to further reduction of the error in pressure jump for all methods. However, the magnitude of the error is still high, above 10%, for the CSF method. For the SSF method the error in pressure jump decreases dramatically from about 16% to 5% as the interface is resolved to \( R/\delta x = 16 \), whereas for the CSF method the reduction in the error is not that significant. It can also be observed that mesh refinement does not decrease the \( L_\infty \) norm of the spurious currents significantly in the CSF method, while for the SSF and CLSF methods spurious currents reduce to very small values, \( O(10^{-11}) \).

Next, we consider a three-dimensional droplet of radius \( R = 10 \mu m \) positioned in the centre of a uniform cubic mesh of resolution \( R/\delta x = 8 \). The system is taken to be the liquid–liquid system and the theoretical pressure jump across the interface is \( \Delta P_{\text{exact}} = 2\sigma/R = 6000 \text{ Pa} \).

Fig. 6 shows a visualization of the final shape of reconstructed fluid–fluid interface used to compute interfacial forces. Note that each interface element corresponds uniquely to one mixed-face on the fixed grid.

Fig. 7 depicts the evolution of the dimensionless maximum velocity for the three-dimensional test case using the CSF, SSF and CLSF methods. The spurious current capillary number is reduced efficiently to a very small value, \( O(10^{-13}) \), with the SSF and CLSF methods. On the other hand, similar to the two-dimensional case, \( Ca \) persists at the order of \( 10^{-3} \), for the CSF method.

The pressure profile for a cross-section passing through the centre of the droplet for the three methods are shown in Fig. 8. Improvements in both the accuracy and profile of the computed pressure with the CLSF method are evident. It yields a sharp pressure profile with a relative pressure jump error of 2%, while the CSF method leads to a smooth pressure distribution with a high relative pressure jump error of approximately 12%. For the SSF method, the pressure profile is not quite sharp, some transitional grid cells can be observed in comparison to the CLSF method, and the pressure jump error is 8%.

These results suggest that the CLSF method can predict the sharp pressure jump across the interface accurately, even for low resolution grids. In addition, it can virtually eliminate spurious currents in a stable manner for both the two- and three-dimensional static droplets.
Fig. 6. Visualization of (a) reconstructed fluid-fluid interface of a three-dimensional static droplet on a uniform Cartesian mesh of resolution $R/\delta x = 8$, at $t_d = 1.5$, and (b) the one-to-one correspondence between mixed-faces and reconstructed quadrilateral interface elements, shown for a quarter of the droplet. Arrows in (b) are the distant vectors from the mixed-face vertices to the corresponding interface element vertices.

Fig. 7. Dimensionless maximum velocity, the spurious current capillary number, as a function of dimensionless time, Eq. (43), for a three-dimensional static droplet when the interfacial force is modelled using the CSF, SSF and CLSF methods. The mesh resolution is $R/\delta x = 8$.

Fig. 8. Computed pressure field for a three-dimensional static droplet (cut through the centre of the droplet) at dimensionless time $t_d = 2$: (a) the CSF method, (b) the SSF method and (c) the CLSF method. The computed pressure jumps across the interface for the CSF, SSF and CLSF methods are 5395 Pa, 5548 Pa and 5824 Pa, respectively. The analytical value of the pressure jump across the interface for this case is 6000 Pa. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)
4.1.2. Static contact line

A two-dimensional domain with a star-shaped solid boundary is considered in this subsection. The domain is partially filled with a non-wetting phase while a wetting phase with the same fluid properties resides in the corners (Fig. 9). The grid cells which lie in a circle of radius 40 μm, centred in the computational domain, are initialized with the non-wetting phase while the rest (corner cells) contain the wetting phase. The aim is to test the ability of our model to simulate static immiscible fluid–fluid interfaces in contact with staircase-like solid walls, on a uniform mesh, with different corner and equilibrium contact angles, 2\(\gamma\) and \(\theta\), respectively. The accuracy and stability of the method are quantified in terms of the infinite error norm in velocity field and the normalized error in the pressure jump across the interface,

\[
E_\eta(\Delta P) = \left| \frac{\Delta P_{\text{numerical}} - \Delta P_{\text{exact}}}{\Delta P_{\text{exact}, \theta=0}} \right|,
\]

