KTH Computer Science
and Communication

Numerical Modeling of Fluid Interface Phenomena

SARA ZAHEDI

Avhandling som med tillstånd av Kungliga Tekniska högskolan
framlägges till offentlig granskning för avläggande av teknologie licentiatexamen
onsdagen den 10 juni 2009 kl 10:00
i sal D42, Lindstedtsvägen 5, plan 4,
Kungliga Tekniska högskolan, Stockholm.

ISBN 978-91-7415-344-6
TRITA-CSC-A 2009:07
ISSN-1653-5723
ISRN-KTH/CSC/A–09/07-SE

Abstract

This thesis concerns numerical techniques for two phase flow simulations; the two phases are immiscible and incompressible fluids. The governing equations are the incompressible Navier–Stokes equations coupled with an evolution equation for interfaces. Strategies for accurate simulations are suggested. In particular, accurate approximations of the surface tension force, and a new model for simulations of contact line dynamics are proposed.

In the popular level set methods, the interface that separates two immiscible fluids is implicitly defined as a level set of a function; in the standard level set method the zero level set of a signed distance function is used. The surface tension force acting on the interface can be modeled using the delta function with support on the interface. Approximations to such delta functions can be obtained by extending a regularized one–dimensional delta function to higher dimensions using a distance function. However, precaution is needed since it has been shown that this approach can lead to inconsistent approximations. In this thesis we show consistency of this approach for a certain class of one–dimensional delta function approximations.

We also propose a new model for simulating contact line dynamics. We consider capillary dominated flows and assume that contact line movement is driven by the deviation of the contact angle from its static value. This idea is in this thesis adapted to the conservative level set method. By providing a diffusive mechanism for contact line movement, the need to allow fluid slip at the boundary is eliminated. Numerical experiments in two space dimensions show that we are able to capture contact line dynamics qualitatively correct.

Preface

This thesis consists of two parts. The first part, consisting of seven chapters, gives a background to the second part. The second part contains three papers in which the main contributions of this thesis are presented. The papers are listed below.

Paper 1
A conservative level set method for contact line dynamics,
S. Zahedi, G. Kreiss, and K. Gustavsson,
submitted to J. Comput. Phys..

The development of the ideas were done in close cooperation between the authors. The author of this thesis performed most of the computations and wrote a large part of the paper.

Paper 2
An interface capturing method for two–phase flow with moving contact lines,
S. Zahedi, G. Kreiss, and K. Gustavsson,
Proceedings of the 1st European Conference on Microfluidics, Bologna, 2008.

The development of the ideas were done in close cooperation between the authors. The author of this thesis performed most of the computations, wrote a large part of the paper and presented the paper at μFLU08.

Paper 3
Delta function approximations in level set methods by distance function extension,
S. Zahedi and A.-K. Tornberg,
manuscript to be submitted to J. Comput. Phys..

The author of this thesis prepared the manuscript, proved the theorems, performed the computations, and developed ideas with assistance from A.-K. T..

Acknowledgements

This thesis would not have been possible without help from many people. I am very grateful for the opportunity I have been given to work with Prof. Gunilla Kreiss. Your enthusiasm, experience, and knowledge make it easy to learn new things from you and I am glad that I have you as supervisor. I appreciate that you make things work out although we are at different universities.

During the work with the first paper in this thesis, I got to know Dr. Katarina Gustavsson which I am very happy for. You have always been very helpful and it has been very easy to work with you. Thanks! I would also like to acknowledge Dr. Anna–Karin Tornberg. At several occasions your knowledge in this field and discussions with you have truly helped me to improve my work and contributed to new ideas.

I am grateful to Dr. Johan Hoffman, Prof. Claes Johnson, Prof. Jesper Oppelstrup, and Prof. Anders Szepessy for taking time to discuss with me during the work with this thesis. I acknowledge all friends and colleagues in the numerical analysis group and the Linné Micro Complex group. You make it fun to come to work. Special thanks to Dr. Minh Do–Quang, Dr. Mohammad Motamed, Elin Olsson, Tomas Oppelstrup, and Jelena Popovic. I would also like to thank the people at Comsol Multiphysics which I have had interesting discussions with and hope for further collaboration in the future.

Without the enormous support I have been given by my parents and Dr. Emanuel Rubensson it had been impossible to write this thesis. Thanks for your confidence in me. Emanuel, I am grateful for your valuable comments which have helped me to significantly improve this thesis, and for our discussions. You have really contributed to this work although multiphase flow simulations have not been your subject. I have often got new ideas and solved many of the problems I had during the work with papers 1 to 3 by discussions with you.

Financial support from the Royal Institute of Technology (excellenstjänst), Danielssons foundation, Björns foundation, and the Lars Hierta Memorial Foundation is gratefully acknowledged.

Contents

Part I

Introductory chapters

Chapter 1

Introduction

There is a growing interest to accurately model and simulate multiphase flow phenomena. Such phenomena are important in a wide range of industrial applications. Two examples are liquid phase sintering and inkjet printing. An inkjet device consists of a long ink channel with a nozzle that ejects small droplets of liquid. With help from numerical simulations the size of droplets and ejecting speeds can be predicted. Such simulations are useful when new inkjet heads are to be designed.

Liquid phase sintering is an important process when manufacturing metal objects from powder. For example cutting tools are manufactured in this way. This process permits the formation of dense, pore–free carbides with superior properties such as high strength, hardness, and toughness. An important part of the sintering process is wetting of the liquid onto solid particles. The wetting is a consequence of the thermodynamic driving force to reduce interfacial energy, and results in elimination of porosity. The properties of the solid surface and the liquid will determine how the liquid wets the surface and how well the pores are eliminated. Numerical methods that are able to handle two phase flow with moving contact lines, are crucial for simulating the sintering process. Accurate simulations can contribute to improvements in the process, and hence to the development of better cutting tools.

In the multiphase models considered in this thesis immiscible fluids are separated by interfaces. These fluids may have different densities and viscosities. Surface tension forces act at the interfaces. An interface separating two fluids may also be in contact with a solid surface as in the liquid phase sintering where the liquid wets the solid surface. The line were the two fluids meet the solid is called a contact line. The representation and evolution of these interfaces, especially at solid surfaces, and the treatment of the discontinuous physical quantities, and the singular surface tension force, are some of the challenges in simulations of multiphase flow problems.

In the first part of this thesis we give an overview of some strategies to

these challenges. When nothing else is said we consider two space dimensions, were contact lines appear as contact points. In Chapter 2 we present some commonly used interface tracking techniques. In Chapter 3 the standard model for viscous, incompressible, immiscible fluids, the time dependent Navier–Stokes equations, are introduced. We also present the finite element technique for the discretization of these equations. In Chapter 4 different strategies of modeling the surface tension force are presented. The treatment of the discontinuous density and viscosity is discussed in Chapter 5. Models used for the interface movement at contact lines and points are studied in Chapter 6. The first part of this thesis ends with a discussion and future outlook in Chapter 7.

The second part of this thesis consists of three papers. In the first two papers we discuss a new model for contact line dynamics. In the third paper we consider approximations of the Dirac delta function supported on an interface. The Dirac delta function is important in this context as it is used to model the surface tension force.

Chapter 2

Interface representation and evolution

Interface tracking techniques have been developed to represent and track moving and deforming interfaces. Existing interface representation techniques can essentially be divided into two classes. In the first class, interfaces are represented explicitly; the interface can for example be defined by the use of so-called marker particles marking the interface. The immersed boundary method by Peskin used to model blood flow in the heart [1] is an example. Another example is the front–tracking method presented by Unverdi and Tryggvason for simulation of incompressible multiphase flow problems [2]. Since a large number of marker particles can be used good accuracy can be obtained. In the second class, there is no explicit representation of the interface. The interface is instead represented implicitly by a function defined on a higher dimension than the interface. This idea was used in the level set method introduced by Osher and Sethian [3]. The volume of fluid method [4] and the phase field method [5] are other examples. Compared to the methods tracking the interface explicitly these methods handle topological changes such as merging and breaking more easily. All such changes are seen as continuous evolution of the function defining the interface implicitly. There are also hybrid methods like the particle level set technique presented by Enright et al. [6] where the level set method is combined with a marker particle scheme.

In this chapter we discuss explicit and implicit representations of interfaces.

2.1 Explicit representation of the interface

The interface Γ can be represented explicitly by a set of markers $\{\mathbf{x}^{(l)}\}_{l=1}^{N} \in \mathbb{R}^2$ together with a parametrization $\mathbf{x}(s) = (X(s), Y(s))$ connecting these points. In two dimensions, the points can be connected by straight line segments. Higher order interpolation rules such as spline fits throw the points can also be used. In three dimensions three points can be connected by a triangular

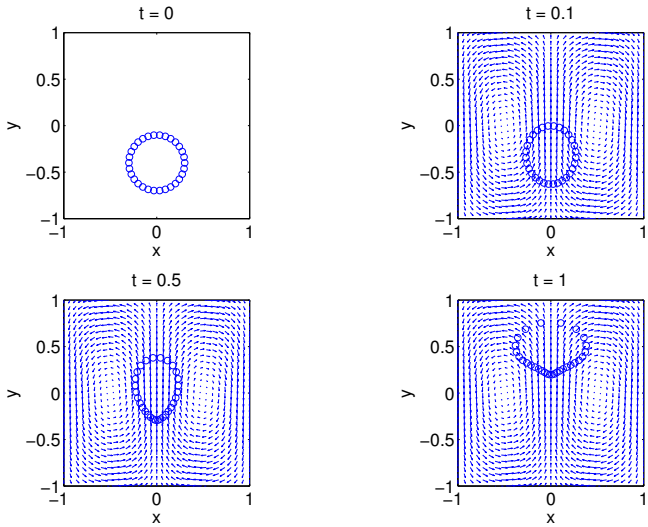


Figure 2.1: The interface initially a circle with radius 0.3 centered at $(0, -0.4)$ is explicitly defined by 30 markers and the parametrization $\Gamma = \{\mathbf{x} \mid \mathbf{x} = (0.3\cos(\theta), -0.4 + 0.3\sin(\theta),\ 0 \le \theta < 2\pi\}$. The grid velocities $u(x_i, y_j) = -\cos((x_i + \frac{1}{2})\pi)\sin(\frac{3\pi}{8}y_j)$, $v(x_i, y_j) = \sin((x_i + \frac{1}{2})\pi)\cos(\frac{3\pi}{8}y_j)$ are given. Interpolation is used to find the velocities at the Lagrangian markers. The interface is advected by solving equation (2.4). No reinitialization is performed. We can see how markers are depleted in some parts and clustered in other parts.

element. Using the parametrization of the curve

$$\Gamma = \{\mathbf{x} \mid \mathbf{x}(s) = (X(s), Y(s)),\ s \in [0, S),\ X(0) = X(S),\ Y(0) = Y(S)\}, \tag{2.1}$$

the normal vector $\mathbf{n}(s)$ and the curvature $\kappa(s)$ can be computed by

$$\mathbf{n}(s) = \frac{(-Y'(s), X'(s))}{\sqrt{(X'(s))^2 + (Y'(s))^2}} \tag{2.2}$$

and

$$\kappa(s) = \frac{X'(s)Y''(s) - X''(s)Y'(s)}{|(X'(s))^2 + (Y'(s))^2|^{3/2}}. \tag{2.3}$$

Given velocities at grid points, interface velocities $\mathbf{u}(\mathbf{x}^l, t)$ can be computed by interpolation. In the two–phase flow problem the velocity field is

usually given by solving the Navier–Stokes equations on an Eulerian grid. The interface is then evolved by advecting each marker by the local fluid velocity $\mathbf{u}(\mathbf{x}^l, t)$. This is done by solving the following ordinary differential equations

$$\frac{d\mathbf{x}^{(l)}}{dt} = \mathbf{u}(\mathbf{x}^l, t), \quad l = 1, \cdots, N. \tag{2.4}$$

As the interface evolves some parts may be depleted of markers and in other parts there may be clustering of markers, see Fig. 2.1. Therefore, a reinitialization step is usually needed. The computed parametrization of the interface is used here to decide where markers need to be added or removed. In two dimensions, markers can be added when the distance between two markers is larger than an upper bound and markers can be deleted when this distance is smaller than some lower bound. This procedure is straightforward in two dimensions but becomes complicated in three dimensions. The advantage of front–tracking algorithms is that a large number of markers can be used to represent the interface and therefore good accuracy can be obtained.

In the segment projection method introduced by Tornberg in Ref. 7 the interface is discretized explicitly by a set of segments. Functions of one variable in two dimensions and of two variables in three dimensions define the segments. Each segment is discretized on an Eulerian grid. The interface is evolved by solving partial differential equations with the same dimension as in front–tracking methods.

In the case of several curves or surfaces, merging and breaking of interfaces may occur. In explicit methods a separate representation for the interface of each curve or surface is used. Having separate representations has both advantages and drawbacks. On one hand, merging and breaking do not occur without explicit action. This allows for adding physically correct models. On the other hand, merging and breaking require complicated reconstructions of interfaces. Such reconstructions are particularly difficult in three space dimensions [8].

In the next section we describe methods where the interface is represented implicitly.

2.2 Implicit representation of the interface

Level set methods, volume of fluid methods and phase field methods are examples where the interface between immiscible fluids is defined implicitly by a higher dimensional function $\phi(\mathbf{x})$. This function is given by values on an Eulerian grid. In volume of fluid methods this so–called volume of fluid function gives the volume fraction of each fluid in each grid cell. The cells that are intersected by the interface will have a volume fraction between zero and

unity. The interface is reconstructed in each time step so that each cell obtains the correct volume fraction. In the first reconstruction algorithms, piecewise constant or "stair–stepped" approximations were used [4]. Later, higher order approximations such as piecewise linear functions [9] or splines [10] were proposed. The interface is evolved by updating the volume of fluid function. An advantage with the volume of fluid methods is that they conserve mass well. However, computations of accurate normal and curvature approximations are not straightforward. This is because the transition from one fluid to the other is very sharp.

In level set methods there is no reconstruction step. The interface is represented by a level set of a function. In standard level set methods [11,12] the interface Γ is defined as the zero level set of a signed distance function

$$\phi(\mathbf{x}) = \begin{cases} \text{dist}(\mathbf{x}, \Gamma) & \text{inside } \Gamma, \\ -\text{dist}(\mathbf{x}, \Gamma) & \text{outside } \Gamma. \end{cases} \tag{2.5}$$

Here $\text{dist}(\mathbf{x}, \Gamma)$ is the shortest distance between $\mathbf{x}$ and Γ. For example the zero level set of $\phi(\mathbf{x}) = r - \sqrt{(x - xc)^2 + (y - yc)^2}$ defines a circle of radius r centered at (xc, yc). Thus, the interface is embedded in a function $\phi(\mathbf{x})$ of higher dimension. The normal vector $\mathbf{n}$ and the curvature κ can be computed as

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|} \tag{2.6}$$

and

$$\kappa = -\nabla \cdot \mathbf{n}. \tag{2.7}$$

The normal vector defined as in equation (2.6) is pointing inward.

In level set methods you want a level set that defines the interface initially to also define the interface at a later time. Thus if for a point initially on the interface $\mathbf{x}(0) \in \Gamma$, $\phi(\mathbf{x}(0), 0) = C$ then for $\mathbf{x}(t) \in \Gamma$ at a later time t, $\phi(\mathbf{x}(t), t) = C$. Differentiating this condition with respect to t yields

$$\phi_t + \nabla \phi \cdot \mathbf{x}_t = 0. \tag{2.8}$$

A point on the interface should move according to the velocity field and thus $\mathbf{x}_t = \mathbf{u}$. The following partial differential equation is solved for the evolution of the interface

$$\phi_t + \mathbf{u} \cdot \nabla \phi = 0. \tag{2.9}$$

Given a divergence free velocity

$$\nabla \cdot \mathbf{u} = 0, \tag{2.10}$$

for example from solving the incompressible Navier–Stokes equations, the advection equation (2.9) can be written in conservative form

$$\phi_t + \nabla \cdot (\mathbf{u}\phi) = 0. \tag{2.11}$$

As the interface evolves in time the level set of ϕ defining the interface moves correctly according to equation (2.9) or (2.11). When ϕ is a signed distance function, then $|\nabla\phi| = 1$. However, the ϕ function may loose its original shape or properties. A signed distance function ϕ may no longer be a distance function after some time. In order to have accurate approximations of the normal and the curvature it is important to have a reinitialization step where essential properties of the function ϕ are recovered. There are several algorithms to create a signed distance function [13, 14]. Sussman et al. proposed in Ref. 13 to solve the following partial differential equations to steady state

$$\begin{aligned} &\psi_{\hat{t}} + \text{sign}(\psi_0)(1 - |\nabla\psi|) = 0 \\ &\psi(\mathbf{x}, 0) = \phi(\mathbf{x}, t^*) \end{aligned} \tag{2.12}$$

where $\phi(\mathbf{x}, t^*)$ is the level–set function at time $t = t^*$. Often a smeared out sign function is used [12].

Rider and Kothe [15] compared different interface tracking methods and found that the schemes that employed marker particles for the representation had better mass conservation properties. In the level set method mass is lost or gained when the interface is stretch or teared. These errors affect the accuracy of the computed location of the interface. Attempts to improve the mass conservation in level set methods have resulted in different methods [6,16–18]. In Ref. 16 the reinitialization scheme is constrained to conserve the volume bounded by the zero level set. However, the mass enclosed by the zero level set is not conserved in the advection step. Several hybrid method has also been developed with the aim of improving the mass conservation. Sussman and Puckett used a hybrid of the level set method and the volume of fluid method [17]. In Ref. 6 a hybrid method where the level set method is combined with a marker particle scheme was presented. However, in these methods the simplicity of the level set methods were lost and the methods become even more complicated and computationally expensive in three dimensions. In Ref. 18 the so–called conservative level set method was presented. The goal with this method was to preserve the good properties of the level set method but improve the conservation. This method is described in the next section.

2.2.1 The conservative level set method

The conservative level set method was introduced by Olsson and Kreiss in Ref. 18 and later in a finite element framework with an improved reinitialization in Ref. 19. In this method the interface is represented by the 0.5 level set of a regularized indicator function

$$\phi(\mathbf{x}) = \frac{1}{1 + e^{-d(\Gamma,\mathbf{x})/\varepsilon_n}}, \tag{2.13}$$

see Fig. 2.2. Here $d(\Gamma, \mathbf{x})$ is the signed distance function and ε_n determines the thickness of the regularized function.

The normal vector and the curvature are computed as in equation (2.6) and (2.7) respectively. The conservative level set method was formulated for incompressible flows. Therefore the interface is advected by solving the conservation law (2.11). The reinitialization step is formulated in conservative form as well:

$$\psi_{\hat{t}} + \nabla \cdot (\psi(1-\psi)\mathbf{n}) - \nabla \cdot (\varepsilon_n(\nabla\psi \cdot \mathbf{n})\mathbf{n}) - \nabla \cdot (\varepsilon_\tau(\nabla\psi \cdot \mathbf{t})\mathbf{t}) = 0, \quad (2.14)$$

$$\psi(\mathbf{x}, 0) = \phi(\mathbf{x}, t^*). \quad (2.15)$$

Here $\mathbf{n}$ is the normal vector satisfying (2.6) and $\mathbf{t}$ is the tangent vector and orthogonal to $\mathbf{n}$. Further, ε_n is a diffusion parameter in the normal direction and ε_τ is a diffusion parameter in the tangential direction. The second term in (2.14) represents compression in the normal direction, the third models diffusion in the normal direction, and the last term models diffusion in the tangential direction. A balance between the second and the third terms establishes a layer of thickness proportional to ε_n, where ϕ changes from 0 to 1. For the method to be accurate, ε_n must be much smaller than typical geometrical features of the interface. In the original work on the conservative level set method standard isotropic diffusion was used [18]. This corresponds to choosing $\varepsilon_\tau = \varepsilon_n = \varepsilon$ in equation (2.14). In the subsequent work, the tangential diffusion ε_τ was set to zero, i.e. $\varepsilon_\tau = 0$, to avoid unnecessary movement of the 0.5 level set in the tangential direction [19].

