

Contact line treatment with the sharp interface method

Claudio Walker* and Bernhard Müller

Department of Energy and Process Engineering
The Norwegian University of Science and Technology, Trondheim, Norway
e-mail: claudio.walker@ntnu.no

Summary In the present study a method to handle contact points in a sharp interface method is presented. The contact points are tracked explicitly and their velocities are a function of the contact angle. For sharp interface methods an accurate approximation of the curvature is important. This cannot be obtained by using the conventional central difference methods adjacent to walls since the level set functions are not defined therein. Therefore we propose to use the tracked contact point and the intersection points of the first two grid lines parallel to the wall with the zero contour line of the signed distance function to approximate the contact angle and curvature adjacent to the walls.

Introduction

Drop impacts on dry surfaces can be found in many industrial and natural processes. Applications where the behavior of the impacting drops plays an important role include ink-jet printing, spray cooling, pesticide spraying, erosion processes due to rain and thermal spray coating. Probably the first study on droplet impacts was performed by [24]. An overview on the different phenomena which can take place during a drop impact is given in the reviews of [17] and [25].

One of the difficulties arising in modelling droplet impacts is the moving contact line problem. If the conventional no-slip boundary condition is applied to the Navier-Stokes equations, the stresses are diverging at the line where the three phases meet. In fact, molecular dynamics (MD) simulations show a near complete slip in the region of the contact line [10]. Although there are several empirical models connecting the dynamic contact angle at the contact line with the velocity of the contact line, a complete understanding of the mechanics at the contact line is still missing.

The surface tension plays an important role during the spreading and receding of an impacting drop. Therefore, accurate modelling of the surface tension in the continuum model is important. The surface tension introduces a jump in the solution across the interface. While this jump is smeared out to obtain numerically smooth solutions in diffuse interface methods, the jump conditions are imposed by sharp interface methods. The ghost-fluid method (GFM) [4], which uses a fixed Cartesian grid, was extended to incompressible two-phase flows [9]. The GFM allows to retain the jumps across the interface and therefore it is possible to eliminate the error which stems from the artificial smearing of the fluid properties. To use the GFM it is necessary to know the location of the interface, as well as its curvature. This information about the interface geometry can be obtained from the the level-set approach [23] [15]. However, if the interface crosses a wall boundary, i.e., if it forms a contact point, a special method to approximate the geometric information is required. Spelt [22] proposed a method to treat contact points in diffuse interface methods, where contact points are tracked explicitly. In the presented paper we present an extension of Spelt's method to a sharp interface method.

Equations

Navier-Stokes equations

We consider incompressible flow of two immiscible viscous fluids. In this study we confine ourselves to two-dimensional problems. There is no fundamental obstacle to extend the presented

method to three dimensions. The continuity and momentum equations, i.e., the Navier-Stokes equations for a Newtonian fluid for incompressible flow:

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

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) = -\nabla p + \mu \nabla^2 \mathbf{u} + \rho \mathbf{g} \quad (2)$$

where \mathbf{u} is the velocity vector, p is the pressure, \mathbf{g} is the gravity vector, μ and ρ are the dynamic viscosity and the density, respectively. The material properties μ and ρ can be different in each fluid and we use a + and – sign to discriminate between the two fluids. These equations have to be fulfilled in both fluids in two-phase flow.

Interface conditions

The boundary conditions at the interface can be derived by considering an infinitesimal control volume across the interface. This volume can be divided into a control volume for each fluid. The conservation laws for mass and momentum hold for the entire volume as well as for each of the partial volumes. Subtracting the sum of the conservation laws applied to the two partial volumes individually from the conservation laws applied to the entire volume, we get the boundary conditions at the interface. For the case with constant surface tension σ and no mass transfer across the interface we get the following jump conditions:

$$[\mathbf{u}] = 0 \quad (3)$$

$$\left[\begin{pmatrix} \mathbf{n}^T \\ \mathbf{t}^T \end{pmatrix} (p\mathbf{I} - \boldsymbol{\tau}) \mathbf{n} \right] = \begin{pmatrix} \sigma \kappa \\ 0 \end{pmatrix}, \quad (4)$$

where square brackets define the jump across the interface, e.g. $[\mathbf{u}] = \mathbf{u}^+ - \mathbf{u}^-$. We further denote \mathbf{n} and \mathbf{t} as unit normal and tangent vectors to the interface. κ is the local interface curvature and $\boldsymbol{\tau}$ is the viscous stress tensor. These conditions imply that the velocity and tangential stresses are continuous across the interface, whereas the pressure and the normal stresses are discontinuous.