where \(\Delta P_{\text{exact}}\) can be calculated from

\[
\Delta P_{\text{exact}} = \frac{\sigma}{R_i},
\]

where \(R_i\) is the radius of the interface curvature in the corners. The wetting phase cross-sectional area, \(A_i\), as a function of \(R_i\) (Fig. 7(c)) is:

\[
A_i = \left( \sin \eta + \frac{\cos \eta}{\tan \gamma} \right) \cos \eta - \left( \frac{\pi}{2} - \eta \right) R_i^2,
\]

where \(\eta = \theta + \gamma\). The layer area, \(A_i\), is estimated numerically and plugged in Eq. (48) to obtain \(R_i\), which is in turn used in Eq. (47) to obtain \(\Delta P_{\text{exact}}\). Table 2 lists the simulation results for the CSF, SSF and CLSF methods at \(t_d = 2.5\), for different mesh resolutions, \(L/\delta x\), where \(L\) is the characteristic length of the geometry, see Fig. 9, which is taken to be 20 μm. One can observe that the errors resulted from the CSF and SSF methods do not follow a consistent behaviour. The accuracy of the results for these two methods is unsatisfactorily high, and apparently dependent on the values of the corner and equilibrium contact angles, which dictate the final configuration of the contact line with respect to the solid wall. On the other hand, the CLSF method reduces the spurious currents to machine accuracy and predicts the pressure jump across the interface to better than 8% in all cases.

Fig. 10 shows a sample visualization of the computed pressure field for the geometry with a corner angle of 2\(\gamma = 45^\circ\), an equilibrium contact angle of \(\theta = 30^\circ\), and a mesh resolution of \(L/\delta x = 8\). The improvement in the profile and magnitude of the pressure jump across the interface with our proposed method is clear.

These results show that the CLSF method is able to model a static contact line accurately in a liquid–liquid–solid system even with a stair-case-like discretization of solid boundaries.

4.2. Dynamic test cases

To complete our numerical investigations, we study two sets of dynamic test cases including (a) a moving droplet in a uniform velocity field (in two and three dimensions), and (b) two- and three-dimensional moving contact lines in micro-channels. The moving droplet test case is selected to test the ability of our method to conserve zero total force on a
Table 2

<table>
<thead>
<tr>
<th>L/δ (m)</th>
<th>θ (°)</th>
<th>CSF</th>
<th>SSF</th>
<th>CLSF</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 45 30</td>
<td>1.01 × 10^{-2}</td>
<td>0.199</td>
<td>4.78 × 10^{-10}</td>
<td>0.696</td>
</tr>
<tr>
<td>4 45 60</td>
<td>2.48 × 10^{-1}</td>
<td>0.432</td>
<td>4.11 × 10^{-6}</td>
<td>0.161</td>
</tr>
<tr>
<td>4 60 30</td>
<td>9.90 × 10^{-3}</td>
<td>0.225</td>
<td>3.47 × 10^{-2}</td>
<td>0.574</td>
</tr>
<tr>
<td>4 60 60</td>
<td>2.55 × 10^{-3}</td>
<td>0.454</td>
<td>5.78 × 10^{-3}</td>
<td>0.848</td>
</tr>
<tr>
<td>8 45 30</td>
<td>1.63 × 10^{-2}</td>
<td>0.120</td>
<td>4.45 × 10^{-11}</td>
<td>0.325</td>
</tr>
<tr>
<td>8 45 60</td>
<td>3.02 × 10^{-3}</td>
<td>0.344</td>
<td>8.57 × 10^{-11}</td>
<td>0.372</td>
</tr>
<tr>
<td>8 60 30</td>
<td>1.39 × 10^{-2}</td>
<td>0.178</td>
<td>3.51 × 10^{-6}</td>
<td>0.317</td>
</tr>
<tr>
<td>8 60 60</td>
<td>2.51 × 10^{-3}</td>
<td>0.494</td>
<td>5.46 × 10^{-11}</td>
<td>0.460</td>
</tr>
<tr>
<td>16 45 30</td>
<td>1.54 × 10^{-2}</td>
<td>0.108</td>
<td>1.12 × 10^{-3}</td>
<td>0.527</td>
</tr>
<tr>
<td>16 45 60</td>
<td>6.89 × 10^{-3}</td>
<td>0.227</td>
<td>3.28 × 10^{-11}</td>
<td>0.282</td>
</tr>
<tr>
<td>16 60 30</td>
<td>2.30 × 10^{-2}</td>
<td>0.026</td>
<td>9.13 × 10^{-5}</td>
<td>0.247</td>
</tr>
<tr>
<td>16 60 60</td>
<td>7.89 × 10^{-3}</td>
<td>0.511</td>
<td>2.37 × 10^{-11}</td>
<td>0.399</td>
</tr>
<tr>
<td>32 45 30</td>
<td>9.34 × 10^{-2}</td>
<td>0.045</td>
<td>1.25 × 10^{-11}</td>
<td>0.018</td>
</tr>
<tr>
<td>32 45 60</td>
<td>1.44 × 10^{-2}</td>
<td>0.356</td>
<td>1.23 × 10^{-11}</td>
<td>0.372</td>
</tr>
<tr>
<td>32 60 30</td>
<td>8.56 × 10^{-2}</td>
<td>0.058</td>
<td>2.12 × 10^{-2}</td>
<td>0.043</td>
</tr>
<tr>
<td>32 60 60</td>
<td>1.46 × 10^{-2}</td>
<td>0.505</td>
<td>1.79 × 10^{-11}</td>
<td>0.430</td>
</tr>
</tbody>
</table>