The reinitialization equation in the conservative level set method is in conservative form. This is in contrast to the reinitialization in equation (2.12) used in standard level set methods. Therefore, using conservative numerical methods to obtain ϕ_h and no flux through the boundaries we get a conserved integral:

$$\frac{d}{dt}\left(\int_\Omega \phi_h(\mathbf{x}) d\mathbf{x}\right) = 0. \quad (2.16)$$

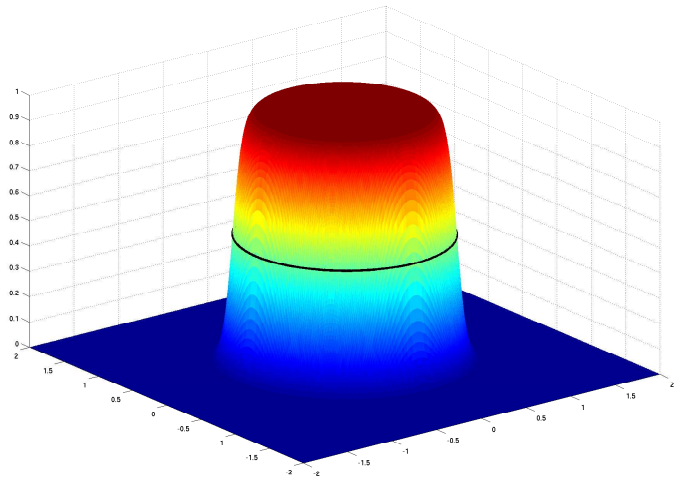
Now, since the ϕ function is a regularized indicator function this integral is almost equal to the area bounded by the interface $\phi_h(\mathbf{x}) = 0.5$. In Ref. 19 it was shown that the analytical error

$$A_{\phi_h=0.5}(t_n) = A_{\phi_h=0.5}(0) + \delta(t_n), \quad (2.17)$$

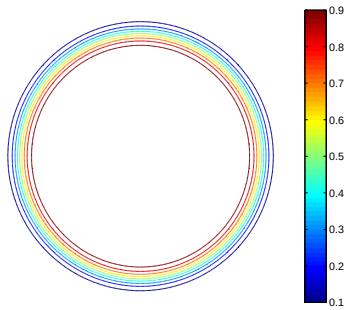
where

$$|\delta(t_n)| < 2L_\Gamma(t_n)|\kappa(t_n)|_\infty \varepsilon^2. \quad (2.18)$$

Here $A_{\phi_h=0.5}$ denotes the area inside the 0.5 level set of ϕ_h, $L_\Gamma(t_n)$ is the length, and $\kappa(t_n)$ is the curvature of the interface Γ at time $t = t_n$. When the interface Γ is a straight line the area is conserved as long as equation (2.16) is satisfied.



(a) $\phi(\mathbf{x})$



(b) Level sets of $\phi(\mathbf{x})$

Figure 2.2: The interface is a circle. The 0.5 level set of a regularized indicator function $\phi(\mathbf{x})$ defines the interface. The indicator function takes the value 0 in one fluid and the value 1 in the other fluid. In Panel (a) $\phi = \frac{1}{1+e^{-(1-\sqrt{x^2+y^2})/\varepsilon_n}}$ is shown. ε_n is the regularization parameter. The interface thickness is proportional to ε_n. Panel (b) shows level sets of $\phi(\mathbf{x})$.

In Paper 1 we have extended the conservative level set method to be able to simulate moving contact points. This extended method is also used in Paper 2. For an introduction, see Chapter 6. In the next chapter we introduce the Navier–Stokes equations which are the standard equations used in two–phase flow problems.

Chapter 3

The Navier–Stokes equations

Assume that a given domain $\Omega \in \mathbb{R}^2$ is occupied by two immiscible fluids separated by an interface Γ, for example a water drop surrounded by oil. The domains occupied by each fluid at time $t \in [0,\ T]$ are denoted $\Omega_1 = \Omega_1(t)$ and $\Omega_2 = \Omega_2(t)$ respectively, see Fig. 3.1. In this chapter, we assume that the interface is not in contact with the boundary of the domain $\partial\Omega$. We will consider the problem of moving contact points in Chapter 6. The density $\rho(\mathbf{x})$ and the viscosity $\mu(\mathbf{x})$ of each fluid is given by

$$(\rho(\mathbf{x}), \mu(\mathbf{x})) = \begin{cases} (\rho_1, \mu_1) & \text{for } \mathbf{x} \text{ in fluid } \Omega_1, \\ (\rho_2, \mu_2) & \text{for } \mathbf{x} \text{ in fluid } \Omega_2. \end{cases} \tag{3.1}$$

These physical quantities are in general discontinuous functions. Any method designed for multiphase flow problems must be able to handle discontinuities. We consider the treatment of discontinuous density and viscosity in Chapter 5.

The standard model for each domain (Ω_1 and Ω_2) is the time dependent incompressible Navier–Stokes equations with the following boundary condition at the interface separating the two fluids

$$[\mathbf{u}]_\Gamma = 0. \tag{3.2}$$

This condition ensures that the velocity field is continuous across the interface. At the interface separating two immiscible fluids, surface tension forces are acting. The surface tension drives fluid surfaces to have minimum surface energy. If the surface tension coefficient σ has spatial variations fluid can flow from regions of lower to higher surface tension. We will throughout this report consider only interfaces with a constant surface tension coefficient σ. To account for the surface tension effects, the surface stress boundary condition

$$[(-pI + \tau)\,\mathbf{n}]_\Gamma = \sigma\kappa\mathbf{n}, \quad \tau = \mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right) \tag{3.3}$$

has to be added. Here τ is the viscous stress tensor for incompressible fluids, I is the identity tensor, $\mathbf{n}$ is the unit outward normal vector on Γ and κ is

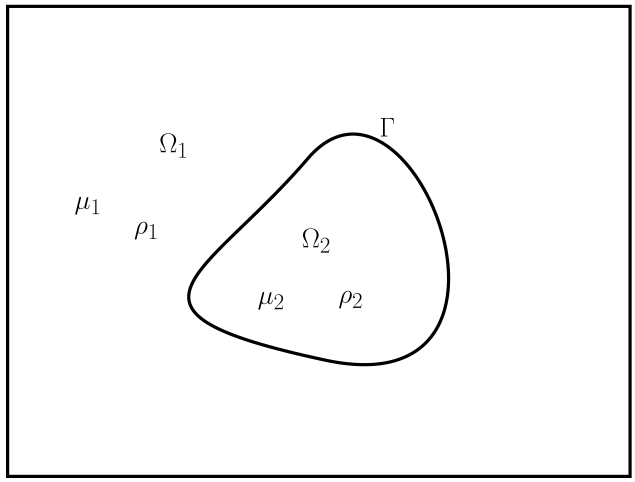


Figure 3.1: The domain $\Omega \in \mathbb{R}^2$ is occupied by two immiscible fluids separated by an interface Γ. The domains occupied by the fluids are denoted Ω_1 and Ω_2. Here $\partial\Omega_2 \cap \partial\Omega \neq \emptyset$. Density is denoted by ρ and viscosity by μ.

the curvature of Γ [20]. Thus, the surface tension balances the jump of the normal stress on the interface.

Instead of solving two equations, one in Ω_1 and another in Ω_2, with the coupling conditions (3.2) and (3.3) at the interface we can reformulate and solve the Navier–Stokes equations in the whole domain. The effect of surface tension can then be expressed in terms of a Dirac delta function δ_Γ with support on the interface Γ, the so–called continuum surface force model [1, 21, 22]. This force is given by

$$\mathbf{f} = \sigma\kappa\mathbf{n}\delta_\Gamma. \tag{3.4}$$

The action of δ_Γ on a smooth function v is given by

$$\int_\Omega \delta_\Gamma v d\Omega = \int_\Gamma v d\Gamma. \tag{3.5}$$

Different methods to approximate the Dirac delta function and the surface tension force are discussed in Chapter 4.

The Navier–Stokes equations in conservative form with gravity and surface tension added as source terms are defined as

$$(\rho\mathbf{u})_t + \nabla\cdot(\rho\mathbf{u}\mathbf{u}) + \nabla p - \nabla\cdot\left(\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) = \rho g\mathbf{e}_g + \mathbf{f}, \tag{3.6}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{3.7}$$

Here $\mathbf{u}$, p, ρ, and μ denote velocity, pressure, density, and viscosity, respectively. Furthermore, $\mathbf{f}$ and $\mathbf{e}_g$ represent surface tension and gravity forces. In addition we need boundary conditions and appropriate initial conditions. To solve the equations, finite difference, finite volume, or finite element discretizations can be used. In Paper 1 and Paper 2 we have used the finite element framework. In the next two sections we discretize the Navier–Stokes equations using the finite element technique.

3.1 Variational formulation

The finite element method is based on the variational formulation of the equations. For the weak formulation of the Navier–Stokes equations [23, 20] we introduce sub–spaces of the Lebesgue function space $L^2(\Omega)$. Let

$$L_0^2(\Omega) = \left\{ q \in L^2(\Omega) \;\middle|\; \int_\Omega q d\mathbf{x} = 0 \right\} \tag{3.8}$$

be the space of square integrable functions with zero mean over Ω equipped with the inner product and norm

$$(q, v) = \int_\Omega q v d\mathbf{x}, \quad ||q|| = (q, q)^{1/2}. \tag{3.9}$$

Consider also the subspace

$$H^1(\Omega) = \left\{ v \in L^2(\Omega) \;\middle|\; \frac{\partial v}{\partial x_k} \in L^2(\Omega), \;\; k = 1, 2 \right\} \tag{3.10}$$

and the corresponding norm

$$||v||_1 = (||v||^2 + ||\nabla v||^2)^{1/2}. \tag{3.11}$$

Denote the subspace of functions $v \in H^1(\Omega)$ which are zero on $\partial\Omega$ by

$$H_0^1(\Omega) = \left\{ v \in H^1(\Omega) \;\middle|\; v|_{\partial\Omega} = 0 \right\} \tag{3.12}$$

and let

$$\mathbf{H}^1(\Omega) = \left\{ v_j \in L^2(\Omega) \;\middle|\; \frac{\partial v_j}{\partial x_k} \in L^2(\Omega) \;\; j, k = 1, 2 \right\} \tag{3.13}$$

be the space of vector valued functions $\mathbf{v} = (v_1, v_2)$ such that each of the components belong to $H^1(\Omega)$ and the norm $||\mathbf{v}||_1 = \left(\sum_{i=1}^2 ||v_i||_1^2 \right)^{1/2}$.

Assuming homogeneous Dirichlet boundary conditions on solid walls for the velocity (referred to as the no slip condition) the weak formulation of the Navier–Stokes equations is as follows [20]: find $\mathbf{u}(\mathbf{x},t) \in H_0^1(\Omega)$ and $p(\mathbf{x},t) \in L_0^2(\Omega)$ such that $\forall t \in [0,T]$

$$((\rho\mathbf{u})_t, \mathbf{v}) + c(\mathbf{u},\mathbf{v},\mathbf{u}) - b(p,\mathbf{v}) + a(\mathbf{u},\mathbf{v}) = (\rho g \mathbf{e}_g, \mathbf{v}) + (\mathbf{f},\mathbf{v}), \tag{3.14}$$

$$b(q,\mathbf{u}) = 0 \tag{3.15}$$

for all test functions $(\mathbf{v},q) \in H_0^1(\Omega) \times L_0^2(\Omega)$. Here the forms are defined as

$$\begin{aligned} &a : H_0^1(\Omega) \times H_0^1(\Omega) \to \mathbb{R}, \\ &a(\mathbf{u},\mathbf{v}) = \int_\Omega \mu \left(\nabla\mathbf{u} : \nabla\mathbf{v} + (\nabla\mathbf{u})^T : \nabla\mathbf{v} \right) d\mathbf{x}, \end{aligned} \tag{3.16}$$

$$\begin{aligned} &b : L_0^2(\Omega) \times H_0^1(\Omega) \to \mathbb{R}, \\ &b(q,\mathbf{v}) = \int_\Omega q \nabla\cdot\mathbf{v} d\mathbf{x}, \end{aligned} \tag{3.17}$$

and

$$\begin{aligned} &c : H_0^1(\Omega) \times H_0^1(\Omega) \times H_0^1(\Omega) \to \mathbb{R}, \\ &c(\mathbf{u},\mathbf{v},\mathbf{w}) = \int_\Omega \rho(\mathbf{u}\cdot\nabla\mathbf{v})\cdot\mathbf{w} d\mathbf{x}. \end{aligned} \tag{3.18}$$

The double dot operation is defined as

$$\mathbf{A} : \mathbf{B} = \sum_i \sum_j a_{ij} b_{ji}. \tag{3.19}$$

In the next section we discuss the discretization for the equations (3.14) and (3.15).

3.2 The finite element discretization

Here we present the spatial discretization of the Navier–Stokes equations and in the next section we discuss the time discretization. Throughout this thesis, we use triangular meshes consisting of regular triangles with side h as shown in Fig. 3.2. In the finite element approximation we introduce the finite dimensional subspaces $\mathbf{V}_0^h \subset \mathbf{H}_0^1(\Omega)$ and $S_0^h \subset L_0^2(\Omega)$ of piecewise polynomials. The problem is to determine the discrete velocities $\mathbf{u}_h \in \mathbf{V}_0^h(\Omega)$ and pressure $p_h \in S_0^h(\Omega)$ such that

$$((\rho\mathbf{u}_h)_t, \mathbf{v}_h)_h + c_h(\mathbf{u}_h,\mathbf{v}_h,\mathbf{u}_h) - b_h(p_h,\mathbf{v}_h) + a_h(\mathbf{u}_h,\mathbf{v}_h) = (\rho g \mathbf{e}_g, \mathbf{v}_h)_h + (\mathbf{f},\mathbf{v}_h)_h, \tag{3.20}$$

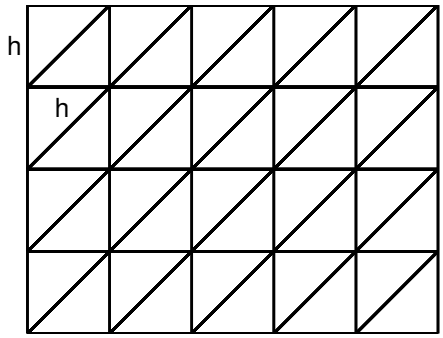


Figure 3.2: A rectangular domain subdivided into regular triangles with mesh size h.

$$b_h(q_h, \mathbf{u}_h) = 0 \tag{3.21}$$

for all test functions $(\mathbf{v}_h, q_h) \in \mathbf{V}_0^h(\Omega) \times S_0^h(\Omega)$. The forms $a_h(\cdot,\cdot)$, $b_h(\cdot,\cdot)$, $c_h(\cdot,\cdot,\cdot)$, the scalar product $(\cdot,\cdot)_h$, and consequently the norms are defined in the piecewise sense.

The finite dimensional spaces $\mathbf{V}_0^h$ and S_0^h being subspaces of $\mathbf{H}_0^1(\Omega)$ and $L_0^2(\Omega)$ is not sufficient to produce meaningful approximations. For some choice of spaces spatial oscillations in the pressure field can be observed that may blow up as the mesh is refined. This problem is also present in the solution of the linear equations of Stokes flow. Hence, the oscillations are not a result of the nonlinearity of the Navier–Stokes equations.

A necessary condition for the finite element spaces to satisfy is the so–called "inf–sup" condition

$$\inf_{0 \neq q_h \in S_0^h} \left\{ \sup_{0 \neq \mathbf{v}_h \in \mathbf{V}_0^h} \frac{b_h(q_h, \mathbf{v}_h)}{||q_h|| ||\mathbf{v}_h||_1} \right\} \geq D > 0, \tag{3.22}$$

where the constant D is independent of h. This condition is also referred to as the LBB condition after Ladyshenskaya [24], Babuska [25], and Brezzi [26]. The consequence of the LBB condition is that we cannot use all combinations of the spaces $\mathbf{V}_0^h$ and S_0^h. The LBB condition is for example satisfied by the Taylor–Hood pair; that is when $\mathbf{V}_0^h(\Omega)$ is the space of piecewise quadratic polynomials and the functions in $S_0^h(\Omega)$ are piecewise linear polynomials. For the piecewise quadratic velocity approximation there are six coefficients that need to be determined at each triangle. These coefficients are uniquely determined by the values of $\mathbf{u}_h$ at the triangle vertices and midpoints of the edges

of the triangles. From a computational cost perspective and, possibly even more important, for simplicity in the implementation it is advantageous to approximate both the velocity and pressure using continuous piecewise linear elements. However, this element pair satisfies the LBB condition only when different meshes for velocity and pressure is used. One way to obtain such meshes is to construct the mesh for the velocity by refining each triangle of the pressure mesh into four triangles.

It has been shown that by a modification of the incompressibility constraint it is possible to circumvent the LBB condition [27, 28]. The simplest modification is to change the incompressibility condition (3.21) to

$$b_h(q_h, \mathbf{u}_h) - \varepsilon_p h^2 (\nabla p, \nabla q_h)_h = 0. \tag{3.23}$$

By doing that we can use piecewise linear elements both for the velocity and the pressure without introducing different meshes. However, there is a new parameter $\varepsilon_p \neq 0$ in the problem that must be determined.

Note that also for finite difference discretizations of the Navier–Stokes equations there are constraints. For example, the use of finite difference scheme where all the derivatives are approximated on the same grid using central difference quotients results in pressure oscillations. This is avoided by use of staggered meshes for the pressure and the velocity [29].

When bases are chosen for the spaces $\mathbf{V}_0^h$ and S_0^h, the finite element approximations for the velocity and the pressure can be expressed in these bases as

$$\begin{aligned} \mathbf{u}_h(t, \mathbf{x}) &= \sum_{k=1}^{K} \beta^k(t) \mathbf{v}_h^k(\mathbf{x}), \\ p_h(t, \mathbf{x}) &= \sum_{j=1}^{J} \alpha^j(t) q_h^j(\mathbf{x}). \end{aligned} \tag{3.24}$$

Here $\{\mathbf{v}_h^k\}$, $k = 1, \cdots, K$ and $\{q_h^j\}$, $j = 1, \cdots, J$ denote bases for $\mathbf{V}_0^h$ and S_0^h respectively. For example Lagrange bases can be used. The coefficients β^k and α^j need to be determined.