Level set method

In order to apply the interface conditions presented in the previous section it is necessary to know its position. A popular method to keep track of the interface position in two-phase flows is the level set method (LSM) [19]. There the interface is defined as the zero contour line of a scalar function ϕ . Typically ϕ is the signed distance function from the interface and it exists and is continuous through the entire computation domain. The signed distance function is advected with the local fluid velocity using the advection equation

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0. \quad (5)$$

Since all discretisations of the advection equation will not be exact, ϕ loses its signed distance property over time and has to be reinitialised solving the following equation to steady state

$$\frac{\partial \phi}{\partial \tau} + \text{sign}(\phi) (|\nabla \phi| - 1) = 0. \quad (6)$$

The interface normal and curvature can be obtained directly from the signed distance function.

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|} \quad (7)$$

$$\kappa = -\nabla \cdot \mathbf{n} \quad (8)$$

Contact point

Around a point where the interface between two fluids meets a solid surface, the conventional no-slip boundary condition cannot be applied. For otherwise the stresses around the interface would become singular. To avoid this singularity the no-slip boundary condition is often replaced by a slip boundary condition of the following form:

$$u_{\parallel} = \lambda \left. \frac{\partial u_{\parallel}}{\partial x_{\perp}} \right|_{\text{wall}}, \quad (9)$$

where λ is the slip length and \parallel and \perp denote the parallel and normal directions relative to the wall. The difference in the surface energies can lead to a motion of the contact point. For small capillary numbers this effect can become extremely important. However, it is not possible to describe the dynamics around the contact point accurately using the Navier-Stokes equations. Some authors choose to fix the contact angle to model the dynamics around the contact line [1, 2, 5, 12]. In the present paper we shall set the contact point velocity u_{CP} as a function of the contact angle. To achieve this goal Spelt [22] proposed to track the contact point explicitly. Thus, the position of the contact point is described by an ordinary differential equation

$$\frac{dx_{\text{CP}}}{dt} = u_{\text{CP}} = f(\theta), \quad (10)$$

where $f(\theta)$ is an arbitrary function describing the dependency of the contact point velocity on the contact angle. The choice of f is important and should be done carefully to obtain good results. Different approaches to model the dependency of the contact point velocity on the contact angle are possible, including the use of empirical data or results from microscale simulations around the interface. In the present paper we use a simple linear relation as an example.

Discretisation

The discretisation is done on a uniform staggered grid, where the scalar quantities, i.e., pressure p and signed distance function ϕ , are stored at the cell centers while the velocity components u and v are stored at the vertical and horizontal cell faces, respectively [7]. The classical marker and cell (MAC) method is used to couple velocity and pressure in the discretisation of the incompressible Navier-Stokes equations (1) and (2).

Navier-Stokes

The advection terms in the Navier-Stokes equation (2) are discretised by a 5th order finite difference WENO scheme using a Lax-Friedrichs flux splitting [20]. For the WENO scheme can handle the discontinuities in the first derivative of the velocity automatically. In points which are not adjacent to the interface the viscous terms are discretised by second order central difference stencils, i.e.,

$$\left(\frac{\partial^2 u}{\partial x^2} \right)_{i-1/2,j} \approx \frac{u_{i+1/2,j} - 2u_{i-1/2,j} + u_{i-3/2,j}}{\Delta x^2} \quad (11)$$

$$\left(\frac{\partial^2 u}{\partial y^2} \right)_{i-1/2,j} \approx \frac{u_{i-1/2,j+1} - 2u_{i-1/2,j} + u_{i-1/2,j-1}}{\Delta y^2}. \quad (12)$$

The velocity component in y -direction is treated analogously. To enforce incompressibility a direct projection is applied. First an intermediate velocity \mathbf{u}^* is obtained by updating the velocity

from the previous time step with the advective, viscous and gravity terms.

$$\mathbf{u}^* = \mathbf{u} + \Delta t \left(-(\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{\mu}{\rho} \nabla^2 \mathbf{u} + \mathbf{g} \right) \quad (13)$$

This intermediate velocity field is then used as the right hand side of the Poisson equation for the scaled pressure correction $p^* = p/\Delta t$. Homogeneous Neumann boundary conditions are applied to solve for p^* .

$$\nabla \cdot \left(\frac{\nabla p^*}{\rho} \right) = \nabla \cdot \mathbf{u}^* \quad (14)$$

Away from the interfaces the density ρ corresponds to the constant density of the fluid we are in and the Laplace operator is approximated by the standard second order 5 point central finite difference stencil. Finally the intermediate velocity is made divergence free using the solution of the pressure Poisson equation.

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \frac{\nabla p^*}{\rho} \quad (15)$$

This projection procedure can be viewed as a special time splitting scheme which is advancing the solution Δt in time like one time step in a forward Euler scheme. Therefore a repetition of the projection procedure can be employed to form a Runge-Kutta time integration scheme. In the present work the 3rd order TVD Runge-Kutta method by Shu and Osher [21] is applied. An expression for the maximum allowed time step for a forward Euler time integration considering advection, diffusion, surface tension and gravity is given by Kang et al. [9]:

$$\frac{\Delta t}{2} \left((C_{\text{CFL}} + V_{\text{CFL}}) + \sqrt{(C_{\text{CFL}} + V_{\text{CFL}})^2 + 4S_{\text{CFL}}^2} \right) \leq 1, \quad (16)$$

where $C_{\text{CFL}} = \frac{|u|_{\text{max}}}{\Delta x} + \frac{|v|_{\text{max}}}{\Delta x}$, $V_{\text{CFL}} = \max \left\{ \frac{\mu^-}{\rho^-}, \frac{\mu^+}{\rho^+} \right\} \frac{4}{\Delta x^2}$ and $S_{\text{CFL}} = \sqrt{\frac{\sigma|\kappa|}{\min\{\rho^-, \rho^+\} \Delta x^2}}$.