Fig. 10. Sample visualizations of the computed pressure field at dimensionless time $t_d = 2.5$, Eq. (43), for a corner angle of $2\gamma = 45^\circ$, and a mesh resolution of $L/\delta x = 8$: (a) the CSF method, (b) the SSF method, and (c) the CLSF method. The computed pressure jumps across the interface for the CSF, SSF and CLSF methods are 1307 Pa, 1893 Pa and 1113 Pa, respectively. The analytical value of the pressure jump across the interface for this case is 1127 Pa. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

closed interface as the indicator function is advected through the domain. This property is crucial, especially for capillary-dominated flows, where even a small error in the computation of interfacial forces can lead to a non-physical solution. The moving meniscus test case is considered to check the accuracy and stability of the new method to model the dynamics of immiscible fluid displacement in the presence of a moving contact line.

4.2.1. Moving droplet in a uniform velocity field

We model the steady movement of a liquid droplet of radius $R = 10$ μm in a uniform velocity field and with zero gravity. Theoretically, the droplet should move with the same velocity as the underlying uniform velocity field, and the pressure jump across the interface is $\Delta P = \sigma / R = 3000$ Pa and $\Delta P = 2\sigma / R = 6000$ Pa for two and three dimensions, respectively.

First, we investigate the effect of the velocity field, $U_{flow}$, on the movement of the droplet in a two-dimensional domain with a uniform grid of resolution $R/\delta x = 16$. Fig. 11 depicts the predicted dimensionless droplet velocity, $C_{d, droplet} = U_{droplet}/\sigma$, for different dimensionless flow velocities, $C_{d, flow} = U_{flow}/\sigma$, as a function of a dimensionless time, $t_d$:

$$t_d = \left( \frac{\mu U}{\rho D^2} \right)^{0.5} t,$$

where $U = U_{flow}$ and $D$ is the diameter of the droplet. It is clear that the CLSF method is able to predict the movement of the droplet accurately for capillary numbers as low as $10^{-5}$. Some relatively small fluctuations in the droplet velocity can be observed for $C_{d, flow} = 10^{-4}$ and $C_{d, flow} = 10^{-5}$. $C_{d, flow} = 10^{-2}$ is the lowest capillary number for which the CSF and SSF methods can predict the movement of the droplet accurately. For capillary numbers lower than $10^{-2}$, the error in the computation of interfacial forces in the CSF and SSF methods causes the violation of the zero net force constraint on the closed interface. This error manifests itself as large spurious currents which leads to the deviation of the droplet velocity.
from the theoretical value. Fig. 12 shows the initial configuration of the droplet and velocity vector plots and their computed values for $C_{a_{\text{flow}}} = 10^{-4}$ at $t_d = 6$ using different methods. One can observe that for the CSF and SSF methods the droplet is not able to move due to high spurious currents caused by error accumulation in front of the droplet. The related indicator function and pressure field are shown in Fig. 13 for the CLSF method. The pressure jump across the interface of the droplet accurately matches the analytical value of 3000 Pa. Next, we test the same moving droplet in three dimensions with a uniform grid of resolution $R/\delta x = 16$ for the CSF, SSF and CLSF methods. Fig. 14 shows the dimensionless average velocity of the droplet, $C_{d_{\text{droplet}}}$, as a function of dimensionless time $t_d$ for three different velocities using the CSF, SSF and CLSF methods. Similar to the two-dimensional simulations, the CLSF formulation can model the movement of the droplet for two orders of magnitude lower capillary number ($C_{a_{\text{flow}}} = 10^{-4}$) compared to that of the CSF and SSF methods ($C_{a_{\text{flow}}} = 10^{-2}$). Fig. 15 shows snapshots of the velocity vector plot, the indicator function profile and computed pressure field through a centre plane at dimensionless time $t_d = 3$. In the CLSF method the predicted pressure jump across the interface is highly accurate.

Finally, we compute the relative error in the average velocity of the two- and three-dimensional moving droplets, $\bar{U}_{\text{droplet}}$. Fig. 16 depicts the error in $\bar{U}_{\text{droplet}}$ over a period time of $t_d = 2$, as a function of mesh resolution, for different capillary numbers,

$$E(\bar{U}_{\text{droplet}}) = \frac{|\bar{U}_{\text{droplet}} - U_{\text{flow}}|}{U_{\text{flow}}}.$$  \hspace{1cm} (50)

The solution exhibits a positive convergence for relatively high capillary numbers, $C_{a_{\text{flow}}} = 10^{-2}$ and $C_{a_{\text{flow}}} = 10^{-3}$, for both two- and three-dimensional test cases. However, the order of convergence decreases as the capillary number decreases and we need finer meshes to obtain a solution within an acceptable error range. For the two- and three-dimensional moving
Fig. 12. A sample visualization of a two-dimensional simulation of a droplet moving in a uniform velocity field with $U_{\text{flow}} = 0.003 \text{ m/s}$ equivalent to a capillary number of $10^{-4}$. Shown in (a) is the initial shape and position of the droplet ($\alpha = 0.5$ contour-line). Shown in (b), (c) and (d) are velocity vector plot and position of the droplet at dimensionless time $t_d = 6$, Eq. (49), using the CSF, SSF and CLSF methods, respectively. The mesh is uniform and has a resolution of $R/\delta x = 16$.

Fig. 13. A sample visualization of: (a) indicator function, $\alpha$, and (b) computed pressure field using the CLSF method at dimensionless time $t_d = 6$, Eq. (49), for a droplet moving in a uniform velocity field with $U_{\text{flow}} = 0.003 \text{ m/s}$. The mesh is uniform and has a resolution of $R/\delta x = 16$. The analytical value of the pressure jump across the interface is 3000 Pa. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

droplet, the lowest capillary numbers for which the solution gives a stable and fairly accurate solution (error less than 20%) are, respectively, $Ca_{\text{flow}} = 10^{-5}$ and $Ca_{\text{flow}} = 10^{-4}$.

4.2.2. Steady contact-line displacement through a micro-channel

One of the most important characteristics of immiscible two-phase flow through porous media is the presence of three-phase contact lines where the three phases (fluid–fluid–solid) are in contact. Therefore, testing the ability of our numerical method to model contact line dynamics is crucial.