Substituting the expansions in equation (3.24) into equations (3.20) and (3.21) and choosing $\mathbf{v}_h = \mathbf{v}_h^k$ for $k = 1, \cdots, K$ and $p_h = p_h^j$ for $j = 1, \cdots, J$, equations (3.20) and (3.21) are equivalent to a system of nonlinear ordinary differential equations with constraints [23]. In the next section we describe a time discretization algorithm.

3.2.1 A fractional step projection method

To advance the Navier–Stokes equations in time, a temporal discretization is needed. To use a fully implicit method directly on the full system of equa-

tions results in solving a coupled nonlinear system at every time step. Such methods have good stability properties but are computationally expensive and cumbersome to implement. Fully explicit methods are not often used because of their stability restrictions on the time step. Fractional step methods have become a popular alternative to solving the full system. Examples of such methods include projection methods, for an overview see [30]. In these methods a sequence of decoupled equations for the velocity and the pressure has to be solved at each time step.

The following pressure correction scheme has been used in Paper 1 and Paper 2: First, an intermediate velocity denoted by $\mathbf{u}_*^{n+1}$ is determined satisfying

$$\left(\frac{\rho^{n+1}\mathbf{u}_*^{n+1}-\rho^n\mathbf{u}^n}{dt},\mathbf{v}_h\right)+c_h(\mathbf{u}^n,\mathbf{v}_h,\mathbf{u}_*^{n+1})-b_h(p^n,\mathbf{v}_h)+a_h(\mathbf{u}_*^{n+1},\mathbf{v}_h)=$$
$$(\rho^{n+1}g\mathbf{e}_g,\mathbf{v}_h)+(\mathbf{f}^{n+1},\mathbf{v}_h). \tag{3.25}$$

This intermediate velocity field is not necessarily divergence free. The pressure is treated explicitly in this first step but is corrected in the next step referred to as the projection step. In the projection step the incompressibility condition is enforced by solving the Poisson equation

$$-\frac{1}{dt}b_h(q_h,\mathbf{u}_*^{n+1})=\left(\frac{\nabla(p^{n+1}-p^n)}{\rho^{n+1}},\nabla q\right) \tag{3.26}$$

for the pressure. Note that in equation (3.26) we have assumed that $\mathbf{u}^{n+1}$ satisfies the incompressibility condition. The divergence free velocity $\mathbf{u}^{n+1}$ is computed from

$$\left(\frac{\mathbf{u}^{n+1}-\mathbf{u}_*^{n+1}}{dt},\mathbf{v}\right)=-\left(\frac{\mathbf{v}\nabla(p^{n+1}-p^n)}{\rho^{n+1}},\mathbf{v}\right). \tag{3.27}$$

In the first step, p^n is a first order extrapolation for $p(t^{n+1})$ and a first order accurate method (the backward Euler formula) has been used to approximate the time derivative. Therefore the velocity is first order accurate in H^1–norm and the pressure is first order accurate in L^2–norm. By taking a second order backward difference formula (BDF2) for the approximation of the time derivative

$$\partial_t\mathbf{u}(t^{k+1})\approx 3/2\mathbf{u}^{k+1}-2\mathbf{u}^k+1/2\mathbf{u}^{k-1} \tag{3.28}$$

one obtains a second order accurate velocity approximation (in H^1–norm) and a first order accurate pressure approximation (in L^2–norm) [30].

Since in Paper 1 and Paper 2 continuous piecewise linear elements were used both for the pressure and the velocity we modify the incompressibility condition according to equation (3.23). Consequently, equation (3.26)

and (3.27) are modified. We do this in order to circumvent the LBB condition, see the previous section.

In the next chapter we discuss how the singular surface tension force is approximated.

Chapter 4

The surface tension force

In this work we use the continuum surface force model [1, 21, 22]. In this model the surface tension effect is treated as a source term added to the Navier–Stokes equations. The force is expressed in terms of a Dirac delta function δ_Γ with support on the interface Γ

$$\mathbf{f} = \sigma\kappa\mathbf{n}\delta_\Gamma. \tag{4.1}$$

In this chapter we will discuss how the surface tension force can be computed.

4.1 Regularized representation of the surface tension force

A popular way of approximating the surface tension force $\mathbf{f}$, defined in equation (4.1) has been to regularize it. To approximate the surface tension force Brackbill et al. suggested to compute

$$\sigma\kappa\nabla\phi = \sigma\kappa\frac{\nabla\phi}{|\nabla\phi|}|\nabla\phi|, \tag{4.2}$$

where ϕ is a regularized indicator function. The model is often used together with volume of fluid methods, and was also used together with the conservative level set method, with ϕ being the Fermi–Dirac function [18, 19]. This model can be seen as approximating the Dirac delta function with support on the interface Γ by $|\nabla\phi|$.

4.1.1 Regularized Dirac delta functions with support on interfaces

Dirac delta functions with support on interfaces can be approximated also in other ways. An early example is the work by Peskin [1]. Peskin introduced

the one–dimensional cosine approximation

$$\delta_{2h}^{\cos}(x) = \begin{cases} \frac{1}{4h}(1+\cos(\pi x/(2h))) & \text{for } |x| < 2h, \\ 0 & \text{for } |x| \geq 2h, \end{cases} \tag{4.3}$$

and used a product rule to extend this function to higher dimensions.

Let $\Gamma \subset \mathbb{R}^d$ be a $d-1$ dimensional closed, continuous, and bounded surface and let S be surface coordinates on Γ. The product formula yields

$$\delta_\varepsilon(\Gamma, g, \mathbf{x}) = \int_\Gamma \prod_{k=1}^{d} \delta_{\varepsilon_k}(x^{(k)} - X^{(k)}(S)) g(S) dS, \tag{4.4}$$

where δ_{ε_k} is a one–dimensional regularized delta function, $\mathbf{x} = (x^{(1)}, \cdots x^{(d)})$, $\mathbf{X}(S) = (X^{(1)}(S), \cdots, X^{(d)}(S))$ is a parametrization of Γ, and $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_d)$ is the regularization parameter. Tornberg and Engquist [31] showed that this extension technique is consistent when used to approximate integrals of the form

$$\int_\Gamma g(S) f(\mathbf{X}(S)) dS = \int_{\mathbb{R}^d} \delta(\Gamma, g, \mathbf{x}) f(\mathbf{x}) d\mathbf{x}. \tag{4.5}$$

Here $\delta(\Gamma, g, \mathbf{x})$ is a delta function of variable strength with support on Γ. The product formula is easy to use when Γ is explicitly defined as in the immersed boundary method, the front tracking method, or the segment projection method.

In level set methods the signed distance function $d(\Gamma, \mathbf{x})$ has been used to extend a one–dimensional delta function approximation to higher dimensions as

$$\delta_\varepsilon(\Gamma, g, \mathbf{x}) = \tilde{g}(\mathbf{x}) \delta_\varepsilon(d(\Gamma, \mathbf{x})). \tag{4.6}$$

Here $\tilde{g}$ is an extension of g to $\mathbb{R}^d$, such that $\tilde{g}(X(S)) = g(S)$. In the case of surface tension $g = \kappa \mathbf{n}$. The great appeal of this extension technique is its simplicity. However, it has been shown that with the common choice of regularization parameter, i.e. $\varepsilon = mh$, this approach may lead to $\mathcal{O}(1)$ errors [31]. Even when consistent one–dimensional delta function approximations are used. To overcome this lack of consistency several delta function approximations have been derived to be used with level set methods [32–34]. These methods are not based on the extension technique defined in equation (4.6).

In Paper 3 we show on regular grids that there is a class of one–dimensional delta function approximations that can be extended to higher dimension by a distance function according to equation (4.6) and give accurate results. This class of delta function approximations have compact support in Fourier

space or in practice decay rapidly in Fourier space. An example of such one–dimensional delta function approximation is the Gaussian approximation

$$\delta_\varepsilon^G(x) = \frac{1}{\varepsilon}\varphi^G(x/\varepsilon), \quad \varphi^G(\xi) = \sqrt{\frac{\pi}{9}}e^{-\pi^2\xi^2/9}. \tag{4.7}$$

Another example is the derivative of the Fermi–Dirac function

$$\delta_\varepsilon^{FD}(x) = \partial_x \frac{1}{1+e^{-x/\varepsilon}} = \frac{1}{\varepsilon}\frac{e^{-x/\varepsilon}}{(1+e^{-x/\varepsilon})^2}. \tag{4.8}$$

used in the conservative level set method [18, 19]. These approximations are shown to be second order accurate.

In the case when the interface is close to a boundary of the computational domain care must be taken when regularization is used. If some part of the regularization zone ends up outside the domain the accuracy of the delta function approximation will usually be lost. A remedy would be to use a skewed delta function approximation with the whole mass always being inside the domain [35]. However, this technique is complicated and the distance to the boundary must be known. The regularized delta function by Smereka with support within one mesh cell could also be an alternative [33]. However, the problem remains when the interface is in contact with the boundary.

In the finite element framework, based on the weak form, this problem can be solved by evaluating the surface tension force through a line integral. This technique is described in the next section.

4.2 Sharp representation of the surface tension force

Sharp treatment of the surface tension has been suggested both in finite element [36] and finite difference [37] settings. In [37], surface tension effects are treated as boundary conditions, see equation (3.3). In the finite element framework the surface tension force is evaluated through a line integral

$$(\mathbf{f}, \mathbf{v}) = \int_\Omega \sigma\kappa\mathbf{n}\cdot\mathbf{v}\delta_\Gamma d\mathbf{x} = \int_\Gamma \sigma\kappa\mathbf{n}\cdot\mathbf{v}d\Gamma, \tag{4.9}$$

where $\mathbf{v}$ is the test function. This approach was used in Paper 1 and Paper 2 when simulations of moving contact points were performed. An advantage of evaluating the line integral is that there is no need to discretize delta functions. A drawback is that particular effort is needed to find the triangles that are intersected by the interface Γ. Using a level set method and linear elements the values of the function ϕ at the three nodes of each triangle will determine if the element is intersected by the level set defining Γ. For those

elements a quadrature scheme that is exact for linear elements can be used to evaluate the integral. Using quadratic elements to approximate the level set function ϕ one uses the three nodes and the midpoints of each triangle to decide whether the element is intersected by the interface or not. Those elements that are intersected are split into four sub–triangles where in each sub–triangle a piecewise linear approximation of the level set is defined. The integration is then performed as above.

From an implementation viewpoint this approach is more complicated than regularizing the delta function. However, as we indicated in the previous section a sharp treatment is preferable in view of accuracy when the interface is close to the boundary of the computational domain.

In order to have an accurate representation of the surface tension force we also need to have an accurate approximation of the curvature. In the next section we discuss this issue.

4.3 Evaluation of the curvature

The curvature in level set methods and volume of fluid methods is often computed from equations (2.6) and (2.7). This is how the curvature is computed in Paper 1. The curvature depends on second derivatives of the function representing the interface. The presence of high frequency errors, that are magnified by differentiation, may lead to an inaccurate approximation of the surface tension force. A common approach to avoid such errors is to filter, or damp the high frequencies. A filtering can be applied to the level set function ϕ, before the curvature is computed or directly to the curvature κ [19]. In Ref. 19 a new filtered curvature $\tilde{\kappa}$ was computed according to

$$\tilde{\kappa} - \varepsilon_\kappa \Delta \tilde{\kappa} = \kappa, \tag{4.10}$$

with ε_κ proportional to the mesh size h. The same idea of filtering has also been used in the volume of fluid method [38]; in this context a convolution technique was used.

Another approach is to reduce the order of differentiation associated with the curvature term by using the Laplace–Beltrami operator and the finite element technique. This idea was introduced by Dziuk [39] in numerical simulations and has been used in simulations of immiscible multiphase flow problems [40–43] for computing the surface tension. Let the tangential gradient of a function $f : U \to \mathbb{R}$, with U an open neighborhood of Γ be defined by

$$\underline{\nabla} f = P \nabla f, \quad P = I - \mathbf{n}\mathbf{n}^T. \tag{4.11}$$

Here P is the orthogonal projection. The Laplace–Beltrami operator of f is given by

$$\underline{\Delta} f(x) = \underline{\nabla} \cdot (\underline{\nabla} f(x)). \tag{4.12}$$

The method is based on the following theorem:

Theorem 4.3.1.

$$\underline{\Delta} id_\Gamma = \kappa \mathbf{n}, \tag{4.13}$$

where κ is the mean curvature, $\mathbf{n}$ is a unit normal vector and $id_\Gamma : \Gamma \to \Gamma$ is the identity mapping on Γ.

For a proof see [44, 45]. Using Theorem 4.3.1 and partial integration we have in two space dimensions

$$\int_\Omega \sigma\kappa\mathbf{n} \cdot \mathbf{v}\delta_\Gamma d\Omega = -\int_\Gamma \sigma \underline{\nabla} id_\Gamma \cdot \underline{\nabla}\mathbf{v} d\Gamma + \int_\gamma \sigma(\mathbf{n}_\gamma \cdot \underline{\nabla} id_\Gamma) \cdot \mathbf{v} d\gamma. \tag{4.14}$$

Here $\mathbf{n}_\gamma \cdot \underline{\nabla} id_\Gamma = \mathbf{n}_\gamma$ is the normal at the contact line γ. The last term in equation (4.14) is an integral along the contact line. In two space dimensions it is replaced by a sum over contact points. This term will only appear if Γ intersects the boundary and $\mathbf{v}$ is non–vanishing on γ. For more details, see [40]. Note that

$$\int_\Gamma \underline{\nabla} id_\Gamma \cdot \underline{\nabla}\mathbf{v} d\Gamma = \sum_{i=1}^{d} \int_\Gamma \underline{\nabla}(id_\Gamma)_i \cdot \underline{\nabla}\mathbf{v}_i d\Gamma, \tag{4.15}$$

where we have $d = 2$ since we consider problems in two dimensions. Further,

$$\underline{\nabla}(id_\Gamma)_i(\mathbf{x}) = \underline{\nabla}\mathbf{x}_i = Pe_i, \tag{4.16}$$

where e_i is the ith basis vector in $\mathbb{R}^2$ and P is the orthogonal projection defined in equation (4.11). The line integral is evaluated as described in the previous section. The advantage of using this approach is that there is no need to compute the curvature. This method was employed in Paper 2.

Next we discuss the treatment of the discontinuous density and viscosity in two phase flow problems.

Chapter 5

Discontinuous physical quantities

In multiphase flow problems the density and the viscosity may be discontinuous functions. We can write the density and viscosity as

$$\begin{aligned} \rho(\mathbf{x}) &= \rho_1 + (\rho_2 - \rho_1)I(\mathbf{x}), \\ \mu(\mathbf{x}) &= \mu_1 + (\mu_2 - \mu_1)I(\mathbf{x}), \end{aligned} \tag{5.1}$$

where $I(\mathbf{x})$ is the indicator function which is 1 in Ω_2 and 0 in Ω_1, see Fig. 3.1. When solving the Navier–Stokes equations these functions as well as the derivative of $\mu(\mathbf{x})$ have to be evaluated. In the finite element framework, Green's formula can be used to move the derivative of $\mu(\mathbf{x})$ to the test function.

In level set methods, where ϕ is the signed distance function $d(\Gamma, \mathbf{x})$, the indicator function is written as

$$I(\mathbf{x}) = H(d(\Gamma, \mathbf{x})), \tag{5.2}$$

where $H(t)$ is the Heaviside function

$$H(t) = \begin{cases} 0 & \text{for } t < 0, \\ 1/2 & \text{for } t = 0, \\ 1 & \text{for } t > 0. \end{cases} \tag{5.3}$$

Thus, in the variational formulation integrals of the form

$$\int_\Omega H(d(\Gamma, \mathbf{x}))G(\mathbf{x})d\mathbf{x}, \tag{5.4}$$

need to be evaluated. Here $G(\mathbf{x})$ is a smooth function.

A common approach to handle the discontinuities in the density and the viscosity is to regularize the indicator function. This approach is very easy to implement. When a regularized indicator function is used the density and the

viscosity varies continuously from ρ_2 and μ_2 in Ω_2 to ρ_1 and μ_1 in Ω_1. The regularization can be done by regularizing the Heaviside function. A common class of regularized Heaviside functions are written as

$$H_\varepsilon(t) = \begin{cases} 0 & \text{for } t < -\varepsilon, \\ \nu(t/\varepsilon) & \text{for } |t| \leq \varepsilon, \\ 1 & \text{for } t > \varepsilon, \end{cases} \tag{5.5}$$

where $\nu(\xi)$ is a smooth transition function with $\nu(-1) = 0$ and $\nu(1) = 1$. The error introduced by this particular class of approximations was analyzed in [35].

In level set methods

$$\nu(\xi) = \frac{1}{2}(1 + \xi + \frac{1}{\pi}\sin(\pi\xi)) \tag{5.6}$$

has been a popular choice. This approximation gives an analytical error that arises when $H(t)$ is replaced with $H_\varepsilon(t)$ which is proportional to ε^2.

In the conservative level set method the level set function ϕ is a regularized indicator function. Thus, the density and the viscosity are represented by replacing the indicator function $I(\mathbf{x})$ in equation (5.1) by $\phi(\mathbf{x})$. It can be shown that the analytical error in this case is proportional to ε^3.

For the treatment of discontinuity in the density and the viscosity another approach is to split the integration over Ω into integration over the subdomains Ω_1 and Ω_2. A benefit of this approach is that we only integrate a smooth function and do not need to discretize the Heaviside function. This method is possibly to prefer when the interface Γ is close to a boundary. When regularizing the indicator function care is needed in order to keep the transition zone within the computational domain. It is difficult to implement the integration over the subdomains Ω_1 and Ω_2. In the discrete form the interface Γ separating the two immiscible fluids is approximated and thus the integration over the domains Ω_1 and Ω_2 is replaced with integration over $\Omega_{h,1}$ and $\Omega_{h,2}$. Triangles that are in $\Omega_{h,2}$ must be found. Some of the triangles are intersected by the interface Γ. Thus, integration should be done only over parts of these triangles.

In the next chapter we consider immiscible fluids in contact with a solid surface and discuss how we treat moving contact points.

Chapter 6

Moving contact lines

Consider two immiscible fluids, for example oil and water in contact with a solid surface. In two dimensions, the point where the interface separating the two fluids intersect the solid surface is called the contact point. In three dimensions there are contact lines. The phenomenon when one of the fluids spreads on the solid surface and displaces the other is called wetting. The situation when the fluid has spread completely is referred to as total wetting. We consider here partial wetting, which occurs when the liquid spreads but at equilibrium makes a nonzero contact angle to the wall, referred to as the static contact angle, see Fig. 6.1. The static contact angle θ_s can be measured in experiments and is given by the surface tension coefficients of the solid/medium σ_{SM}, solid/liquid σ_{SL}, and liquid/medium σ_{LM} interfaces via Young's equation:

$$\cos(\theta_s) = \frac{\sigma_{SM} - \sigma_{SL}}{\sigma_{LM}}. \tag{6.1}$$

As the liquid drop wets the solid surface the contact angle changes and approaches the static contact angle. Besides being an interesting physical phenomenon, the understanding of the contact line movement is important in industrial applications. However, the physics behind the contact line movement is not entirely known.

A difficulty has been that the Navier–Stokes equations coupled with the no–slip boundary condition for the velocity at solid walls, and interface movement by advection only leads to a singular stress tensor [46]. Many models have been developed to avoid this singularity. A popular model has been to allow the contact line to move by introducing a slip boundary condition. We discuss this in the next section. In the diffuse interface model presented by Jacqmin [5], the contact line moves by diffusion rater than advection. Then, the standard no–slip boundary conditions at solid walls can be used. The idea to move the contact line by diffusion is employed in Paper 1 and Paper 2.