Interface jump conditions

The jump conditions at the interface (4) can be rewritten such that jumps are separated for the derivatives of the velocity components [9].

$$\begin{aligned} \begin{pmatrix} [\mu u_x] & [\mu u_y] \\ [\mu v_x] & [\mu v_y] \end{pmatrix} &= [\mu] \begin{pmatrix} (\nabla u)^T \\ (\nabla v)^T \end{pmatrix} \begin{pmatrix} 0 \\ \mathbf{t}^T \end{pmatrix} \begin{pmatrix} 0 \\ \mathbf{t}^T \end{pmatrix} + [\mu] \mathbf{nn}^T \begin{pmatrix} (\nabla u)^T \\ (\nabla v)^T \end{pmatrix} \mathbf{nn}^T \\ &\quad - [\mu] \begin{pmatrix} 0 \\ \mathbf{t}^T \end{pmatrix} \begin{pmatrix} 0 \\ \mathbf{t}^T \end{pmatrix} \begin{pmatrix} (\nabla u)^T \\ (\nabla v)^T \end{pmatrix}^T \mathbf{nn}^T \end{aligned} \quad (17)$$

$$[p^*] = \Delta t \left(\sigma \kappa + 2[\mu] \begin{pmatrix} \nabla u \cdot \mathbf{n} \\ \nabla v \cdot \mathbf{n} \end{pmatrix} \cdot \mathbf{n} \right) \quad (18)$$

The jumps are continuous functions which are defined in the whole domain. Therefore they can be computed at the grid centers and then interpolated to the location where they are required. To compute the jump conditions one needs to approximate the derivatives of the velocity components at the cell centers. Kang et al. [9] choose to first interpolate the velocity components to the cell centers and then use standard central differences to approximate the gradients of the velocity. Here a different strategy is adopted, where the derivatives u_x and v_y are directly computed as the difference approximations of the velocities at the cell faces,

i.e., $u_x|_{i,j} = (u_{i+1/2,j} - u_{i-1/2,j}) / \Delta x$, and the other two derivatives namely u_y and v_x are approximated at the cell corners, i.e., $u_y|_{i+1/2,j+1/2} = (u_{i+1/2,j+1} - u_{i+1/2,j}) / \Delta y$ and those two derivatives are subsequently interpolated to the cell centers. Both methods lead to the same stencil for u_y and v_x , but for u_x and v_y the direct approximation results in a smaller stencil with the same order of accuracy. In addition we point out that u_x and v_y at the cell centers and u_y and v_x at the cell corners have to be computed anyway to approximate the viscous terms, cf. equations (11) and (12).

The separated jump conditions together with the ghost-fluid method [13] allow to treat the jumps in a sharp manner. If there is an interface crossing the grid lines of a stencil, the known jumps are added or subtracted from the points on the opposite side of the interface. This addition or subtraction is then included into the second order finite difference method. The treatment of the jump condition is improved by interpolating the jump condition to the place where the interface intersects with the grid lines. This procedure allows to discretise the Poisson equation

$$(\beta\varphi_x)_x + (\beta\varphi_x)_x = f(x) \quad (19)$$

with the interface conditions $[\varphi] = a$ and $[\beta\varphi_n] = b$ in the following way:

$$\begin{aligned} & \frac{\beta_{i+1/2,j} \left(\frac{\varphi_{i+1,j} - \varphi_{i,j}}{\Delta x} \right) - \beta_{i-1/2,j} \left(\frac{\varphi_{i,j} - \varphi_{i-1,j}}{\Delta x} \right)}{\Delta x} \\ & + \frac{\beta_{i,j+1/2} \left(\frac{\varphi_{i,j+1} - \varphi_{i,j}}{\Delta y} \right) - \beta_{i,j-1/2} \left(\frac{\varphi_{i,j} - \varphi_{i,j-1}}{\Delta y} \right)}{\Delta y} = f_{i,j} + F. \end{aligned} \quad (20)$$