Here, a two-dimensional micro-channel of width $D = 20 \text{ μm}$, as shown in Fig. 17, is considered to investigate the steady movement of a contact line. A no-slip boundary condition is applied at the solid walls while the velocity at the inlet is set to a uniform fixed value in the $x$-direction. We apply a zero-gradient boundary condition for the pressure on all the boundaries and for the velocity at the outlet. The solid boundaries are assumed to be perfectly flat and homogeneous having a fixed equilibrium contact angle of $\theta = 30^\circ$. First, we investigate the contact line dynamics when the wetting phase
displaces the non-wetting phase for several steady injection velocities, $U_{\text{inj}}$, at the inlet, using a uniform grid of resolution $D/\delta x = 32$. A sample visualization of the velocity vector field and the shape of the interface ($\alpha = 0.5$) at dimensionless time $t_d = 0.5$, for the case when wetting fluid is injected at an injection capillary number of $Ca_{\text{inj}} = U_{\text{inj}} \mu / \sigma = 10^{-3}$, is shown in Fig. 17. The dimensionless time is defined based on Eq. (49) where $U = U_{\text{inj}}$ and $D$ is the width of the channel. Fig. 18(a) shows the dimensionless average velocity of the interface, $C_a_{\text{interface}}$, as a function of dimensionless time $t_d$ for a range of injection capillary numbers, $Ca_{\text{inj}}$. The interface translates accurately and its average displacement velocity is the same as the injection velocity. This implies that the numerical method is mass conservative and the effect of spurious currents on the overall displacement of the interface is insignificant. However, in our simulations, we observe that the contact line does not translate perfectly smoothly and experiences a periodic stick–slip motion, which introduces some small velocity fluctuations. The error resulted from these numerical spikes is quantified by using the following $L_2$ error norm:

$$L_2(Ca) = L_2(Ca_{\text{interface}} - Ca_{\text{inj}}).$$

and depicted as a function of dimensionless time $t_d$ in Fig. 18(b). The results suggest that the $L_2$ error norm increases as the capillary number decreases; however, the numerical errors do not introduce significant errors in the overall shape and movement of the interface. Moreover, Fig. 19 presents a mesh sensitivity analysis for the case with $Ca_{\text{inj}} = 10^{-3}$. It can be observed that the amplitude of the numerical spikes in the $L_2$ error norm decreases as the grid is refined.

Although the assumption of perfectly smooth solid walls can be useful to simplify the study of contact line dynamics, it should not be forgotten that solid surfaces in porous rocks are usually rough. Therefore, the roughness of the solid surface can lead to physical stick–slip behaviour at the micro-scale level. As shown in Fig. 20, the solid boundaries of the micro-channel are roughened using a triangular wave pattern characterized by roughness amplitude $\lambda = 225$ nm and period $P = 2 \mu m$. The flow domain is discretized using the mesh generator from OpenFOAM [38] that uses a uniform initial mesh.
Fig. 15. A sample visualization of a three-dimensional simulation of a moving droplet in a uniform velocity field of $U_{\text{flow}} = 0.03 \text{ m/s}$ equivalent to capillary number of $10^{-3}$ using the CLSF method. Shown in (a) is the initial position of the droplet ($\alpha = 0.5$). (b), (c) and (d) are snapshots of respectively, velocity vector, indicator function ($\alpha$), and computed pressure through a centre plane perpendicular to the flow direction at dimensionless time $t_d = 3$, Eq. (49). The mesh has a uniform resolution of $R/\delta x = 16$. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

Fig. 16. Plots of relative error in average velocity, $E(U_{\text{droplet}})$, Eq. (50), of (a) a two-dimensional moving droplet, and (b) a three-dimensional moving droplet, as a function of mesh resolution, $R/\delta x$, for different capillary numbers, $Ca_{\text{flow}}$. and snaps the boundary points to the solid wall, here the triangular wave roughness, so that all the solid boundary faces align with the solid wall. We also ensure that each edge of boundary faces connect to only one internal face.

Fig. 21 compares the $L_2$ error norm, Eq. (51), for the smooth and rough-wall micro-channels with $Ca_{\text{inj}} = 10^{-3}$. It is evident that the physical spikes have an amplitude of about two order of magnitudes higher than the numerical spikes in this test case.