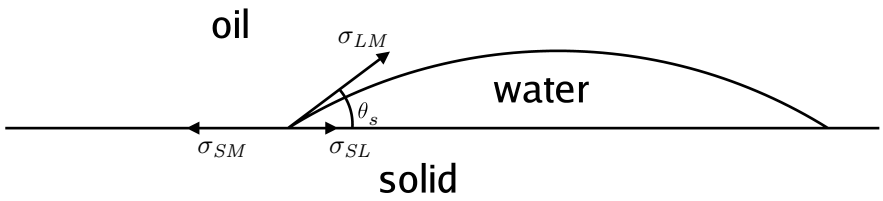


Figure 6.1: A water droplet surrounded by oil is lying on a solid wall. The drop is at equilibrium and the interface separating the two immiscible fluids has formed an angle to the solid surface referred to as the static contact angle θ_s. This angle can be determined from the interfacial tensions by Young's equation: $\cos(\theta_s) = \frac{\sigma_{SM}-\sigma_{SL}}{\sigma_{LM}}$.

6.1 Slip models

It has been shown that the stress tensor singularity can be avoided when the fluid is allowed to slip [47]. Therefore, simulations where moving contact lines or points occur often employ the Navier–Stokes equations together with some slip model near the contact line. Different slip models exist; a common choice is the Navier boundary condition:

$$\mathbf{u}_{\text{slip}} = -l_s \mathbf{n} \cdot (\nabla(\mathbf{u}) + (\nabla(\mathbf{u}))^T). \tag{6.2}$$

Here the amount of slip is proportional to the shear stress. The parameter l_s is the slip length and $\mathbf{n}$ is the normal vector of the solid surface pointing out of the fluid. Another model employs complete slip at the contact line but away from the contact line the no–slip boundary condition is used. Zhou and Sheng used a slip that decays exponentially to no–slip far from the contact line [48].

Experiments by Hoffman [49], Cox theory [50], and molecular dynamics simulations [51,52] all suggest that the deviation of the contact angle from its static value is important for the contact line motion. These results have lead to construction of new boundary conditions incorporating the dependence of the slip on the contact angle [51,52]. Others have used the Navier slip boundary condition but explicitly reconstruct the interface close to the contact line or points in order to obtain a prescribed contact angle at the wall [53,54].

In all the models above, parameters such as the slip length and the contact angle at the wall must be determined. These parameters are important for

the movement of the contact line and influence the flow [50]. In Ref. 52, these parameters were determined based on molecular dynamics simulations. In other numerical simulations the parameters are tuned to make the simulations comparable to experiments.

In the next section, we will consider a diffuse interface model where the force singularity is not present and movement of contact lines and points is possible even with the no–slip boundary condition for the velocity at solid walls. We also present the model we used in Paper 1 and Paper 2. In these methods a contact angle is prescribed without explicit reconstruction of the interface.

6.2 Moving contact lines with diffusion

Consider two incompressible immiscible fluids. In diffuse interface methods the two fluids are separated by a diffuse interface with finite thickness. Density and viscosity change continuously over the interface. A diffuse interface model applicable to problems with moving contact lines was presented by Jacqmin [5], who used the standard no–slip boundary condition for the velocity at solid walls. A coupled Cahn–Hilliard/Navier–Stokes formulation was used; a function describing the concentration of each fluid defined the diffuse interface. Two parabolic equations were solved, one for the concentration and another for the chemical potential. Near the wall the contact line moves by a diffusive process. In this region the curvature is high and consequently the surface tension force is strong. Villanueva and Amberg [55] and Khatavkar et al. [56] used this Cahn–Hilliard/Navier–Stokes model and used a boundary condition for the concentration which lead to the contact angle in the contact line region to be equal to the static contact angle. Thus, at the wall the fluid was at equilibrium. In Ref. 5, an expression was given that allows also for a non–equilibrium contact angle.

In the diffuse interface method above, a very fine mesh near the interface is required. The conservative level set method requires less computational effort away from contact points since a relatively coarse grid is sufficient to capture the dynamics. Next we give a brief description of how to extend the conservative level set method to be able to simulate moving contact points.

6.2.1 The conservative level set method with moving contact points

The model in Paper 1 and Paper 2 mimics the Cahn–Hilliard/Navier–Stokes model. Also in our model, the interface becomes highly curved in the contact point regions. In these regions, the surface tension forces are strong and balanced by viscous forces. By including diffusion in the reinitialization

step contact points can move without advection, and the no–slip boundary condition for velocities at solid walls can be retained.

In each time step, first a regularized normal vector $\tilde{\mathbf{n}}$ is computed by

$$\begin{aligned} &\tilde{\mathbf{n}} - \nabla \cdot (\gamma^2 \nabla \tilde{\mathbf{n}}) = \frac{\nabla \phi}{|\nabla \phi|}, \quad \mathbf{n} = \frac{\tilde{\mathbf{n}}}{|\tilde{\mathbf{n}}|}, \\ &\tilde{\mathbf{n}} = \mathbf{n}_{\alpha_s}, \end{aligned} \tag{6.3}$$

where $\mathbf{n}_{\alpha_s}$ is the normal vector corresponding to the static contact angle. Then the normal vector $\mathbf{n}$ is used in the reinitialization equation (2.14). The tangential diffusion in the reinitialization process is essential for controlling the contact angle at the solid boundary without a reconstruction step. The result of this procedure is a boundary layer in ϕ, of thickness proportional to the regularization parameter γ. The shape and size of the contact point region, denoted by A in Fig. 6.2, will depend on the regularization parameter, the diffusion in the reinitialization, and the deviation of the contact angle from the static angle. However, we demonstrate numerically in Paper 1 that when the thickness of the boundary layer and the diffusion in the reinitialization decrease, the movement of the interface becomes more and more independent of these parameters.

In Paper 2 we implemented this model and used the Laplace–Beltrami characterization of the curvature together with partial integration to avoid direct computation of the curvature, see Section 4.3. It is difficult to accurately compute the high curvature of the interface that may appear in contact point regions. In our model all test functions $\mathbf{v}$ are zero at solid walls, due to the no–slip boundary conditions for velocities. Thus the second term in equation (4.14) vanishes. In Ref. 41 the Laplace–Beltrami characterization of the curvature was also used, but together with slip boundary conditions for the velocity. In that setting the boundary term in equation (4.14) does not vanish.

To summarize, our model for two–phase flow with moving contact points consists of the incompressible Navier–Stokes equations coupled with evolution of the interface by equations (2.11) and (2.14). All equations are in conservative form. Therefore, using conservative numerical methods to compute ϕ_h and no flux boundary conditions, the integral in equation (2.16) is conserved.

In Paper 1 and Paper 2 we considered capillary dominated flows and prescribed the static contact angle. Future work includes simulation of problems where it may be better to prescribe other contact angles than the static, for example in problems where inertial effects dominate. Another possible project is the use of adaptive mesh refinement in the contact line region.

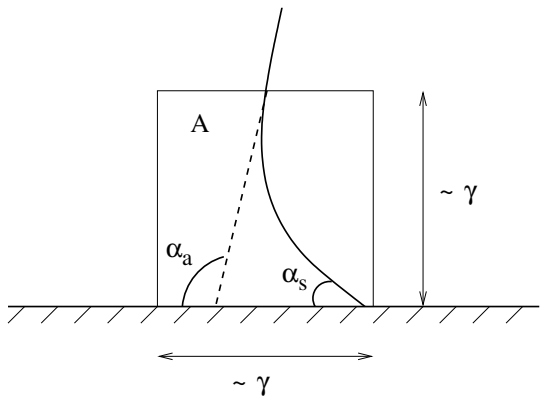


Figure 6.2: A boundary layer in ϕ, of thickness proportional to the regularization parameter γ. The static contact angle is prescribed at the solid surface. The apparent dynamic contact angle is the angle formed between the boundary and the interface away from the contact point region.

Chapter 7

Discussion and future outlook

A common computational approach to treat discontinuities in multiphase flow problems has been to smear them out across the interface. See Chapter 5 for the treatment of the density and the viscosity. Also, the delta function used to express the singular surface tension force is often regularized. This makes the pressure a continuous function. However, in the mathematical model the pressure is a discontinuous function.

In simulations where the pressure has been treated as a continuous function, large oscillations of the velocity near the interface have sometimes been observed. These oscillations, referred to as spurious currents, have resulted in unphysical movements of the interface. As a result of the spurious currents some methods have failed to converge with grid refinement.

One reason to why spurious currents occur is the use of unsatisfactory representations of the surface tension force acting at the interface separating the two immiscible fluids. Other sources that affect the size of the spurious velocities are the curvature and the pressure approximations [57–59].

Within the finite difference technique, methods have been developed for solving the pressure jump directly by treating the surface tension effects as boundary conditions, see equation (3.3). Examples are the ghost fluid method [37] and the pressure boundary method [57] which treat the jump discontinuities at the interface in a sharp way.

In the finite element framework, a strategy has been to enrich the pressure space by discontinuous basis functions to better approximate the discontinuous pressure [58]. In Ref. 58 a line integral formulation of the surface tension force was used and the discontinuous basis functions allowed for an accurate pressure approximation. However, some stability problems has been reported. In Ref. 60 Green's formula was used to rewrite the surface tension force in weak form so that the pressure jump across the interface becomes correct in the case of a static drop.

In this work we have treated the surface tension effect using the continuum

surface force model [1, 21, 22]. A source term has been added to the Navier–Stokes equations. This was done in order to solve one equation in the whole region instead of two equations, one in Ω_1 and another in Ω_2, coupled by the conditions (3.2) and (3.3), see Fig. 3.1.

I have considered a simulation of a static drop in equilibrium, where the velocity should be zero and the pressure jump is given by the Young–Laplace equation

$$[p]_\Gamma = \sigma\kappa. \tag{7.1}$$

I have compared use of the regularized surface tension force equation (4.2) and the line integral formulation equation (4.9). Preliminary results show that the problem with spurious oscillations is much less pronounced when the regularized force is used. However, when the interface is in contact with the boundary the regularization zone usually ends up outside the computational domain and the accuracy of the surface tension force is lost. Therefore a sharp representation of the surface tension force that do not give rise to spurious currents is desired.

One reason that the regularized force gives better results could be that the pressure gradient and the surface tension force are balanced. A subject for future work could be to better understand how to approximated the surface tension and the pressure in order to reduce spurious currents. How do the discretizations of the level set function, pressure, and surface tension affect the unphysical oscillations in the velocity?

References

[1] C. S. Peskin, Numerical analysis of blood flow in the heart, J. Comput. Phys. 25 (1977) 220–252.

[2] S. O. Unverdi, G. Tryggvason, A front–tracking method for viscous, incompressible, multi–fluid flows, J. Comput. Phys. 100 (1992) 25–37.

[3] S. Osher, J. A. Sethian., Fronts propagating with curvature dependent speed: Algorithms based on hamilton–jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.

[4] C. W. Hirt, B. D. Nichols, Volume of fluid (vof) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.

[5] D. Jacqmin, Contact-line dynamics of a diffuse fluid interface, J. Fluid Mech. 402 (2000) 57–88.

[6] D. Enright, R. Fedkiw, J. Ferziger, I. Mitchell, A hybrid particle level set method for improved interface capturing, J. Comput. Phys. 183 (2002) 83–116.

[7] A.-K. Tornberg, Interface Tracking Methods with Application to Multiphase Flows, Doctoral Dissertation, Royal Institute of Technology, ISBN 91–7170–558–9, 2000.

[8] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, A front–tracking method for the computations of multiphase flow, J. Comput. Phys. 169 (2001) 708–759.

[9] W. J. Rider, D. B. Kothe, Reconstructing volume tracking, J. Comput. Phys. 141 (1998) 112–152.

[10] I. Ginzburg, G. Wittum, Two–phase flows on interface refined grids modeled with vof, staggered finite volumes, and spline interpolants, J. Comput. Phys. 166 (2001) 302–335.

[11] J. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, 1999.

[12] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer-Verlag, 2003.

[13] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two–phase flow, J. Comput. Phys. 114 (1994) 146–159.

[14] J. A. Sethian., Theory, algorithms, and applications of level set methods for propagating interfaces, Acta Numer. 5 (1996) 309–395.

[15] W. J. Rider, D. B. Kothe, Stretching and tearing interface tracking methods, AIAA papers (1995) 95–1717.

[16] M. Sussman, E. Fatemi, An efficient, interface–preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow., SIAM J. Sci. Comput. 20 (1999) 1165–1191.

[17] M. Sussman, E. G. Puckett, A coupled level set and volume–of–fluid method for computing 3d and axisymmetric incompressible two–phase flows, J. Comput. Phys. 162 (2000) 301–337.

[18] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, J. Comput. Phys. 210 (2005) 225–246.

[19] E. Olsson, G. Kreiss, S. Zahedi, A conservative level set method for two phase flow ii, J. Comput. Phys. 225 (2007) 785–807.

[20] S. Grob, V. Reichelt, A. Reusken, A finite element based level set method for two–phase incompressible flows, Comput. Vis. Sci. 9 (2006) 239–257.

[21] J. U. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys 100 (1992) 335–353.

[22] Y. C. Chang, T. Y. Hou, B. Merriman, S. Osher, A level set formulation of eulerian interface capturing methods for incompressible fluid flows, J. Comput. Phys. 124 (1996) 449–464.

[23] M. D. Gunzburger, Finite element methods for viscous incompressible flows, Academic press, 1989.

[24] O. A. Ladyshenskaya, The Mathematical Theory of Viscous Incompressible Flow, 2nd ed. Gordon and Breach, New York, 1969.

[25] l. Babuska, The finite element method with lagrangian multipliers, Numer. Math 20 (1973) 179.

[26] F. Brezzi, On the existence uniqueness and approximation of saddle–point problems arising from lagrangian multipliers, Revue Francaise d'Automatique, d'Informatique et de Recherche Operationelle R.2 (1974) 129.

[27] F. Brezzi, J. Pitkaranta, On the stabilization of finite element approximations of the stokes problem., Efficient Solutions of Elliptic Systems Vieweg, Braunschweig (1984) 11–19.

[28] T. Hughes, L. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. circumventing the babuska–brezzi condition: A stable petrov–galerkin formulation of the stokes problem accommodating equal–order interpolations, Comput. Meths. Appl. Mech. Engrg. 59 (1986) 85–99.

[29] W. H. Press, B. P. Plannery, S. A. Teukolsky, W. T. Vetterling, Numerical Recipes–The Art of Scientific Computing, Cambridge University Press, 1986.

[30] J. L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Engrg. 195 (2006) 6011–6045.

[31] A.-K. Tornberg, B. Engquist, Numerical approximations of singular source terms in differential equations, J. Comput. Phys. 200 (2004) 462–488.

[32] B. Engquist, A.-K. Tornberg, R. Tsai, Discretization of dirac delta functions in level set methods, J. Comput. Phys. 207 (2005) 28–51.

[33] P. Smereka, The numerical approximation of a delta function with application to level set methods, J. Comput. Phys. 211 (2006) 77–90.

[34] J. D. Towers, Two methods for discretizing a delta function supported on a level set, J. Comput. Phys. 220 (2007) 915–931.

[35] A.-K. Tornberg, Multi–dimensional quadrature of singular and discontinuous functions, BIT 42 (2002) 644–669.

[36] A.-K. Tornberg, B. Engquist, A finite element based level set method for multiphase flow applications, Comput. Vis. Sci. 3 (2000) 93–101.

[37] M. Kang, R. Fedkiw, X. Liu, A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput. 15 (2000) 323–360.

[38] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, M. W. Williams, A balanced–force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework, J. Comput. Phys. 213 (2006) 141–173.

[39] G. Dziuk, An algorithm for evolutionary surfaces, Numerische Mathematik 58 (1991) 603–611.

[40] E. Bänsch, Numerical methods for the instationary navier–stokes equations with a free capillary surface, Habiliation thesis, University of Freiburg, Germany.

[41] S. Ganesan, L. Tobiska, Finite element simulation of a droplet impinging a horizontal surface, Proceedings of ALGORITMY (2005) 1–11.

[42] S. Hysing, A new implicit surface tension implementation for interfacial flows, Int. J. Numer. Meth. Fluids 51 (2006) 659–672.

[43] S. Gross, A. Reusken, Finite element discretization error analysis of a surface tension force in two–phase incompressible flows, SIAM J. Numer. Anal. 45 (2007) 1679–1700.

[44] D. Gilbarg, N. S. Trudinger, Elliptic partial differential equations of second order, Springer–Verlag, Berlin, 2001.

[45] S. Gallot, D. Hulin, J. Lafontaine, Riemannian Geometry, Springer, Berlin, 2004.

[46] E. B. Dussan V., S. H. Davis, On the motion of a fluid–fluid interface along a solid surface, J. Fluid Mech. 65 (1974) 71–95.

[47] L. M. Hocking, A moving fluid interface. part 2. the removal of the force singularity by a slip flow, J. Fluid Mech. 79 (1977) 209–229.

[48] P. Sheng, M. Zhou, Immiscible–fluid displacement: Contact–line dynamics and the velocity–dependent capillary pressure, Phys. Rev. A 45 (1992) 5694.

[49] R. L.Hoffman, A study of the advancing interface, J. Colloid Interface Sci. 50 (1975) 228–241.

[50] R. G. Cox, The dynamics of the spreading of liquids on a solid surface. part1, J. Fluid Mech. 168 (1986) 169–194.

[51] T. Qian, Z.-P. Wang, P. Sheng, Molecular hydrodynamics of the moving contact line in two–phase immiscible flows, Comm. Comput. Phys. 1 (1) (2006) 1–52.

[52] W. Ren, W. E, Boundary conditions for the moving contact line problem, Phys. Fluids 19 (2007) 022101.

[53] H. Liu, S. Krishnan, S. Marella, H. Udaykumar, Sharp interface cartesian grid method ii: A technique for simulating droplet interactions with surfaces of arbitrary shape, J. Comput. Phys. 210 (2005) 32–54.

[54] M. Renardy, Y. Renardy, J. Li, Numerical simulation of moving contact line problems using a volume–of–fluid method, J. Comput. Phys. 171 (2001) 243–263.

[55] W. Villanueva, G. Amberg, Some generic capillary-driven flows, Int. J. Multiphase Flow 32 (2006) 1072–1086.

[56] V. V. Khatavkar, P. D. Anderson, H. E. H. Meijer, Capillary spreading of a droplet in the partially wetting regime using a diffuse–interface model, J. Fluid Mech. 572 (2007) 367–387.

[57] A. Y. Tong, Z. Wang, A numerical method for capillarity–dominant free surface flows, J. Comput. Phys. 221 (2007) 506–523.

[58] S. Grob, A. Reusken, An extended pressure finite element space for two–phase incompressible flows with surface tension, J. Comput. Phys. 224 (2007) 40–58.

[59] S. Ganesan, G. Matthies, L. Tobiska, On spurious velocities in incompressible flow problems with interfaces, Comput. Methods Appl. Mech. Engrg. 196 (2007) 1193–1202.