F is a correction term for the jumps and therefore dependent on the interface position and the jump conditions a and b . If no interface crosses the stencil, F is zero and equation (20) reduces to the conventional second order central difference method. The correction appears only on the right hand side, which is beneficial since the same linear solvers can be used as for conventional central difference discretisations. If for example the interface intersects the left arm of the stencil, i.e., it is located between $x_{i-1,j}$ and $x_{i,j}$ and at $x_{i-1,j}$ we have the fluid with the material constant β^+ , the correction term reads:

$$F = \frac{\beta_{i-1/2,j} a_\Gamma}{\Delta x^2} - \frac{\beta_{i-1/2,j} b_\Gamma \Theta}{\beta^+ \Delta x}, \quad (21)$$

where $\Theta \cdot \Delta x$ is the distance between $x_{i-1,j}$ and the intersection point, and the subscript Γ indicates that the jump conditions are interpolated to the intersection point. In the context of two-phase flow this method is used for the viscous terms, where φ corresponds to the velocity components, $a = 0$ and b are the jump conditions from equation (17). For the pressure Poisson equation φ corresponds to the pressure p^* and the derivative is continuous, i.e., $b = 0$ and the pressure jump a is given by equation (18). For details concerning the GFM and its implementation for two-phase flows we refer the reader to the literature [9, 13].

Level set method

The gradients in the level set equations (5) and (6) are discretised by a 5th order finite difference WENO scheme [14]. To minimise spurious displacements of the interface during the reinitialisation the constrained reinitialisation CR-1 by Hartmann et al. [8] is applied. The normal is approximated by a conventional second order finite difference scheme. The expression for the

curvature (8) can be rewritten as:

$$\kappa = -\frac{\phi_x^2 \phi_{yy} - 2\phi_x \phi_y \phi_{xy} + \phi_y^2 \phi_{xx}}{(\phi_x^2 + \phi_y^2)^{3/2}}, \quad (22)$$

where the derivatives are approximated by second order finite differences as well.

Contact point treatment

The contact point is tracked explicitly with the ordinary differential equation (10). Immediately after the advection of ϕ the contact point is updated using the velocity which was computed from the contact angle θ at the last time step.

$$x_{\text{CP}}^{n+1} = x_{\text{CP}}^n + \Delta t \cdot f(\theta^n) \quad (23)$$

Since the updating of the contact point is a part of each Euler step in the projection algorithm, which forms one step of the Runge-Kutta scheme, the order of the temporal discretisation of the contact point position is consistent with the temporal order of the other equations.

The curvature κ and the contact angle cannot be computed with central differences at the wall since the stencil for the first grid point inside the fluid would contain points which are located inside the wall. As discussed in more detail below, the interface is assumed to continue as a straight line into the wall and the level set values at the ghost points are set as signed distances to that line. As a result the curvature at the first grid point would be compromised if computed by central differences. Instead a circle is fitted through the contact point x_{CP} and the intersection points of the interface with the first two grid lines parallel to the wall. To get convergent curvatures the intersection points of the grid lines and the zero level set contour are determined by a quadratic interpolation. The curvature at the first grid point is then given as the inverse of the radius R from the fitted circle. The smaller angle between the wall and the fitted circle is:

$$\hat{\theta} = \sin^{-1} \left(\frac{|x_{\text{CP}} - x_m|}{R} \right) \quad (24)$$

where x_m is the x -position of the center from the fitted circle, cf. Fig. 1. Using the y -position of the circle center and the sign of the first grid point left of the interface it can be checked whether $\hat{\theta}$ is the contact angle in the desired fluid and we set $\theta = \hat{\theta}$. Otherwise the correct angle is obtained by $\theta = \pi - \hat{\theta}$. In Figure 1 the fitted circle at the contact point is illustrated. During the reinitialisation the interface suffers from spurious displacement. Therefore the reinitialisation will change the measured contact angle and curvature since they depend on the first two interpolated intersection points. The spurious displacement and therefore the change in the contact angle and curvature can efficiently be reduced using a constrained reinitialisation.

The reinitialisation equation (6) can be rewritten as a hyperbolic equation where the characteristics are pointing perpendicular away from the interface [18]. Therefore the position of the interface decides where a boundary condition for the reinitialisation is required. In areas where no boundary condition is required the ghost points for the signed distance function can simply be filled with a linear extrapolation from the fluid points. To fill the ghost points in areas where a boundary condition is required, the interface is prolonged as a straight line from the contact point with a slope given by θ . For all ghost points which are below a line which is perpendicular to the interface at the contact point (shaded area in Figure 1) the distance to the prolonged interface is computed analytically. In addition the sign of the distance can be decided using

the first fluid point adjacent to the interface. To ensure that the characteristics are obeyed the ghost points are finally filled with either the extrapolated value or the analytical signed distance whichever has the smallest absolute value. If there are multiple contact points the analytically determined distance to the closest prolonged interface is used.