Finally, we model the displacement of a meniscus through a three-dimensional square micro-channel, as shown in Fig. 22. We set the equilibrium contact angle to $\theta = 60^\circ$ and keep the other simulation parameters the same as in the two-
Fig. 17. Plots of (a) initial configuration of wetting (blue) and non-wetting (red) liquids in a two-dimensional micro-channel of width $D = 20 \, \mu m$, and (b) a snapshot of the velocity vector field and the interface shape ($\alpha = 0.5$ contour-line) at dimensionless time $t_d = 0.5$. The interface translates steadily through the micro-channel at dimensionless injection velocity of $C_{\text{inj}} = 10^{-3}$. The mesh resolution is $D/\delta x = 32$. The contact angle is $\theta = 30^\circ$ for all the solid boundaries and interfacial forces are computed using the CLSF method. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 18. Plots of (a) the dimensionless average interface velocity, $C_{\text{interface}}$, and (b) $L_2(Ca)$, Eq. (51), as a function of the dimensionless time, $t_d$, for steady contact line displacement in a micro-channel. The computational grid is a two-dimensional uniform mesh of resolution $D/\delta x = 32$. The contact angle is $\theta = 30^\circ$ for all the solid boundaries and the interfacial force is computed using the CLSF method. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)

Fig. 19. $L_2(Ca)$, Eq. (51), as a function of the dimensionless time, $t_d$, for a steady contact line displacement in a micro-channel using different mesh resolutions. The contact angle is $\theta = 30^\circ$ and the interfacial force is computed using the CLSF method. (For interpretation of the references to colour in this figure, the reader is referred to the web version of this article.)
Fig. 20. Plots of (a) initial configuration of wetting (blue) and non-wetting (red) fluids in a micro-channel of width $D = 20 \, \mu m$ with triangular roughness on its solid boundaries, and (b) a close-up view of the computational grid near the solid boundary. The amplitude and period of the roughness is chosen to be $A = 225 \, \text{nm}$ and $P = 2 \, \mu m$, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 21. Spikes in the velocity field presented in terms of $L_2(Ca)$, Eq. (51), as a function of the dimensionless time, $t_d$, for a steady contact line displacement through two-dimensional smooth and rough micro-channels on a grid of resolution $D/\delta x = 32$. The contact angle is $\theta = 30^\circ$ and the interfacial force is computed using the CLSF method.

Fig. 22. Plots of (a) initial configuration of wetting (blue) and non-wetting (red) liquids in a three-dimensional square micro-channel of width $D = 20 \, \mu m$, and (b) a snapshot of the velocity vector field and the interface shape ($\alpha = 0.5$ contour-line) at dimensionless time $t_d = 0.5$. The interface translates steadily through the micro-channel at dimensionless injection velocity $Ca_{inj} = 10^{-3}$. The mesh resolution is $D/\delta x = 32$. The contact angle is $\theta = 60^\circ$ for all the solid boundaries and interfacial forces are computed using the CLSF method. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

dimensional micro-channel test case. The dimensionless average velocity of the interface, $Ca_{interface}$, and $L_2(Ca)$, Eq. (51), as a function of dimensionless time $t_d$ for a different range of injection capillary numbers, $Ca_{inj}$, are depicted in Fig. 23. The results are similar to the two-dimensional case and in spite of some velocity fluctuations, especially for lower capillary numbers, the interface motion is stable.
These results show that the CLSF method can be applied to model capillary-dominant two-phase flow in the presence of solid boundaries where the dynamics of the contact line plays an important role.

4.3. Application to porous media

We now study two-phase flow problems typically encountered in subsurface rock geometries at the pore scale, namely flow of a wetting phase along the corners of a rock surface [8]. The dynamics of the corner flow of the wetting phase, especially at low capillary numbers, plays an important role in multiphase displacement process and also in the trapping and remobilisation of the non-wetting phase [30,43,12]. Therefore we examine the capability of our method to model the imbibition of a wetting phase in the corners of non-circular capillary tubes.

A 60 μm square capillary tube with a length of \( L = 300 \mu m \) is considered (Fig. 24) to study the spatial convergence of the dimensionless flow resistance factor of the wetting layer, \( \beta \) [43],

\[
\beta = -\frac{R^2}{\bar{\pi}_x \mu} \frac{\partial p}{\partial x},
\tag{52}
\]

where \( \mu \) and \( \bar{\pi}_x \) are the viscosity and axial average flow velocity of the wetting phase, respectively, \( \partial p/\partial x \) is the pressure gradient in the wetting phase along the axial direction, and \( R_x \) is the radius of the interface curvature in the corner, calculated based on Eq. (48), with \( \gamma = \pi/4 \) and \( \theta = \pi/6 \) as discussed in Section 4.1.2.