[60] E. Aulisa, S. Manservisi, R. Scardovelli, A novel representation of the surface tension force for two-phase flow with reduced spurious currents, Comp. Meth. Appl. Mech. Eng. 195 (2006) 6239–6257.

Part II

Included papers

Paper 1

A conservative level set method for contact line dynamics

Sara Zahedi, Gunilla Kreiss, and Katarina Gustavsson
Submitted to Journal of Computational Physics

A Conservative Level Set Method for Contact Line Dynamics

Sara Zahedi and Katarina Gustavsson
School of Computer Science and Communication,
Royal Institute of Technology, 100 44 Stockholm, Sweden
Gunilla Kreiss
Division of Scientific Computing, Department of Information Technology,
Uppsala University, 751 05 Uppsala, Sweden

Abstract

A new model for simulating contact line dynamics is proposed. We apply the idea of driving contact line movement by enforcing the equilibrium contact angle at the boundary, to the conservative level set method for incompressible two phase flow [J. Comput. Phys. 210 (2005) 225-246]. A modified reinitialization procedure provides a diffusive mechanism for contact line movement, and results in a smooth transition of the interface near the contact line without explicit reconstruction of the interface. We are able to capture contact line movement without loosing the conservation. Numerical simulations of capillary dominated flows in two space dimensions demonstrate that the model is able to capture contact line dynamics qualitatively correct.

1 Introduction

Accurate modeling and simulations of contact line movement are of interest in areas like lubrication, oil recovery, and immiscible fluid flow through porous media. Another example is the liquid phase sintering process [1] which is important for industrial operations such as grinding, drilling, and cutting. This process permits the formation of dense, pore–free carbides with superior properties such as high strength, hardness and toughness. An important part of the sintering process is wetting of the liquid onto solid particles. The wetting liquid acts on the solid particles to eliminate porosity and reduce interfacial energy. Simulations can contribute to a better understanding of the liquid phase sintering process and play an important role in the development of cutting tools.

Contact line dynamics is the movement of the intersection line between the interface of two immiscible fluids and a solid surface. An example of such a phenomenon is when a droplet with zero velocity is placed on a solid surface and starts to spread. As the drop spreads, the contact line will move until it reaches an equilibrium state determined by the surface energies of the interfaces involved. The angle between the interface at equilibrium and the solid surface is often referred to as the equilibrium or static contact angle (hereinafter, *the static contact angle*). For a more detailed discussion of this kind of phenomena see [2].

The immiscible and incompressible flow is described by the incompressible Navier–Stokes equations with the surface tension and gravity forces added as source terms. Some representation of the interface separating the two fluids is required, for example a level set function or markers, and an evolution equation for the advection of the interface. At all interior points, the physically correct model is that the interface is advected by the fluid velocity. At contact lines, however, the standard boundary condition for the velocity is no–slip, which means that the contact line cannot move. Most often the no–slip boundary condition is adequate. However, in many cases it is unphysical, which is reflected by the shear stress becoming singular. Also, in molecular dynamics simulations, see for example [3] and references therein, the existence of fluid-wall slipping has been observed. Over the years, various techniques to enable the contact line to move have been presented. A common approach is to allow the contact line to move by introducing a so called slip length [4, 5]. However, this technique has difficulties in capturing flows dominated by capillary forces. Typically such flows are driven by a deviation of the contact angle from the static angle.

Another approach was suggested by Jacqmin [6] who used a coupled Cahn–Hilliard/ Navier–Stokes formulation consisting of the Navier–Stokes equations coupled to a system of two parabolic equations for the chemical potential and the concentration of one of the fluids. In this model, the interface is modeled as a layer with a continuous transition from one fluid to the other, and the contact line as a small region; the part of the interface layer close to the solid boundary. When the angle of the interface differs from the static angle the contact line moves by a diffusive process on a fast time scale in the small region at the boundary so that the contact angle is adjusted to the static value. The result is a region close to the boundary with high curvature of the interface, and consequently a strong surface tension force. The fact that the interface can move by diffusive processes eliminates the need for modeling fluid slip. A drawback with the model suggested by Jacqmin is that the interface must be highly resolved to achieve accuracy. This formulation has been used to investigate basic wetting phenomena, dominated by capillary forces [7].

Another alternative is to use the mismatch between the dynamic and static contact angles directly to obtain movement of the contact line. This approach has for example been used in a volume–of–fluids and a level set context in [8] and [9], respectively. In these methods the interface is explicitly reconstructed close to the contact line so that the contact angle takes the prescribed value.

In this paper we focus on capillary dominated flow. We introduce an extension of the conservative level set method presented in [10] and [11], which is capable of capturing contact line dynamics while keeping the conservation property. We formulate a reinitialization procedure for the level set function which makes us able to control contact angles at solid boundaries. For the velocity we use a no–slip boundary condition. The procedure does not involve explicit reconstruction. It is formulated as a partial differential equation, and the contact line moves by diffusion. Similarly to the work by Jacqmin, a contact line region with high curvature is achieved and the surface tension forces the fluids to move. Compared to methods that use the Cahn–Hilliard/Navier–Stokes formulation, our conservative level set method requires less computational effort since a relatively coarse grid is sufficient to capture the dynamics, at least away from contact lines. In this paper we consider problems in two space dimensions where contact lines become contact points.

The paper is organized as follows. In Section 2 we present the incompressible Navier–Stokes equations for two–phase flow, together with a description of our interface representation. We also introduce the main idea behind our new contact line capturing model and present the new reinitialization. In Section 3, the new technique is tested numerically on a scalar model problem. We discuss how to interpret and choose values for the model parameters. In Section 4, we apply the new model to channel flow and the spreading of a droplet on a flat plate. We show that if the contact point region is sufficiently small , the large scale interface movement is essentially independent of the size of this region. Section 5 contains a summary and a discussion of the results.

2 Mathematical model

The fundamental model of two–phase flow with contact line dynamics consists of the incompressible Navier–Stokes equations coupled with evolution of the interface. At contact points special treatment is needed to provide a mechanism for interface movement. In this section we will present this model.

2.1 The incompressible Navier–Stokes equations

Assume that a given domain Ω is occupied by two immiscible fluids separated by an interface Γ. The equations describing this immiscible flow are the incompressible Navier–Stokes equations with the contributions of the surface tension and gravity forces added as source terms:

$$(\rho\mathbf{u})_t + \nabla\cdot(\rho\mathbf{u}\mathbf{u}) = -\nabla p + \nabla\cdot\left(\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) + \rho g\mathbf{e}_g + \sigma\kappa\mathbf{n}\delta_\Gamma \qquad (1)$$

$$\nabla\cdot\mathbf{u} = 0. \qquad (2)$$

Here $\mathbf{u}$, p, ρ, and μ denote velocity, pressure, density, and viscosity, respectively. In general ρ and μ are discontinuous across the interface separating the two fluids. The curvature and normal of the interface Γ are denoted by κ and $\mathbf{n}$, and δ_Γ is a Dirac delta function with support on Γ. Its action on any smooth test function v is given by

$$\int_\Omega \delta_\Gamma v d\Omega = \int_\Gamma v d\Gamma. \tag{3}$$

The direction of gravitation is denoted by $\mathbf{e}_g$, g is the gravity, and σ is the surface tension coefficient.

Non–dimensionalize by introducing

$$\mathbf{x} = l_{\text{ref}}\mathbf{x}', \mathbf{u} = u_{\text{ref}}\mathbf{u}', t = t_{\text{ref}}t', p = p_{\text{ref}}p', \rho = \rho_{\text{ref}}\rho', \mu = \mu_{\text{ref}}\mu'. \tag{4}$$

Here t_{ref}, p_{ref}, ρ_{ref}, μ_{ref}, l_{ref}, and u_{ref} are constant reference time, pressure, density, viscosity, length, and velocity. Omitting the primes we obtain

$$\frac{\rho_{\text{ref}}u_{\text{ref}}}{t_{\text{ref}}}(\rho\mathbf{u})_t + \frac{\rho_{\text{ref}}u_{\text{ref}}^2}{l_{\text{ref}}}\nabla\cdot(\rho\mathbf{u}\mathbf{u}) = \tag{5}$$
$$= -\frac{p_{\text{ref}}}{l_{\text{ref}}}\nabla p + \frac{\mu_{\text{ref}}u_{\text{ref}}}{l_{\text{ref}}^2}\nabla\cdot\left(\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) + \rho_{\text{ref}}\rho g\mathbf{e}_g + \frac{\sigma}{l_{\text{ref}}^2}\kappa\mathbf{n}\delta(\Gamma)$$

$$\nabla\cdot\mathbf{u} = 0. \tag{6}$$

Suitable choices of the reference time t_{ref} and the reference pressure p_{ref} are

$$u_{\text{ref}}\cdot t_{\text{ref}} = l_{\text{ref}}, \quad p_{\text{ref}} = \rho_{\text{ref}}u_{\text{ref}}^2. \tag{7}$$

Introduce the dimensionless Reynolds, Capillary, and Froude numbers, given by

$$Re = \frac{\rho_{\text{ref}}u_{\text{ref}}l_{\text{ref}}}{\mu_{\text{ref}}},\ Ca = \frac{\mu_{\text{ref}}u_{\text{ref}}}{\sigma},\ \text{and}\ Fr = \frac{u_{\text{ref}}}{\sqrt{l_{\text{ref}}g}}, \tag{8}$$

respectively. After dividing (5) by $\frac{\mu_{\text{ref}}u_{\text{ref}}}{l_{\text{ref}}^2}$ we obtain

$$((\rho\mathbf{u})_t + \nabla\cdot(\rho\mathbf{u}\mathbf{u})) = \tag{9}$$
$$= -\nabla p + \frac{1}{Re}\nabla\cdot\left(\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) + \frac{1}{Fr^2}\rho\mathbf{e}_g + \frac{1}{ReCa}\kappa\mathbf{n}\delta(\Gamma).$$

In order to complete the formulation, a representation of the interface and a model for its motion are needed. This will be addressed in the next section.

2.2 An improved level set representation of the interface

Level set representations of the interface in two phase flow have been used in many flow simulations, see for instance [12] or [13]. The basic idea is that if a level set of a function ϕ_0 defines the interface at $t = 0$, the same level set of the solution to

$$\phi_t + \nabla\cdot(\phi\mathbf{u}) = 0, \quad \phi(\cdot, 0) = \phi_0, \tag{10}$$

defines the interface at later times.

The method proposed in this paper is an extension of the conservative level set method, introduced in [10]. Instead of the signed distance function [12, 13] usually used to define the interface, this method uses a regularized indicator function ϕ. The indicator function takes the value 0 in one fluid and the value 1 in the other fluid. The 0.5–level defines the interface. In this way, good conservation properties can be achieved simply by using a conservative discretization. It is essential to use no–flux boundary conditions for ϕ except at in– or out–flow boundaries. The shape of the regularized step function is controlled in a reinitialization step. This step is modeled by a partial differential equation where a non–linear term, resembling a compressive limiter (see [14]), is balanced by diffusion in the normal direction. The normal direction is given by the gradient of the level set function. In the following we introduce a new reinitialization that can be used to also capture contact line dynamics. This is a generalization of the reinitialization procedures used in [10] and [11].

Our new model mimics the Cahn–Hilliard/Navier–Stokes model in the following way. We have included diffusion of the phases, and this diffusion moves the contact point so that the angle between the interface, defined by the 0.5–level set of ϕ, and the boundary always equals the static contact angle α_s, see Fig. 1. A specific static contact angle corresponds to a specific normal vector to the interface $\mathbf{n}_{\alpha_s}$. Since the gradient of the level set function is related to the normal of the interface by $\mathbf{n} = \nabla\phi/|\nabla\phi|$, the condition on the angle can be formulated as a condition on the gradient. We cannot, however, directly prescribe the gradient of the level set function at the boundary without creating a flux of ϕ over the boundary that would destroy the conservative properties of the model.

Instead we introduce a regularized normal vector field $\mathbf{n}$, satisfying

$$\tilde{\mathbf{n}} - \nabla \cdot (\gamma^2 \nabla \tilde{\mathbf{n}}) = \frac{\nabla\phi}{|\nabla\phi|}, \quad \mathbf{n} = \frac{\tilde{\mathbf{n}}}{|\tilde{\mathbf{n}}|}, \tag{11}$$

with Dirichlet boundary conditions, $\tilde{\mathbf{n}} = \mathbf{n}_{\alpha_s}$ along solid walls, where $\mathbf{n}_{\alpha_s}$ is the normal corresponding to the static contact angle. The regularization parameter γ should preferably be chosen $\gamma \ll L$, where L is the length of typical features in the problem at hand. The regularized normal vector will have approximately the same direction as the gradient of the unperturbed ϕ, except along the boundary, where a boundary layer with thickness proportional to γ will form. In the boundary layer the regularized normal vector field changes smoothly to the prescribed $\mathbf{n}_{\alpha_s}$. This regularized normal vector field is then used in the reinitialization step in the following fashion:

$$\phi_{\hat{t}} + \nabla \cdot (\phi(1-\phi)\mathbf{n}) - \nabla \cdot (\varepsilon_n(\nabla\phi \cdot \mathbf{n})\mathbf{n}) - \nabla \cdot (\varepsilon_\tau(\nabla\phi \cdot \mathbf{t})\mathbf{t}) = 0. \tag{12}$$

Here $\mathbf{n}$ is the normalized and regularized gradient of ϕ, satisfying (11), and $\mathbf{t}$ is the tangent, orthogonal to $\mathbf{n}$. Further, ε_n is a diffusion parameter in the normal direction and ε_τ is a diffusion parameter in the tangential direction. The second

term in (12) represents compression in the normal direction, while the third models diffusion in the normal direction, and the last term models diffusion in the tangential direction. A balance between the second and the third terms establishes a layer of thickness proportional to ε_n, where ϕ changes from 0 to 1. For the method to be accurate ε_n must be much smaller than typical geometrical features of the interface. In the original work on the conservative level set method [10] the standard isotropic diffusion was used. This corresponds to choosing $\varepsilon_\tau = \varepsilon_n = \varepsilon$ in equation (12). In the subsequent work, presented in [11], $\varepsilon_\tau = 0$. The idea was to avoid unnecessary movement of the 0.5–level set in the tangential direction. However, the tangential diffusion is essential in the contact point region. It ensures that the level set function is constant in the tangential direction. At the boundary this implies that the 0.5–level set forms the prescribed angle to the boundary.

Using this model, the interface should be advected with the fluid velocity everywhere except in the contact point region. There the reinitialization step establishes a boundary layer in ϕ, of thickness proportional to the regularization parameter γ. The angle of the interface (the 0.5–level set of ϕ) to the boundary changes over the boundary layer from the static contact angle α_s to the apparent contact angle α_a. The apparent contact angle is the angle formed between the boundary and the interface away from the contact point region, see Fig. 1. Close to the boundary the curvature of the interface will in general be large if γ is small. As long as the apparent angle remains the same the reinitialization step will ensure that the shape of the interface in the contact point region is constant. This means that the contact point moves with the same speed as the tangential fluid velocity just outside the contact point region (region A in Fig. 1).

We expect the velocity of the contact point to be determined by a balance between surface tension and viscous forces. When the two fluids have the same viscosity the following simple analysis shows that for small γ the speed is essentially independent of γ. Let the dimensionless speed of the contact point be U. Then, since the fluid velocity changes from 0 to U over a layer of thickness proportional to γ, the viscous stress in the tangential direction, in the Navier–Stokes equations (9), is proportional to U/γ^2. The resulting viscous force on A is obtained by integrating over A, yielding $F_{\text{visc}} \sim U$. The term representing surface tension is $\frac{1}{Ca}\kappa\mathbf{n}\delta(\Gamma)$, with $Ca = \frac{\mu_{\text{ref}} u_{\text{ref}}}{\sigma}$. The curvature κ depends on the static and apparent angles in some non–trivial way, but scales inversely with γ, that is $\kappa \sim \frac{f(\alpha_a,\alpha_s)}{\gamma}$. Integrating over A yields a tangential force $F_{\text{surface}} \sim \frac{f(\alpha_a,\alpha_s)}{Ca}$. The two forces, F_{visc} and F_{surface}, must be balanced, yielding $U \sim \frac{f(\alpha_a,\alpha_s)}{Ca}$. This indicates that at least to lowest order in γ, U is independent of γ. Also note that in the limit $\gamma \to 0$ the shear stress is still singular, but now the singularity is integrable. In the next section these ideas are tested numerically for a model problem.

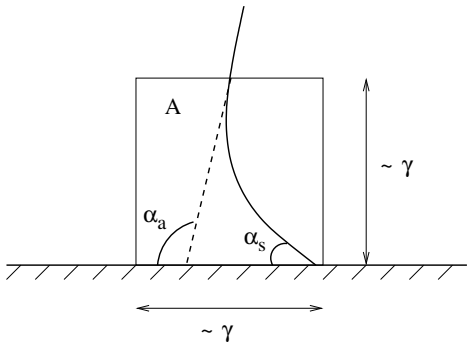


Figure 1: Definition of different angles.

3 Model problem

In this section numerical tests are performed with the intention to show how different parameters in the new model for the reinitialization, introduced in Section 2.2, influences the level set function ϕ and to give an idea of how they should be chosen. In this section, we are only interested in the contact point region and therefore we only consider the reinitialization process, and we do not solve any advection equation for ϕ. We recall that the reinitialization process is governed by equations (11) and (12).

In this section we model a contact point region of a real problem, scaled so that the curvature of the interface away from the contact point is small compared to unity. Also, by a simple scaling argument it is clear that only the ratio between the modeling parameters is important for the shape of the interface in the contact point region. We therefore keep ε_n fixed in this section, and vary only γ and ε_τ.

Throughout this section the computational domain is $\{(x, y) : 0 \leq x \leq 2, 0 \leq y \leq 2\}$. Initially

$$\phi(x, y) = \frac{1}{1 + e^{\frac{1-x}{\varepsilon_n}}}$$

which gives an interface at $x = 1$ normal to the x–axis. Thus, the normal at the interface $\frac{\nabla\phi}{|\nabla\phi|} = (1, 0)$, corresponding to an apparent contact angle of $90°$. With this normal, equation (11) can be solved analytically for the regularized normal vector field.

Let $\tilde{\mathbf{n}} = (n_1, n_2)$ be the component of the normal vector in the x and y direction respectively. Then

$$n_1(y) = c_1 e^{y/\gamma} + c_2 e^{-y/\gamma} + 1 \tag{13}$$

$$n_2(y) = c_3 e^{y/\gamma} + c_4 e^{-y/\gamma} \tag{14}$$

where the constants c_1, c_2, c_3, and c_4 are given by the boundary conditions,

$$n_1(0) = \sin(\alpha) \quad \frac{dn_1}{dy}(2) = 0$$
$$n_2(0) = \cos(\alpha) \quad \frac{dn_1}{dy}(2) = 0.$$

Here α is the prescribed contact angle of the interface at the contact point. In this work we prescribe the contact angle to be equal to the static contact angle α_s.

The reinitialization equation (12) is discretized on a uniform mesh with grid size h in both x– and y–directions using a conservative second order finite difference scheme. On the boundaries, the numerical flux functions are set to zero. For the time stepping we use a second order Runge–Kutta scheme and for stability reasons, the time step is chosen as $\Delta t = h^2/(2(\varepsilon_n + \varepsilon_\tau))$. The reinitialization equation is solved until the relative residual of ϕ, $\frac{\|\phi^{n+1} - \phi^n\|_2}{\|\phi^n\|_2} \leq 10^{-5}$, which we consider to be the steady solution. In this section we will use a mesh size of $h = 0.005$ in all numerical experiments. We have performed mesh convergence studies which showed that the value of h chosen here is sufficiently small for the computations in this section to be well resolved.