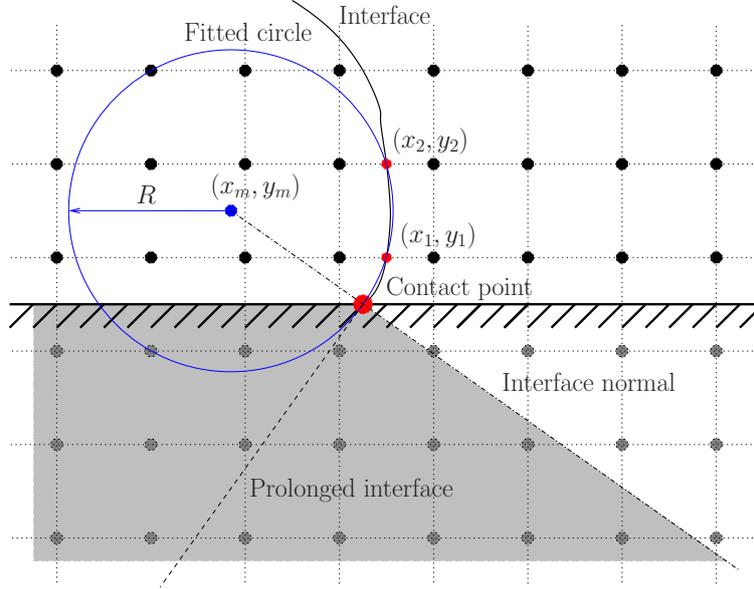


Figure 1: Illustration of the contact point treatment, in the shaded region the ghost points are filled with the analytical distance to the interface if this is smaller than the extrapolated value.

Results

To test our implementation of the GFM we first compute two examples of two-phase flow with dynamic interfaces for which analytical solutions exist. The ability of our method to handle contact points is demonstrated in an additional example of a droplet receding on a solid wall.

Oscillating droplet

An oscillating droplet surrounded by another fluid where both fluids are inviscid is a popular test case for multiphase solvers. An analytical solution to this problem can be found using potential theory [11]. Initially the droplet has the shape of an ellipse with a semimajor and a semiminor axis which is 5% longer and smaller than the radius $r = \frac{1}{3}$.

The surface tension $\sigma = 1$ N/m and the density $\rho^+ = \rho^- = 1$ kg/m³ will lead to an oscillation which is not damped in the inviscid case. The frequency of the oscillation is $\omega_0^2 = \frac{6\sigma}{(\rho^+ + \rho^-)r_0^3}$. Hansen [6] showed that the influence of the no-slip boundaries are becoming small if the domain becomes larger than 2×2 m². This was also verified in the present case. Therefore all the reported results were computed with a domain size of 2×2 m². The time step was chosen such that it is 1% of the maximum allowed time step and the number of grid points was $m = 52$, $m = 100$ and $m = 200$.

The time evolution of the semimajor axis is plotted in Figure 2. As the grid is refined the numerical solution approaches the exact solution. The amplitude of the semimajor axis is damped in the numerical results and the frequency is slightly too low as well. To analyse the influence of the time step we computed a solution on the coarsest grid with a 10 times larger Δt . It can

be seen in Figure 2 that the results are converging with decreasing time step. The area loss after two oscillation periods is 0.356%, 0.077% and 0.004% for the three grid resolutions. This is comparable to the results by Hansen [6].

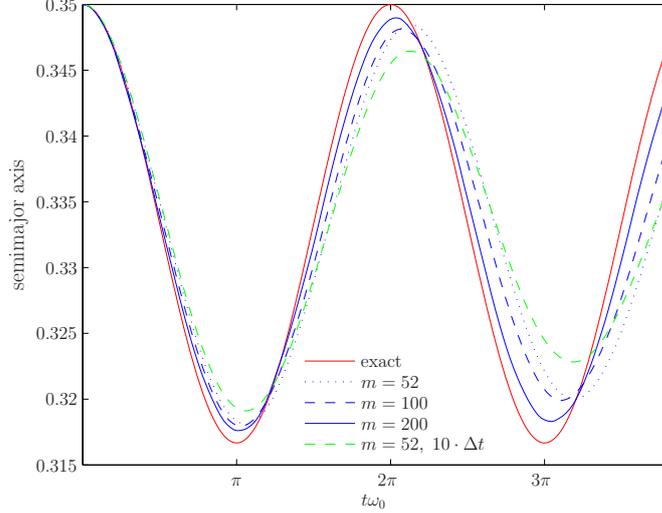


Figure 2: Time evolution of semimajor axis for the oscillating droplet.

Standing wave

Viscous damping of surface waves is used to verify the two phase solver for cases where viscous and surface tension forces interact. An analytical solution for this problem was presented by Prosperetti [16]. We use non-dimensional variables here. Initially two fluids are separated by a flat interface in the middle of the domain which is slightly perpetuated by a sine wave. The wavelength of the perturbation is $\lambda = 2\pi$ and the initial amplitude is set to $A_0 = 0.01\lambda$. According to the analytical solution the oscillation frequency is $\omega_0^2 = \frac{\sigma}{\rho^+ + \rho^-}$.