The initial liquid–liquid interface is a circular arc of radius \( r_0 = 30 \mu m \). The flow domain is discretized by a uniform Cartesian mesh of different resolutions. The no-slip boundary condition is applied at the solid walls. At the inlet and outlet boundaries, zero-gradient boundary conditions are applied for pressure and indicator function, \( \alpha \). Both fluids, with the same properties, viscosities of \( 10^{-3} \) Pa·s and densities of 1000 kg/m\(^3\), are injected at the inlet boundary in the \( x \)-direction, using a cyclic boundary condition; the velocity profile of the outlet boundary is replicated on the inlet boundary, with an initially uniform velocity of \( U_{inj} = 0.0003 \) m/s. In addition, no surface shear viscosity between the fluid phases and no gravity are considered. The interfacial tension is \( \sigma = 0.03 \) N/m and the equilibrium contact angle is \( \theta = 30^\circ \). In the results that follow, the simulations are conducted for the whole computational domain. However, the flow resistance is calculated and analyzed for only one corner of the tube.

Fig. 25 depicts the evolution of \( \beta \) as a function of time for five mesh resolutions. Through refining the mesh, the computed \( \beta \) converges to a value of about 52. Fig. 25(b) depicts the error in \( \beta \), at time \( t = 10 \) ms, defined as:

\[
E(\beta) = \frac{|\beta - \beta_{D/\delta x=54}|}{\beta_{D/\delta x=54}},
\tag{53}
\]

where \( \beta_{D/\delta x=54} \) is the evaluated dimensionless flow resistance on a mesh of resolution \( D/\delta x = 54 \). Here we observe a positive convergence behaviour for \( \beta \).

The computed dimensionless flow resistance in this test case, \( \beta = 52 \), is significantly lower than the value reported in Ransohoff and Radke [43] for the same geometry, \( \beta = 290.7 \). The main reason is that the dimensionless flow resistance in Ransohoff and Radke [43] is calculated for a gas–liquid system in which gas is assumed to be an inviscid non-wetting phase.
Therefore, the stress exerted by the non-wetting phase on the wetting phase is ignored. In our example, the co-current flow of the non-wetting phase boosts the flow of the wetting phase, leading to a much lower flow resistance.

We simulate wetting layer flow on a mesh of resolution $D/\delta x = 18$ for a case where the non-wetting phase is a gas with viscosity $\mu = 10^{-5}$ Pa·s and density $\rho = 1$ kg/m$^3$. Fig. 26 compares the computed dimensionless resistance factor for the gas-occupied and liquid-occupied pores as a function of time. Clearly, through reducing the stress exerted by the non-wetting phase on the wetting phase in the gas-occupied system, $\beta$ is increased to approximately 250, which is closer to $\beta = 290.7$ found by Ransohoff and Radke [43].

These results confirm that the numerical method can be used to model corner flow in non-circular capillary tubes to quantify their flow resistivity coefficients. However, the dynamics of corner flow can also be affected by fluid properties, here the viscosity of the non-wetting phase. A more extensive analysis of the effect of fluid and geometric properties on the corner flow will be the subject of further investigation.

5. Conclusions

A simple and efficient numerical method for modelling two-phase flow in porous media at the micro-scale has been devised. We compute interfacial forces using a sharp surface force model that can efficiently eliminate the problem of spurious currents. The explicit representation of the interface used to compute interfacial forces leads to an accurate estimation of capillary pressure even for low grid resolutions. This can be very important to model multiphase flows directly on micro-CT images of porous media for which simulations with a very fine grid resolution are prohibitively expensive using currently available computational resources.
In our test cases we have shown, for instance, how the consideration of co-current flow of two viscous fluids leads to a much lower flow resistance in layers than computed previously assuming that the centre of the pore space was gas-filled. This opens up the opportunity for future work where high-resolution pore-scale simulations are used to quantify capillary pressure and flow conductance for input into larger scale flow models.

Acknowledgements

This work is supported by the Imperial College Consortium on Pore-Scale Modelling. The authors would like to thank OpenFOAM developers and contributors for the use of their codes.

References