3.1 The regularization parameter γ

We will now investigate how the parameter γ affects the contact point region. To study the influence of γ on the level set function ϕ, the reinitialization equation (12) is solved with a regularized normal vector field for different values of the parameter γ. The other parameters are kept fixed at $\varepsilon_n = 8h$ and $\varepsilon_\tau = 6\varepsilon_n$.

The 0.5–level set of ϕ defining the interface after the reinitialization is shown in Fig. 2 for different values of γ and for a prescribed contact angle of $\alpha_s = 45°$. We can clearly see the influence of γ on the interface. A boundary layer proportional to γ is formed in the reinitialization process. In this layer the normal of the interface varies smoothly from the prescribed normal at $y = 0$ to the gradient of the unperturbed level set function ϕ. Reinitialization with small values of γ will produce an interface with large curvature. Also, the position of the point where the 0.5–level set intersects the boundary $y = 0$ depends on the value of γ. The contact angle at steady state comes closer to the prescribed angle when γ is increased.

In Fig. 3 we show how the contact angle changes with time for the different values of γ. Starting from $90°$ the contact angle decreases with time towards the prescribed static contact angle. We can see a similar behavior for all four values of γ and that γ has a small impact on the obtained contact angle.

To conclude this section, γ should be small, in the order of ε_n, so that the contact point region does not become a substantial part of the domain. However, a smaller γ generally gives larger curvature. The mesh size h must be chosen small enough to resolve the curvature.

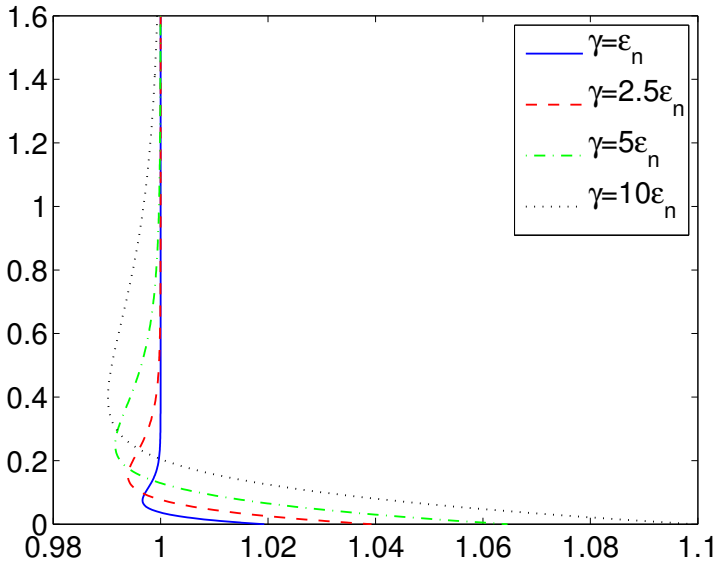


Figure 2: The steady 0.5–level set of ϕ for different values of γ. Here $\varepsilon_n = 8h$, $\varepsilon_\tau = 6\varepsilon_n$ and $\alpha_s = 45°$. The contact angle of the interface at the boundary where the 0.5–level set intersects $y = 0$, has reached approximately 50, 49, 48, and $46°$ for $\gamma/\varepsilon_n = 1$, 2.5, 5, and 10 respectively.

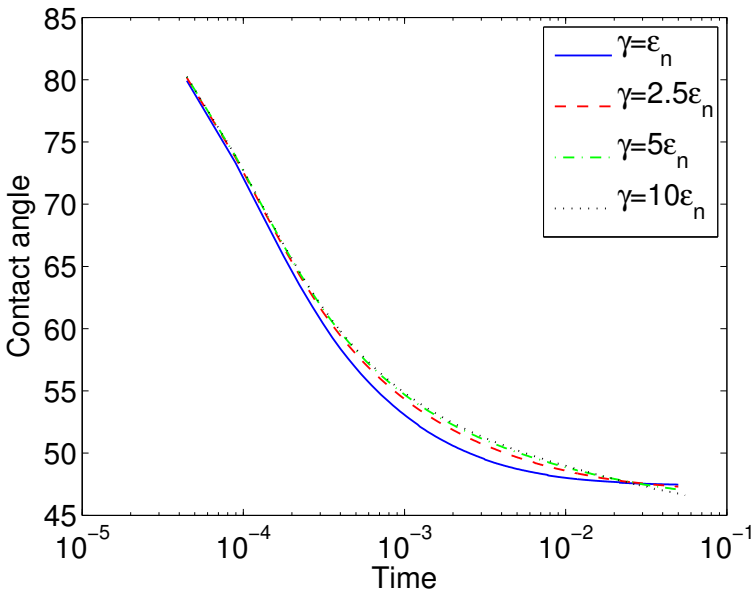


Figure 3: The contact angle as a function of time in the reinitialization process for different values of γ. Here $\varepsilon_n = 8h$, $\varepsilon_\tau = 6\varepsilon_n$ and $\alpha_s = 45^\circ$.

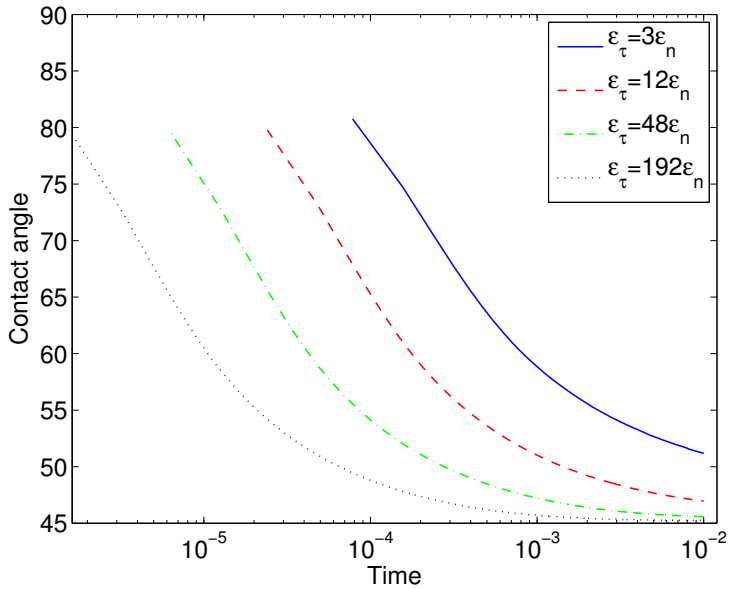


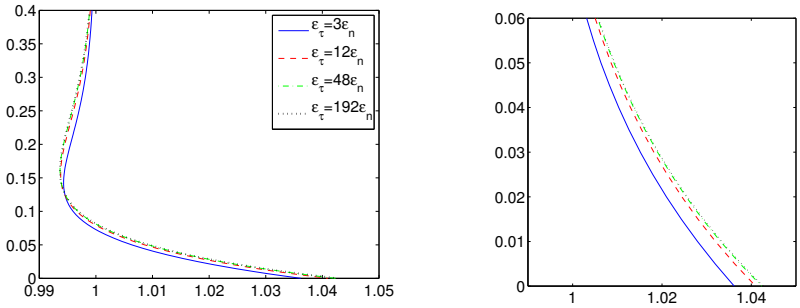
Figure 4: Contact angle as a function of time in the reinitialization process for different values of ε_τ. Here we have used fixed values of $\varepsilon_n = 8h$ and $\gamma = 2.5\varepsilon_n$ and $\alpha_s = 45°$. Note that for the smaller values of ε_τ, the angle presented here has not yet reached the steady value. In Fig. 6 you can see how close the steady state angle comes to the prescribed value for different ε_τ–values.

3.2 The tangential viscosity parameter ε_τ

Diffusion in the tangential direction, given by the tangential viscosity parameter, ε_τ, is essential for the ability of the interface to form the prescribed angle to the boundary. To study the influences of ε_τ on the 0.5–level set and more specifically on the angle of the interface to the boundary we have performed simulations with different values of ε_τ for fixed values of $\varepsilon_n = 8h$ and $\gamma = 2.5\varepsilon_n$. The prescribed static angle is $\alpha_s = 45°$.

In Fig. 4 we see that the value of the tangential viscosity has a large impact on the contact angle and as ε_τ increases, the contact angle converges faster towards the static angle α_s. As the tangential viscosity increases, the shape of the foot converges, as can be seen in Fig. 5. However, a larger value of ε_τ will also affect the level set function ϕ away from the contact point region.

The same numerical test were made also for a prescribed static contact angle of $\alpha_s = 25°$. The behavior of the contact angle is essentially the same. As the tan-



(a) The foot region

(b) Close up at the boundary

Figure 5: The steady 0.5–level set of ϕ for different values of ε_τ. Here $\varepsilon_n = 8h$, $\gamma = 2.5\varepsilon_n$ and $\alpha_s = 45°$.

gential viscosity increases the contact angle converge faster towards the static angle. However, larger values of the tangential viscosity is needed when the difference between α_s and the initial contact angle is large. This can be seen in Fig. 6 where the contact angle of the 0.5–level set at steady state is presented as a function of the ratio $\varepsilon_\tau/\varepsilon_n$, for $\alpha_s = 45$ and $25°$. We have found that varying ε_τ or ε_n without changing their ratio does not affect the obtained contact angle. This means that an improved convergence towards the static contact angle can be obtained either by increasing ε_τ or decreasing ε_n while keeping the other one fixed. This exposes a trade off. On one hand, we do not want to increase ε_τ too much since a large ε_τ will affect the level set function ϕ also in regions away from the foot region. On the other hand, decreasing ε_n requires a finer mesh. A solution could be to let ε_τ vary with the distance from the boundary. In this way we would get tangential diffusion only in the foot region.

3.3 Conclusions

A general conclusion from the results presented here is that larger values of γ and ε_τ result in contact angles closer to the prescribed angles. The value of the regularization parameter γ does not have as large effect on the contact angle as the tangential viscosity parameter, ε_τ, but γ has a larger impact on the size of the foot region.

Also, at a fixed set of parameters, the discrepancy between the prescribed contact angle α_s and the angle of the interface at the boundary, increases as the difference between α_s and the initial contact angle increases. Larger values of ε_τ are then needed in order to obtain the correct angle, however at the cost of the parameters influencing a larger region. A remedy for this could be to add tangential viscosity only in the contact point region.

An improved convergence towards the static contact angle can also be obtained

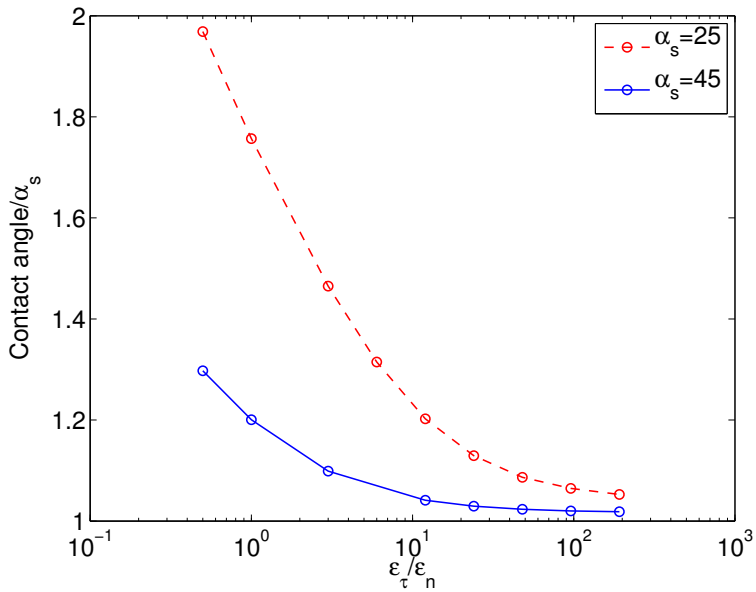


Figure 6: The contact angle normalized by the static angle as a function of tangential viscosity, ε_τ, for two different values of α_s. Here $\varepsilon_n = 8h$, $\gamma = 2.5\varepsilon_n$.

by decreasing ε_n. This would, however, require a finer mesh.

4 Application

In this section we discuss the numerical treatment and results for two types of applications: channel flow and droplets on flat plates. Both involve two immiscible fluids and moving contact lines. In the channel flow and one of the droplet on a flat plate applications, capillary effects dominate. In the second droplet on a flat plate application, we also see the effect of inertia. The proposed model is used to simulate the contact line dynamics.

4.1 Numerical treatment

The motion of the two immiscible fluids is governed by the incompressible Navier–Stokes equations. As in [10, 11], we solve the Navier–Stokes equations by using a projection method with an added pressure stabilization term as introduced by Guermond and Quartapelle [15]. The surface tension effect is treated as a force, $F = \sigma\kappa\mathbf{n}\delta_\Gamma$ added to the Navier–Stokes equations. We have implemented two approaches to model the surface tension force. In the first approach, the singular surface tension force is smoothed out over a finite thickness and is given by $F = \sigma\kappa\nabla\phi$ as proposed by Brackbill et al. [16]. In the second approach, the singular surface tension force is evaluated through a line integral along the interface Γ as suggested by Tornberg et al. [17]. Our model works well together with both approaches. However, in the first approach the delta function δ_Γ is regularized and if part of the regularization zone ends up outside the computational domain the accuracy will usually be lost. Therefore evaluation of a line integral may be favorable when the interface is close to a boundary.

At contact points diffusion of ϕ tangential to the interface is essential. The diffusion can be added in the reinitialization equation (12), or in the advection equation. In each time step the reinitialization equation (12) is solved until the relative residual of ϕ, $\frac{\|\phi^{n+1}-\phi^n\|}{\|\phi^{n+1}\|} \leq 10^{-5}$, which we consider to be the steady state solution.

As in [11] we use a second order semi–implicit discretization in time. Usually two reinitialization steps are required. However, in case a large change of the contact angle is needed, more steps are required to reach steady state.

The numerical simulations were carried out using FemLego, a parallel finite element code for the solution of partial differential equations [18]. All equations were discretized in space using piecewise linear functions.

4.2 Capillary dominated channel flow

Consider two fluids with the same density and viscosity, in a two dimensional channel initially separated by an interface, normal to the channel, without curvature. Thus,

the initial apparent contact angle is $\alpha_a = 90°$. We assume that the static contact angle α_s is smaller than $90°$. Clearly the initial condition is not a steady state solution. In the absence of an outer pressure gradient or prescribed flux, the only driving force is the capillary effect at the channel walls, which will move the interface to the right.

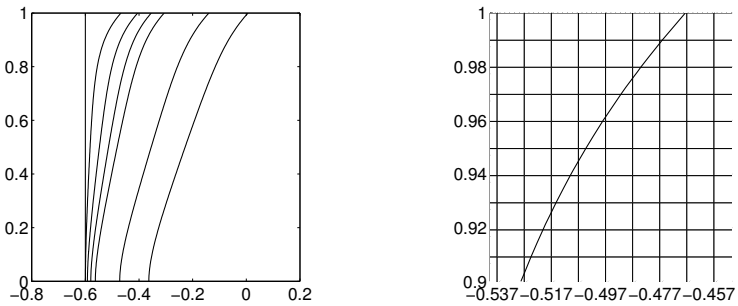
If the densities and viscosities of the two fluids are of similar size we expect that after an initial, transient process, the interface will move at a constant speed U. We expect the interface to consist of an interior part with curvature $\kappa_a = 2\cos(\alpha_a)$, and two small sections close to the walls with curvature $\kappa \sim \frac{f(\alpha_a - \alpha_s)}{\gamma}$, where f is some smooth function.

4.2.1 Computations

In this subsection we present computations that demonstrate the qualitative behavior of our model. In the computations we consider a channel of non–dimensional length 1 or 2 and width 1. The Capillary number and the Reynolds number are set to unity. At the channel walls we prescribe zero velocity, i.e. no slip, and the flux of the level set function is set to zero. The normal vectors at the lower and upper walls are prescribed so that the contact angles equal the static contact angles. At the inlet and outlet we use vanishing tangential velocity, vanishing normal derivative of the normal velocity, and Dirichlet conditions for the level set function and the pressure. Throughout this subsection we use $h = \Delta x = \Delta y = 0.01$, $\Delta\hat{t} = \Delta t = 0.05$, and $\varepsilon_\tau = \varepsilon_n = \varepsilon$.

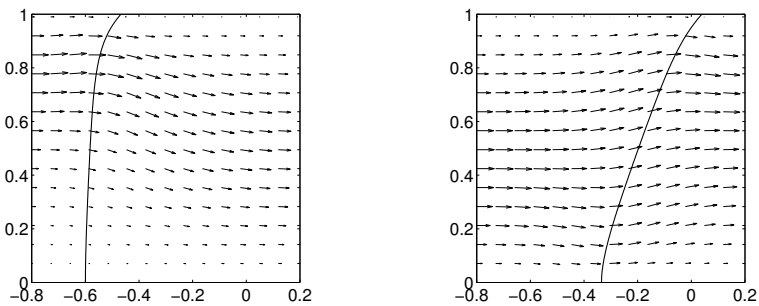
In Fig. 7 and 8 we see results for a case with static contact angle $\alpha_{s_1} = 45°$ at the upper channel wall, and static contact angle $\alpha_{s_2} = 90°$ at the lower wall. In this computation we used $\varepsilon = 2.5h$ and $\gamma = 3\varepsilon$. As expected, the capillary forces at the upper wall exhort a pull which sets the fluid into motion. In Fig. 7(a) we see how the interface (the 0.5–level set of ϕ) develops over time. In Fig. 7(b) a detail of the interface close to the upper wall at an early stage is shown. The angle between the interface and the upper wall is somewhat larger than $45°$. Two representative velocity fields are plotted in Fig. 8. The interface develops into a steady shape after an initial transient period of time. Here we note that away from the interface the velocity field approaches the standard pipe flow parabola.

In Fig. 9 we have plotted results when the static angle is the same at the upper and the lower channel walls, $\alpha_{s_1} = \alpha_{s_2} = 45°$ and $\alpha_{s_1} = \alpha_{s_2} = 25°$, respectively. In these computations the channel length is 2, and we used $\varepsilon = 0.05$, $\gamma = 0.025$. Also in these cases, after a transient period, a steady shape develops which moves at a constant speed. As expected, the interface moves faster when the static contact angle is smaller. In Fig. 10 we have plotted the velocity field, pressure field and the level set function at $t = 10$ for the case $\alpha_{s_1} = \alpha_{s_2} = 25°$.



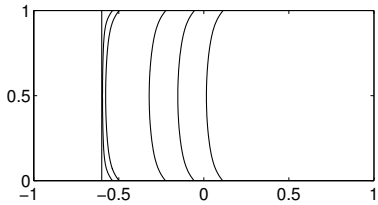
(a) (b)

Figure 7: Capillary dominated channel flow. The static contact angles at the upper and lower wall are 45° and 90° respectively. (a) The location of the interface, represented by the 0.5–level set of ϕ, at $t = 0, 1, 2, 3, 4, 8$, and 12. (b) Detail of the 0.5–level set of ϕ close to the upper wall at $t = 1$ with the grid.