All solutions are obtained in a domain of size $2\pi \times 2\pi$. The boundary conditions parallel to the interface are slip walls, and perpendicular to the interface periodicity is enforced. Both densities are set to unity, the surface tension is $\sigma = 2$ and the viscosities are set to $\mu = 0.064720863$. For all reported grid resolutions the time step was set to $\Delta t = 0.01$.

Figure 3 shows the time evolution of the amplitude and the amplitude error. Again the frequency predicted by the numerical computations is too low, but converges with grid refinement. Opposed to the inviscid droplet oscillation, the damping of the oscillation amplitude is not strong enough. The RMS values of the amplitude errors are summarised in Table 1. A convergence rate between first and second order is observed. The RMS errors are slightly higher than those reported by Desjardins et al. [3]. The lower accuracy in our method could be caused by the completely explicit time integration, whereas a semi-implicit Crank-Nicholson scheme is employed by Desjardins et al.

Droplet on plate

Our last example is a droplet on a plate receding to its static contact angle. Again, non-dimensional variables are used. A droplet of radius $r_0 = 1.66$ and contact angle of 30° is placed on a solid surface. A different fluid is surrounding the droplet. The static contact angle is $\theta_s = 120^\circ$. This

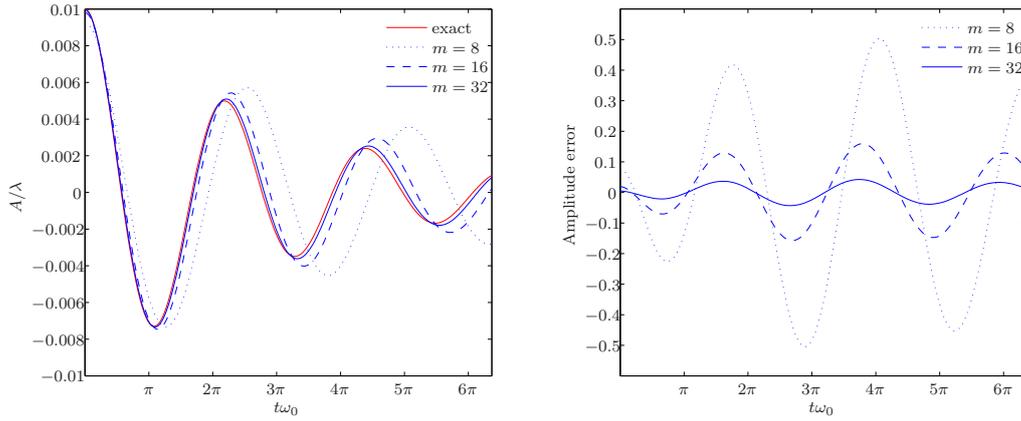


Figure 3: Time evolution of the wave amplitude (left) and amplitude error (right).

Table 1: RMS of amplitude error for the standing wave

Mesh	Error
8×8	0.2962
16×16	0.0958
32×32	0.0261

configuration will lead to moving contact points until the static contact angle is reached. In the steady state all velocities should vanish and the interface will assume a circular shape.

A rectangular computational domain of size $[-1, 1] \times [0, 1]$ is used. All boundaries are no-slip walls except for the bottom wall where the drop is placed on, here a slip boundary condition, cf. equation (9) with a slip length of $\lambda = 0.02$ is applied. The fluid parameters are chosen as in Case II of Spelt [22], i.e., $\rho^- = 1$, $\rho^+ = 20$, $\mu^- = \mu^+ = 0.0495$ and $\sigma = 2.21$, where the fluid denoted with $-$ is inside the droplet. The function of the contact point velocity is $u_{CP} = 0.1 \cdot (\theta - \theta_s)$. A time step of 0.7 times the maximum stable time step is chosen for all grid resolutions.

The velocity field at $t = 2$ is shown in Figure 4. During the receding of the contact points two vortices develop. The maximum absolute value of the velocity is always located around the contact points and the slip allowed from the boundary condition is clearly visible. In Figure 5 the time evolution of the interface is plotted. Since the Capillary number of the flow is relatively low $Ca = \frac{\mu u_{CP}}{\sigma} \leq 3.528 \cdot 10^{-3}$, the interface keeps a circular shape during the entire receding process. The evolution of the contact angle θ and the position of the right contact point x_{CP} as a function of time are presented in Figure 6. The contact angle and contact point position are almost identical for both grid resolutions. Initially the contact angle is small resulting in a high contact point velocity. This is causing a relative large curvature at the contact point which in turn leads to a relatively large pressure jump adjacent to the contact point. Therefore high slip velocities are produced. As the contact points are receding, the contact angle gets smaller and therefore the velocities are reduced as well. Finally the contact points approach their equilibrium

positions, the vortices are dissipated and steady state is reached.