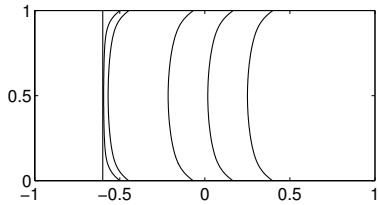


(a) $t = 1$ (b) $t = 13$

Figure 8: The same capillary dominated channel flow as in Fig. 7. The normalized velocity field at two different times, $t = 1$ and 13.

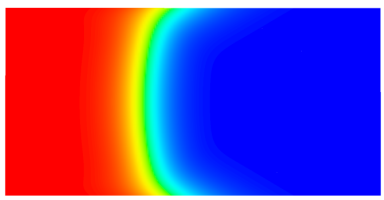


(a) $\alpha_{s_1} = \alpha_{s_2} = 45^\circ$

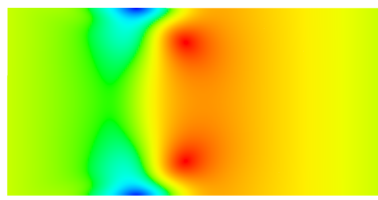


(b) $\alpha_{s_1} = \alpha_{s_2} = 25^\circ$

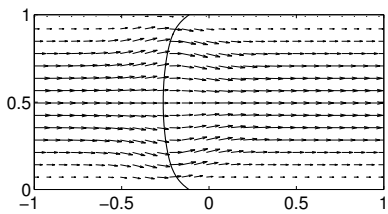
Figure 9: The location of the interface (the 0.5–level set of ϕ) at $t = 0, 1, 2, 10, 15,$ and 20. The angles at the upper and the lower channel walls are $\alpha_{s_1} = \alpha_{s_2} = 45^\circ$ in Panel (a) and $\alpha_{s_1} = \alpha_{s_2} = 25^\circ$ in Panel (b).



(a) The level set function, ϕ



(b) Pressure field



(c) Normalized velocity field

Figure 10: Solution at $t = 10$ for $\alpha_{s_1} = \alpha_{s_2} = 25^\circ$.

4.2.2 Convergence studies

In Section 3 we saw that the shape of the contact point region, or the foot region, is determined by the relation between the model parameters γ, ε_τ, and ε_n. The purpose of this section is to show that when the size of the foot region is decreased, at some point, the movement of the interface is no longer affected.

To obtain an angle which deviates at most 10% from the desired static contact angle of 45° we use the suggested ratios between the modeling parameters, $\varepsilon_\tau/\varepsilon_n = 3$, $\gamma/\varepsilon_n = 2.5$, see Fig. 6. The time step is $\Delta t = 10^{-3}$ and $\Delta\hat{t} = \frac{h^2}{\varepsilon_n+\varepsilon_\tau}$. Initially the interface is a straight line at $x = -0.3$.

In the first set of computations the resolution is set to $h = 0.0025$. Fig. 11 shows the results at $t = 3$ for three values of ε_n, differing by successive factors of two. For the smallest value of ε_n we have $\varepsilon_n = 3\sqrt{2}h = 0.0106$.

We compute the order of convergence by comparing the position of 400 points along the interface at $t = 1, 2, 3, 4, 5$ for different values of the modeling parameters. Let x_i^ε be the x–coordinate of the interface at $y = y_i$ at a particular time for $\varepsilon_n = \varepsilon = 0.0106$. If we assume the position of the interface to depend on ε as

$$x_i^\varepsilon = x + \alpha\varepsilon^p + \mathcal{O}(\varepsilon^{p+1}), \tag{15}$$

we can estimate the convergence rate p from x_i^ε, $x_i^{2\varepsilon}$, and $x_i^{4\varepsilon}$ by

$$p \approx \log\left(\frac{\sum_{i=1}^{400} |x_i^{4\varepsilon} - x_i^{2\varepsilon}|}{\sum_{i=1}^{400} |x_i^{2\varepsilon} - x_i^{\varepsilon}|}\right) / \log(2). \tag{16}$$

The results of such computations for $t = 1, 2, 3, 4, 5$ indicate linear convergence for the position of the interface, see Table 1. We also obtained linear convergence in the velocity. The velocity was computed by the difference between the x–coordinates of the interface at $y = 0.5$ at two different times divided by the time difference.

In the second set of computations we varied the resolution in space, but kept the modeling parameters fixed with $\varepsilon_n = 4\varepsilon = 0.0424$, $\varepsilon_\tau/\varepsilon_n = 3$, and $\gamma/\varepsilon_n = 2.5$. In Fig. 12 the results for three values of h, differing by successive factors of $\sqrt{2}$, are plotted. Convergence rates are computed as above, and indicate linear convergence, see Table 1.

4.3 Droplet on a flat plate – Capillary dominated

We consider here a two–dimensional droplet with diameter l_{ref}, density ρ_{ref}, and viscosity μ_{ref}, lying on a solid surface surrounded by another liquid with the same density and viscosity. The static contact angle is $\alpha_s = 25^\circ$. Capillary effects dominate and gravity is neglected. We will simulate how this droplet wets the surface. This wetting phenomenon was previously studied using a phase field method by Villanueva and Amberg [7].

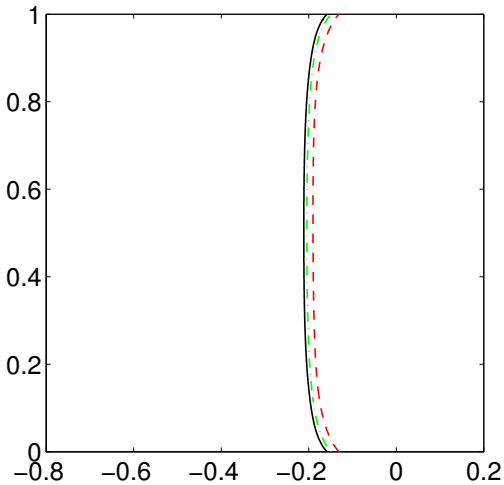


Figure 11: Comparison between the 0.5–level set of ϕ at $t = 3$ for $\alpha_{s_1} = \alpha_{s_2} = 45°$. The width of the interface $\varepsilon_n = 0.0106$, the diffusion parameter $\varepsilon_t = 3\varepsilon_n$, and the regularization parameter $\gamma = 2.5\varepsilon_n$ were all scaled with 2 and 4. The mesh resolution is fixed to be $h = 0.0025$. The contour furthest to the right corresponds to the largest parameter values.

	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$
$\varepsilon \to 0$	1.0912	1.0401	1.003	0.9702	0.9616
$h \to 0$	0.6831	0.9295	1.1576	1.3031	1.0696

Table 1: Convergence rate for the interface position at different times. The first row shows the convergence rates when the model parameters ε_n, ε_τ, and γ are decreased. The mesh size h is fixed. The second row shows the convergence rates when the mesh size is decreased. The time step in the reinitialization $\Delta\hat{t} = \frac{h^2}{\varepsilon_n + \varepsilon_\tau}$.

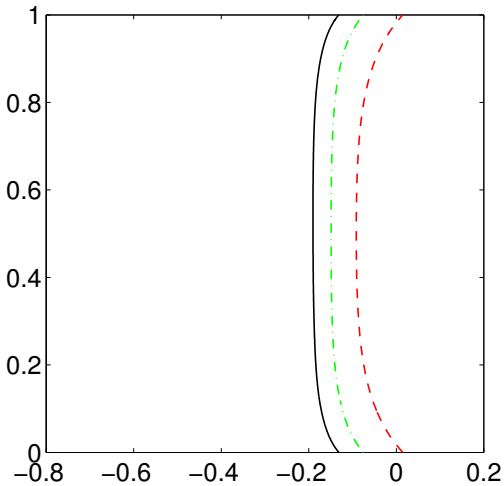


Figure 12: Comparison between the 0.5–level set of ϕ at $t = 3$ for $\alpha_{s_1} = \alpha_{s_2} = 45°$. Varying the mesh resolution: $h = 0.0025, 0.0025\sqrt{2}, 0.005$ but keeping all the other parameters fixed.

Experiments with gas/liquid/solid systems where the viscosity ratio $\lambda = \frac{\mu_M}{\mu_L}$ ranges from 10^{-5} to 10^{-8} has been performed by Hoffman [19]. The experiments show that the apparent contact angle is primarily a function of the capillary number Ca^*. Later Cox [20] presented a more general analysis of the dynamics of wetting. We will compare our results with data from Cox's theory and the phase field simulation.

4.3.1 Cox's Theory

Given two immiscible fluids with viscosity ratio λ, Cox's theory states that at a leading order in Ca^*,

$$g(\alpha_D, \lambda) - g(\alpha_s, \lambda) = Ca^* \ln(\delta^{-1}), \tag{17}$$

where α_D is the macroscopic dynamic contact angle, α_s is the static contact angle, δ is a small constant and the function $g(\alpha, \lambda)$ is given by

$$g(\alpha, \lambda) = \int_0^\alpha \frac{d\alpha}{f(\alpha, \lambda)} \tag{18}$$

where

$$f(\alpha,\lambda)=\frac{2\sin\alpha(\lambda^2(\alpha^2-\sin^2\alpha)+2\lambda(\alpha\pi-\alpha)+\sin^2\alpha)+(\pi-\alpha)^2-\sin^2\alpha)}{\lambda(\alpha^2-\sin^2\alpha)((\pi-\alpha)+\sin\alpha\cos\alpha)+((\pi-\alpha)^2-\sin^2\alpha)(\alpha-\sin\alpha\cos\alpha)}. \tag{19}$$

It is interesting to note that when Hoffman plotted the observed macroscopic contact angle α_D versus the capillary number Ca^* plus a shift he observed that his experimental results fell on a single curve. He did not, however, provide an explicit mathematical form for the function $\alpha_D(Ca^*)$. Later Cox showed good agreement between his formula in (17) with $\delta = 10^{-4}$ and Hoffman's curve for all values of α_D, except those very close to 180°. We have confirmed that a value of $\delta \approx 10^{-4}$ minimizes the sum of squares of the relative difference between the contact–line speeds Ca^* obtained from the experiments by Hoffman 1975 using silicone fluid with $\lambda = 0$ and $\alpha_s = 0$ and those obtained from Cox's theory. In the following we will use $\delta = 10^{-4}$.

4.3.2 Computations

The computational domain is in non–dimensional coordinates $\{(x,y) : -2 \le x \le 2, 0 \le y \le 2\}$ and the initial drop is symmetric around $x = 0$ and has non–dimensional radius $r = 0.5$. At initial time the drop is in contact with the solid wall at an angle of 156°. The Reynolds number $Re = \frac{\rho_{\text{ref}} u_{\text{ref}} l_{\text{ref}}}{\mu_{\text{ref}}} = 1$ and the Capillary number $Ca = \frac{2\sqrt{2}\mu_{\text{ref}} u_{\text{ref}}}{3\sigma} = 1$. The simulation of the wetting is performed using a mesh consisting of regular triangles with 270×135 nodes and a time step $\Delta\hat{t} = \Delta t = 0.0011$. The liquid–liquid interface (the 0.5–level set of ϕ) is advected by solving the advection equation (10) with a diffusion term equal to 0.0011. In the reinitialization described in equation (12) the viscosity parameter in the normal direction is set to $\varepsilon_n = 4.5h$ where h is the mesh size and the viscosity parameter in the tangential direction is set to $\varepsilon_\tau = 0$. The regularization parameter $\gamma = 2.5\varepsilon_n$. The boundary condition for the regularized normal vector field, equation (11), is set to

$$\mathbf{n}|_{y=0} = (-\text{sign}(x)0.4226, -0.9063)$$

aiming for a contact angle of 25° at the wall $y = 0$. At the other boundaries we use homogeneous Neumann conditions. The model parameters were chosen to keep the contact point region small without having to use a very fine mesh. However, we recall from Section 3 that the actual contact angle will be larger than the static contact angle of 25° with our choice of model parameters. We measure the dynamic contact angle that the liquid–liquid interface makes with the solid surface at the inflexion point of the foot as illustrated in Fig. 13. The wetting speed is given by dividing the difference in position of the intersection point (see Fig. 13) at two successive times by the time difference.

In Fig. 14 the wetting on the solid wall is illustrated for six different snapshots. The liquid–liquid interface immediately forms a foot due to the boundary condition

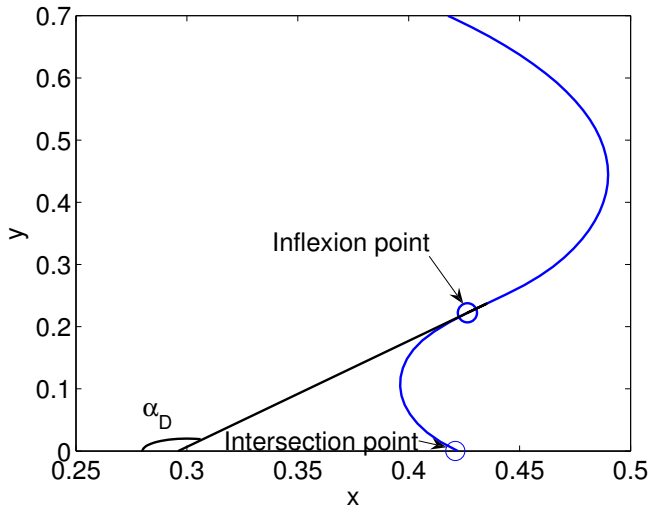


Figure 13: The dynamic contact angle is measured by considering the tangent at the inflexion point. The wetting speed is given from the position of the intersection point at different times.

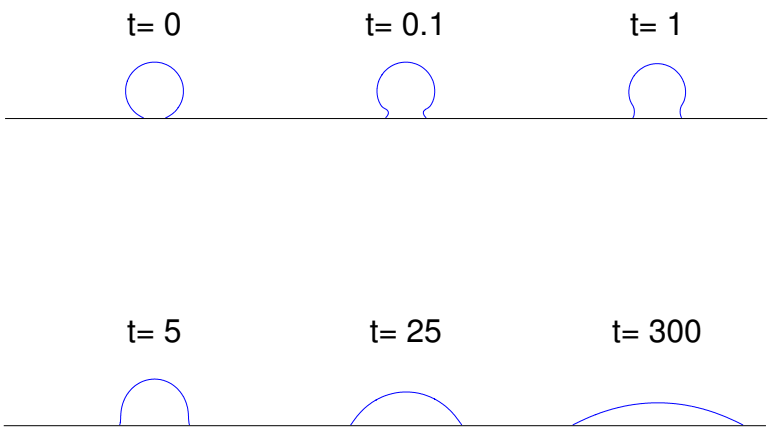


Figure 14: Wetting of a liquid drop on a solid surface. The interface separating the two fluids at times $t = 0, 0.1, 1, 5, 25, 300$ with $Re = 1$, $Ca = 1$. The contact angles at the different times are $\alpha \approx 156, 149, 120, 88, 55, 27$.

for the normal at the wall described above. The strong curvature in the foot region causes the fluid to move and the drop starts to wet the surface. The contact–line movement is fast in the initial steps but slows down as the drop approaches equilibrium. In Fig. 15 the change of the dynamic contact angle with time can be seen. The dynamic contact angle initially decreases rapidly with time but the decrease slows down as the angle approaches the equilibrium value.

In Fig. 16 the dynamic contact angle is plotted as a function of the Capillary number $Ca^* = \frac{2\sqrt{2}\mu_{\text{ref}}u}{3\sigma} = \frac{u}{u_{\text{ref}}}$, which in this case is equal to the wetting speed. The results are compared with a phase field simulation [7] and Cox's theory [20]. The qualitative behavior of the simulation in this work is in agreement with previous results.

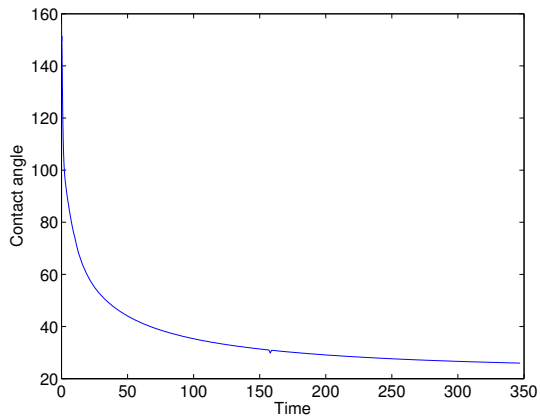


Figure 15: Dynamic contact angle versus dimensionless time for a liquid droplet.

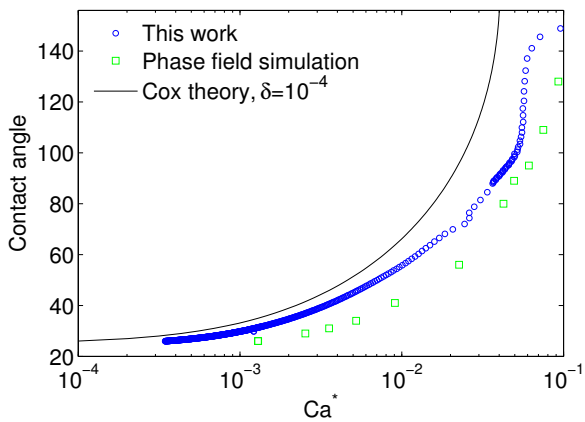


Figure 16: The contact angle as a function of the capillary number. The initial angle is 156°. We compare the result of this work with the phase field simulation in [7] and Cox's theory [20]. The parameter δ in Cox's theory is chosen to be 10^{-4}, see Section 4.3.1.

4.4 Droplet on a flat plate – Role of inertia

We consider here another two–dimensional droplet lying on a solid surface surrounded by another liquid with the same density and viscosity. The fluid is in rest and the configuration is in equilibrium except for the contact angle. The static contact angle is $\alpha_s = 70.53^\circ$. The dimensionless parameters are Reynolds number $Re = \frac{\rho_{\text{ref}} u_{\text{ref}} l_{\text{ref}}}{\mu_{\text{ref}}} = 20$ and the Capillary number $Ca = \frac{\mu_{\text{ref}} u_{\text{ref}}}{\sigma} = 0.03$. In this example inertia plays an important role. Gravity effects are neglected. A simulation of this problem has previously been performed by Renardy et al. using a volume–of–fluid method [8].

4.4.1 Computations

The computational domain is in non–dimensional coordinates $\{(x, y) : -3 \leq x \leq 3, 0 \leq y \leq 3\}$ and the initial drop is circular with non–dimensional radius $r = 1.0$ centered at $(x, y) = (0, 0.75)$. The simulation of the wetting is performed using a mesh consisting of regular triangles with 406×203 nodes and a time step $\Delta t = 10^{-4}$. In the reinitialization described in equation (12) the viscosity parameter in the normal direction is set to $\varepsilon_n = 3h$, where h is the mesh size, the viscosity parameter in the tangential direction is set to $\varepsilon_\tau = 3\varepsilon_n$, and the time step $\Delta\hat{t} = \Delta t$. In equation (11) giving the regularized normal vector field, the regularization parameter $\gamma = 2.5\varepsilon_n$.