At steady state the interface position error is computed by interpolating the position of all intersection points between the grid lines and the zero contour line using a 1-dimensional cubic spline interpolation. The position error of the interface $\varepsilon_{\text{interface}}$ is then given as the 2-norm of the distance between each interpolated intersection point and its correct position. The relative error of the contact point at steady state ε_{CP} is given as the difference between the contact point and its analytically position. Table 2 summarises the errors when the steady state is reached ($t = 24$). Since the interface is slightly displaced during reinitialisation of the signed distance function the curvature at the contact point is changed as well. This causes small spurious velocities around the contact point. The reported norms of the velocity are computed directly after reinitialisation, they will decrease until the next reinitialisation. Tests showed that they will converge to zero if no reinitialisation is applied after the steady state is reached. An important contribution to the errors of the interface and contact positions is the mass loss which occurs during the computation. The interface approaches the shape of a circle but with a radius which is too small. Often the signed distance function is lifted after reinitialisation to eliminate the mass loss. But this procedure would lead again to a spurious change in the curvature at the contact point. Therefore we did not adopt this practice in the presented examples.

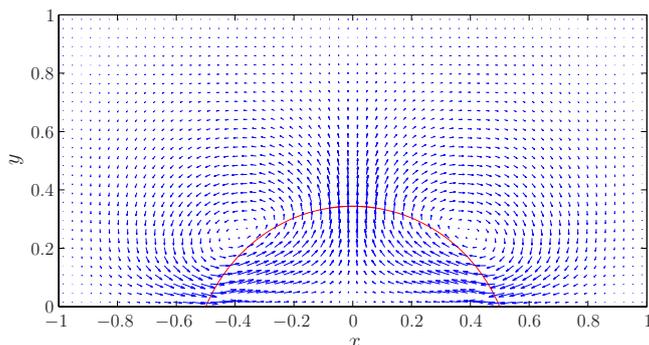


Figure 4: Velocity field at $t = 2$, the largest velocity 0.087 appears at the contact points.

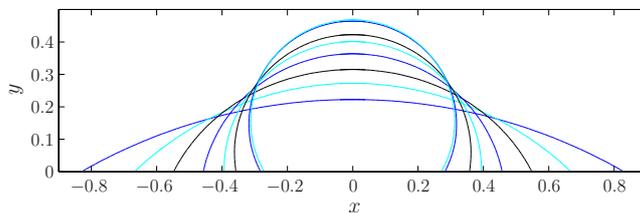


Figure 5: Interface at $t = 0, 1, 2, 3, 4, 5, 10$ and 20 .

Conclusions

We devise a way to handle contact points in sharp interface methods. The main difficulty is the computation of the curvature at the first grid point adjacent to the wall. A method is presented where a contact point is tracked explicitly. A circle is fitted through this contact point and the intersection points of the zero contour line of the signed distance function with the first two

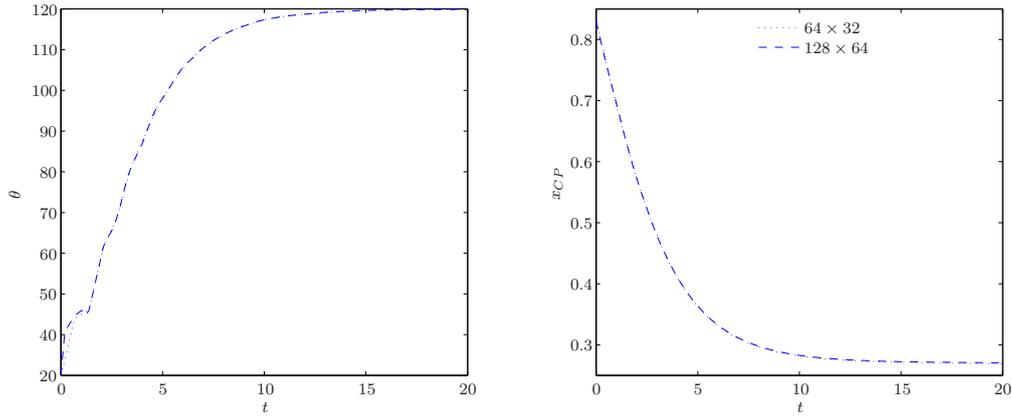


Figure 6: Time evolution of the contact angle (left) and the contact point position (right).

Table 2: Errors at $t = 24$ for a droplet on a plate.

	$\ u\ _2$	$\ v\ _2$	ε_{CP}	$\varepsilon_{\text{interface}}$
64×32	$0.719 \cdot 10^{-3}$	$0.858 \cdot 10^{-3}$	$13.359 \cdot 10^{-3}$	$5.320 \cdot 10^{-3}$
128×64	$0.391 \cdot 10^{-3}$	$0.381 \cdot 10^{-3}$	$7.903 \cdot 10^{-3}$	$3.353 \cdot 10^{-3}$

grid lines parallel to the wall is used to calculate the contact angle and the curvature adjacent to the wall. Numerical experiments show that the present two-phase solver based on the sharp interface method can correctly simulate dynamic interfaces and the steady state of contact lines for standard test cases, although an improvement is still possible.

Acknowledgements

The authors thank Martin Kronbichler and Prof. Gunilla Kreiss, Department of Information Technology, Uppsala University, Sweden, for the fruitful discussions on contact line dynamics.

References

- [1] M.Bussmann, S.Chandra and J.Mostaghimi Modeling the splash of a droplet impacting a solid surface *Physics of Fluids*, **vol.12**(12), 3121–3132, 2000.
- [2] D.Caviezel, C.Narayanan and D.Lakehal Adherence and bouncing of liquid droplets impacting on dry surfaces *Microfluidics and Nanofluidics*, **vol.5**, 469 – 478, 2008.
- [3] O.Desjardins, V.Moureau and H.Pitsch An accurate conservative level set/ghost fluid method for simulating turbulent atomization *Journal of Computational Physics*, **vol.227**(18), 8395–8416, 2008.
- [4] R. P.Fedkiw, T.Aslam, B.Merriman and S.Osher A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method) *Journal of Computational Physics*, **vol.152**(2), 457–492, 1999.
- [5] P. R.Gunjaj, V. V.Ranade and R. V.Chaudhari Dynamics of drop impact on solid surface: Experiments and vof simulations *AIChE Journal*, **vol.51**(1), 59–78, 2005.

- [6] E. B.Hansen *Numerical Simulation of Droplet Dynamics in the Presence of an Electric Field* PhD thesis, Norwegian University of Science and Technology, 2005.
- [7] F. H.Harlow and J. E.Welch Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface *Physics of Fluids*, **vol.8**(8), 2182, 1965.
- [8] D.Hartmann, M.Meinke and W.Schröder The constrained reinitialization equation for level set methods *Journal of Computational Physics*, **vol.229**(5), 1514 – 1535, 2010.
- [9] M.Kang, R. P.Fedkiw and X.-D.Liu A boundary condition capturing method for multiphase incompressible flow *Journal of Scientific Computing*, **vol.15**(3), 323–360, 2000.
- [10] J.Koplik, J. R.Banavar and J. F.Willemsen Molecular dynamics of fluid flow at solid surfaces *Physics of Fluids A: Fluid Dynamics*, **vol.1**(5), 781–794, 1989.
- [11] H.Lamb *Hydrodynamics* Cambridge University Press, 1932.
- [12] H.Liu, S.Krishnan, S.Marella and H.Udaykumar Sharp interface Cartesian grid method II: A technique for simulating droplet interactions with surfaces of arbitrary shape *Journal of Computational Physics*, **vol.210**(1), 32–54, November 2005.
- [13] X.-D.Liu, R. P.Fedkiw and M.Kang A boundary condition capturing method for poisson’s equation on irregular domains *Journal of Computational Physics*, **vol.160**(1), 151–178, May 2000.
- [14] X.-D.Liu, S.Osher and T.Chan Weighted essentially non-oscillatory schemes *Journal of Computational Physics*, **vol.115**(1), 200 – 212, 1994.
- [15] S.Osher and R.Fedkiw *Level Set Methods and Dynamic Implicit Surfaces*, volume 153 of *Applied Mathematical Sciences* Springer-Verlag New York, Inc., 2003.
- [16] A.Prosperetti Motion of two superposed viscous fluids *Physics of Fluids*, **vol.24**(7), 1981.
- [17] M.Rein Phenomena of liquid drop impact on solid and liquid surfaces *Fluid Dynamics Research*, **vol.12**(2), 61–93, August 1993.
- [18] G.Russo and P.Smereka A remark on computing distance functions *Journal of Computational Physics*, **vol.163**(1), 51 – 67, 2000.
- [19] J. A.Sethian and P.Smereka Level set methods for fluid interfaces *Annual Review of Fluid Mechanics*, **vol.35**(1), 341–372, 2003.
- [20] C.-W.Shu *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, chapter Essentially Non-oscillatory and Weighted Essentially Non-oscillatory Schemes for Hyperbolic Conservation Laws, pages 325–432 Springer Berlin / Heidelberg, 1998.
- [21] C.-W.Shu and S.Osher Efficient implementation of essentially non-oscillatory shock-capturing schemes *Journal of Computational Physics*, **vol.77**(2), 439 – 471, 1988.
- [22] P. D.Spelt A level-set approach for simulations of flows with multiple moving contact lines with hysteresis *Journal of Computational Physics*, **vol.207**(2), 389 – 404, 2005.
- [23] M.Sussman, P.Smereka and S.Osher A level set approach for computing solutions to incompressible two-phase flow *Journal of Computational Physics*, **vol.114**(1), 146–159, September 1994.
- [24] A. M.Worthington On the forms assumed by drops of liquids falling vertically on a horizontal plate *Proceedings of the Royal Society of London*, **vol.25**, 261–272, 1876.
- [25] A.Yarin Drop impact dynamics: Splashing, spreading, receding, bouncing *Annual Review of Fluid Mechanics*, **vol.38**(1), 159–192, 2006.