In Fig. 17 the wetting on the solid wall is illustrated for nine different snapshots. The liquid–liquid interface starts to wet the surface because of the difference in the initial and the static contact angle. During the transient phase it passes the equilibrium position but then retracts and oscillates toward equilibrium. The snapshots can be compared with the snapshots from the volume–of–fluid computation in [8]. Note that, contrary to [8], the snapshots are in non–dimensional time. In Fig. 18, the spreading diameter, i.e. the distance between the two contact points, is plotted as a function of time. Note that this droplet oscillates toward equilibrium. This is because of the resistance of the drop to change its motion.

5 Discussion

We have developed and presented a conservative level set model for two–phase flow, which can capture contact line dynamics using no–slip boundary conditions for the velocity. In this model, which is an extension of the model presented in [11], the contact angle at equilibrium is used to induce a movement of the contact point by diffusion. In our model, a diffusion parameter and a regularization parameter are related to the size of the contact point region. Since the model is formulated in terms of partial differential equations on conservative form, conservation properties are retained by using standard conservative numerical methods.

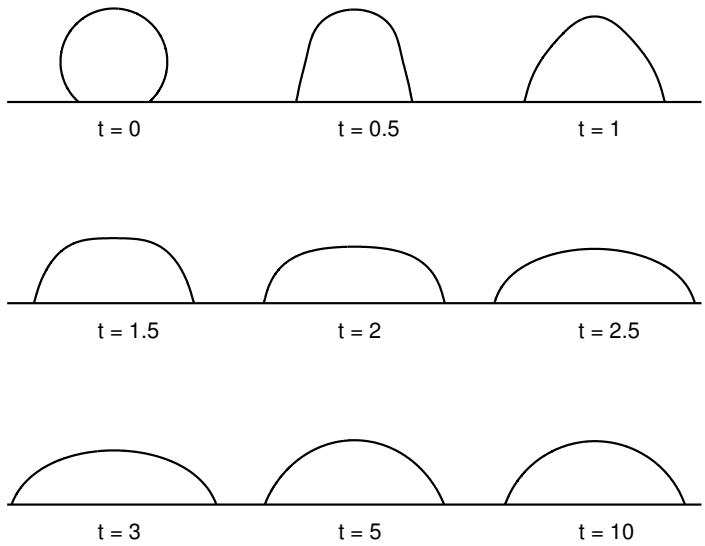


Figure 17: Wetting of a liquid drop on a solid surface. The interface separating the two fluids at dimensionless times $t = 0,\ 0.5,\ 1,\ 1.5,\ 2,\ 2.5,\ 3,\ 5,\ 10$ with $Re = 20$, $Ca = 0.03$. The static contact angle $\alpha_s = 70.53$.

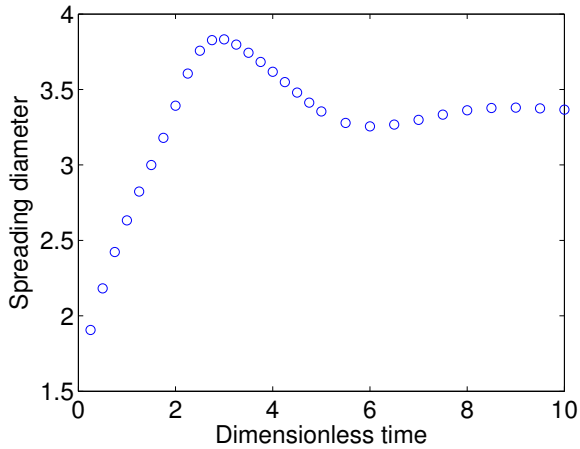


Figure 18: Spreading diameter of the droplet in Fig. 17 as a function of dimensionless time. The drop oscillates towards equilibrium.

Numerical computations for two different applications demonstrate that important features of contact line dynamics are captured. The results from the calculations of the droplets wetting walls have shown good agreement with other numerical methods and available theory. Parameter studies have indicated convergence as the model parameters approach their limit values. For cases when the mismatch in angle is moderate the method yields good results. For a fixed foot size, larger differences in angles would result in higher curvature and require a finer mesh in the foot region, making adaptive mesh refinement necessary. The refinement of the mesh could be based on the curvature.

In this work we have prescribed the static contact angle and have aimed to obtain an angle close to the static angle. In the first droplet on a flat plate calculation in Section 4.3, the difference between the initial and the prescribed static contact angle was large. Therefore, we had to choose between a large foot with the correct angle at the wall and a small foot but a larger discrepancy between the obtained angle and the prescribed static angle. We obtained much better results with a small foot region. The qualitative behavior was correct even though the small foot region resulted in a larger discrepancy between the obtained angle and the prescribed static angle. An important question is if the contact angle in computations has to be exactly equal to the static value. If it has to be equal, more tangential diffusion is needed. However, we do not want too much diffusion away from the foot region. Therefore, it could be beneficial to let the tangential diffusion vary with the distance from the contact point. We have performed numerical simulations with a tangential viscosity that

decays exponentially with the distance from the boundary, and we have obtained promising results. This approach needs to be further investigated.

We have found that our method works well also in cases when the flow is not dominated by capillary forces, even without high resolution. For such cases it is not important that the interface attains the static angle at all times, only that the reinitialization procedure provides a mechanism for contact line movement.

Acknowledgements

The authors acknowledge Gustav Amberg, Minh Do–Quang, and Emanuel Rubensson for helpful discussions on fluid dynamics, Femlego and numerics. This project has partially been financed by Swedish Foundation for Strategic Research (SSF), Center for Industrial and applied mathematics at KTH (CIAM) and Linné Flow center at KTH.

References

[1] R. M. German, Liquid Phase Sintering, Springer, 1985.

[2] P. G. de Gennes, Wetting: statics and dynamics, Rev. Mod. Phys 57 (1985) 827–863.

[3] T. Qian, Z.-P. Wang, P. Sheng, Molecular hydrodynamics of the moving contact line in two–phase immiscible flows, Comm. Comput. Phys. 1 (1) (2006) 1–52.

[4] L. M. Hocking, A moving fluid interface on a rough surface, J. Comput. Phys. 76 (1976) 801–817.

[5] P. Sheng, M. Zhou, Immiscible–fluid displacement: Contact–line dynamics and the velocity–dependent capillary pressure, Phys. Rev. A 45 (1992) 5694.

[6] D. Jacqmin, Contact-line dynamics of a diffuse fluid interface, J. Fluid Mech. 402 (2000) 57–88.

[7] W. Villanueva, G. Amberg, Some generic capillary-driven flows, Int. J. Multiphase Flow 32 (2006) 1072–1086.

[8] M. Renardy, Y. Renardy, J. Li, Numerical simulation of moving contact line problems using a volume–of–fluid method, J. Comput. Phys. 171 (2001) 243–263.

[9] H. Liu, S. Krishnan, S. Marella, H. Udaykumar, Sharp interface cartesian grid method ii: A technique for simulating droplet interactions with surfaces of arbitrary shape, J. Comput. Phys. 210 (2005) 32–54.

[10] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, J. Comput. Phys. 210 (2005) 225–246.

[11] E. Olsson, G. Kreiss, S. Zahedi, A conservative level set method for two phase flow ii, J. Comput. Phys. 225 (2007) 785–807.

[12] J. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, 1999.

[13] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer-Verlag, 2003.

[14] A. Harten, The artificial compression method for computation of shocks and contact discontinuities.i.single conservation laws, Comm. Pure Appl. Math. (1977) 611–638.

[15] J.-L. Guermond, L. Quartapelle, Calculation of incompressible viscous flows by an unconditionally stable projection FEM, J. Comput. Phys 132 (1997) 12–33.

[16] J. U. Brackbill, D. Kothe, C. Zemach, A continuum method for modeling surface tension, J. Comput. Phys 100 (1992) 335–353.

[17] A.-K. Tornberg, B. Engquist, A finite element based level set method for multiphase flow applications, Computing and Visualization in Science 3 (2000) 93–101.

[18] M. Do-Quang, W. Villanueva, G. Amberg, I. Loginova, Parallel adaptive computation of some time–dependent materials–related microstructural problems, Bull. Polish Acad. Sci. 55 (2007) 229–237.

[19] R. L.Hoffman, A study of the advancing interface, J. Colloid Interface Sci. 50 (1975) 228–241.

[20] R. G. Cox, The dynamics of the spreading of liquids on a solid surface. part1, J. Fluid Mech. 168 (1986) 169–194.

Paper 2

An interface capturing method for two–phase flow with moving contact lines

Sara Zahedi, Gunilla Kreiss, and Katarina Gustavsson

Proceedings of the 1st European Conference on Microfluidics, Bologna, 2008.

μFLU08–118

AN INTERFACE CAPTURING METHOD FOR TWO–PHASE FLOW WITH MOVING CONTACT LINES

Sara Zahedi[1]
Royal Institute of Technology, CSC/NA, 100 44 Stockholm, Sweden
sara7@csc.kth.se

Gunilla Kreiss, Katarina Gustavsson
Uppsala University, 751 05 Uppsala, Sweden,
Royal Institute of Technology, CSC/NA, 100 44 Stockholm, Sweden
gunillak@it.uu.se, katarina@csc.kth.se

KEY WORDS

contact line dynamics, contact angle, level set method, two phase flow, conservative method

ABSTRACT

This paper presents a conservative numerical method capable of capturing contact line movement for the simulation of capillary dominated flow of two immiscible, incompressible fluids. The interface between the two fluids is represented implicitly by the 0.5–level set of a function varying smoothly from zero to one. The flow of each phase is governed by the incompressible Navier–Stokes equations with no–slip boundary condition at solid walls. The key idea is to drive the contact line movement by prescribing the normal vector at the boundary to be the static normal vector corresponding to the equilibrium contact angle. Our model is formulated as a system of partial differential equations and takes into account differences in density and viscosity, gravity, and surface tension effects. We discretize using the finite element method which allows for use of the weak form of the equations. The surface tension forces are included as line integrals along the interfaces. We eliminate the second order derivatives in these forces, coming from the curvature term, by applying partial integration of the Laplace–Beltrami operator. Numerical experiments of capillary dominated flows in two space dimensions are presented and include tests of convergence rates and comparison with other numerical methods. Our numerical scheme show good accuracy and our results compare well with available theory.

1. INTRODUCTION

Accurate modeling and simulations of contact line movement is of crucial interest in simulations of 1) two–phase flow in microfluidic devices, such as pumps, valves, separators etc, 2) inkjet printing, spray coating, and spray cooling, and 3) two–(or multi–)phase flow in pore structures occurring for example when oil is recovered by water injection and in liquid phase sintering. In all these cases simulations can contribute to a better understanding of the process considered.

Contact line dynamics is the movement of the intersection line between the interface of two immiscible fluids and a solid boundary. An example of such a phenomenon is when a liquid drop is spreading over a solid surface. As the drop spreads, the contact line will move until it reaches an equilibrium state determined by the surface energies of the interfaces involved. The angle between the contact line at equilibrium and the boundary is often referred to as the equilibrium or static contact angle (hereinafter, *the static contact angle*). For a more detailed discussion of this kind of phenomena see [1].

[1] Corresponding author

The immiscible and incompressible flow is described by the incompressible Navier–Stokes equations with the surface tension forces and gravity forces added as source terms. Some representation of the interface separating the two fluids is required, for example a level set function or markers, and an evolution equation for the advection of the interface. At all interior points, the physically correct model is that the interface is advected by the fluid velocity. At contact lines, however, the standard boundary condition for the velocity is no–slip, which means that the contact line cannot move. This is unphysical in many cases, which is reflected by the shear stress becoming singular. Over the years, various techniques to overcome this difficulty have been introduced. One direction is to introduce a so called slip length, which permits the contact line to move, see for instance [2] and references therein. The difficulty with this approach is to provide an appropriate value for the slip coefficient. The existence of fluid–wall slipping has been observed in molecular dynamics simulations, see [3] and references therein. Another approach was suggested by Jacqmin [4] who used a coupled Cahn–Hilliard/ Navier–Stokes formulation consisting of the Navier–Stokes equations coupled to a system of two parabolic equations for the chemical potential and the concentration of one of the fluids. In this model, the interface is modeled as a layer with a continuous transition from one fluid to the other, and the contact line as a small region; the part of the interface layer close to the solid boundary. The contact line moves by diffusive processes on a fast time scale in the small region at the boundary so that the contact angle is adjusted to the static value. The result is a region close to the boundary with high curvature of the interface, and consequently a strong surface tension force. The fact that the interface can move by diffusive processes eliminates the need for modeling fluid slip. A drawback with the model suggested by Jacqmin is that the interface must be highly resolved to achieve accuracy. In [5] this formulation was used to investigate basic wetting phenomena.

A new level set method for two phase flow with good mass conservation of each fluid was introduced in [6] and [7]. In this method the interface is represented implicitly by the 0.5–level set of a regularized Heaviside function ϕ. By construction, the method conserves $\int \phi$. We have extended this method to be able to capture contact–line movement while retaining the conservative property [8, 9]. The new model mimics the Cahn–Hilliard/Navier–Stokes behavior of the interface at solid walls by using diffusion to move the contact point. Compared to the standard Cahn–Hilliard/Navier–Stokes formulation we believe the new model reduces the needed computational effort, since a relatively coarse grid is sufficient to capture the dynamics, at least away from contact points. In this work we use an improved representation of the surface tension, based on partial integration of the Laplace–Beltrami operator, see [10, 11] and references therein. In this way we avoid computation of the curvature. We also present new computations, where viscosity and density of the fluids differ. All our computations are in two space dimensions, but there is no principal difficulty to extend the method to three space dimensions.

The paper is organized as follows. In section 2 we present the governing equations of the model and give a description of our interface representation and the numerical method. The treatment of the surface tension force based on partial integration of the Laplace–Beltrami operator is explained. In section 3, we use the model in three different applications; the spreading of a droplet on a flat plate, the rising of a droplet and a channel flow. Section 4 contains a summary and a discussion of possible improvements of the model.

2. MATHEMATICAL MODEL AND NUMERICAL METHOD

The fundamental model of two–phase flow with contact line dynamics consists of the incompressible Navier–Stokes equations coupled with evolution of the interface. Evolution of the interface is, in this work, achieved by use of the advection equation together with a reinitialization step. In this section we will present this model and how it is treated numerically.

2.1 The Incompressible Navier–Stokes Equations

Assume a given domain Ω is occupied by two immiscible fluids separated by an interface Γ. The equations describing this immiscible flow are the incompressible Navier–Stokes equations with the contribution of the surface tension forces and the gravity forces added as source terms:

$$Re\left((\rho\mathbf{u})_t + \nabla\cdot(\rho\mathbf{u}\mathbf{u})\right) = -Re\nabla p + \nabla\cdot\left(\mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)\right) + \frac{Re}{Fr^2}\rho\mathbf{e}_g + \frac{1}{Ca}\kappa\mathbf{n}\delta_\Gamma, \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0.$$

Here $\mathbf{u}$, p, ρ and μ denote velocity, pressure, density, and viscosity, respectively. In general ρ and μ are discontinuous across the interface separating the two fluids. The curvature and normal of the interface Γ are denoted by κ and $\mathbf{n}$, and δ_Γ is a Dirac delta function with support on Γ. Its action on any smooth test function v is given by

$$\int_\Omega \delta_\Gamma v d\Omega = \int_\Gamma v d\Gamma. \tag{2}$$

The direction of gravitation is denoted by $\mathbf{e}_g$. The equations are formulated on a non–dimensional form where the dimensionless Reynolds, Capillary and Froude numbers are given by

$$Re = \frac{\rho_{\text{ref}} u_{\text{ref}} l_{\text{ref}}}{\mu_{\text{ref}}},\ Ca = \frac{\mu_{\text{ref}} u_{\text{ref}}}{\sigma},\ \text{and}\ Fr = \frac{u_{\text{ref}}}{\sqrt{l_{\text{ref}} g}}. \tag{3}$$

Here ρ_{ref}, μ_{ref}, l_{ref}, and u_{ref} are constant reference density, viscosity, length and velocity. The coefficient σ is the surface tension coefficient and g is the gravitational constant. The Reynolds number is the ratio between inertial and viscous forces. The Capillary number gives the ratio between the viscous and surface tension forces and the Froude number gives a measure of the ratio between inertial and gravitational forces.

In order to complete the formulation, a representation of the interface and a model for its motion are needed. This will be addressed in the next subsection.

2.2 An Improved Level set Representation of the Interface

Level set representations of the interface in two phase flow have been used in many flow situations, see for instance [12] or [13]. The basic idea is that if a level set of a function ϕ_0 defines the interface at $t = 0$, the same level set of the solution to

$$\phi_t + \nabla \cdot (\phi \mathbf{u}) = 0,\ \phi(\cdot, 0) = \phi_0, \tag{4}$$

defines the interface at later times.

The method proposed in this paper is an extension of the conservative level set method, introduced in [6]. Instead of the signed distance function [12, 13] usually used to define the interface, this method uses a regularized indicator function ϕ that takes the values 0 and 1 in the two fluids, respectively. The 0.5–level defines the interface. In this way, good conservation properties can be achieved simply by using a conservative discretization. The shape of the regularized step function is controlled in a reinitialization step. This step is modeled by a partial differential equation where a non–linear term, resembling a compressive limiter (see [14]), is balanced by diffusion in the normal direction. In the following we introduce a new reinitialization that can be used to capture contact line dynamics. This is a generalization of the reinitialization procedures used in [6] and [7].

The new model mimics the Cahn–Hilliard/Navier–Stokes model in the following way. Contact points are modeled as regions and diffusion is used to move the contact point so that the angle between the interface, defined by the 0.5–level set of ϕ, and the boundary always equals the static contact angle α_s. The static contact angle corresponds to a normal to the interface $\mathbf{n}_{\alpha_s}$. Since the gradient of the level set function is related to the normal of the interface by $\mathbf{n} = \nabla\phi/|\nabla\phi|$, the condition on the angle can be formulated as a condition on the gradient. We cannot, however, directly prescribe the gradient of the level set function at the boundary without creating a flux of ϕ over the boundary that would destroy the conservative properties of the model.

Our idea is to instead use a regularized normal vector field satisfying

$$\mathbf{n} - \nabla \cdot (\gamma^2 \nabla \mathbf{n}) = \frac{\nabla\phi}{|\nabla\phi|}, \tag{5}$$

with Dirichlet boundary conditions, $\mathbf{n} = \mathbf{n}_{\alpha_s}$ along solid walls. The regularization parameter γ should preferably be chosen $\gamma \ll L$, where L is the length of typical features in the problem at hand. The regularized normal vector will then be close to the gradient of the unperturbed ϕ, except along the boundary, where a boundary

layer with thickness proportional to γ will form. The modified vector field is then used in the reinitialization in the following fashion

$$\phi_{\hat{t}} + \nabla \cdot (\phi(1-\phi)\mathbf{n}) - \nabla \cdot (\varepsilon_n(\nabla\phi \cdot \mathbf{n})\mathbf{n}) - \nabla \cdot (\varepsilon_\tau(\nabla\phi \cdot \mathbf{t})\mathbf{t}) = 0 \tag{6}$$

where $\mathbf{n}$ is the normalized and regularized gradient of ϕ, satisfying (5), and $\mathbf{t}$ is the tangent, orthogonal to $\mathbf{n}$, ε_n is a viscosity parameter in the normal direction and ε_τ is a viscosity parameter in the tangential direction. The second term in (6) represents compression in the normal direction, while the third models diffusion in the normal direction and the last term models diffusion in the tangential direction. In the original work on the conservative level set method [6] the standard isotropic diffusion was used. This corresponds to choosing $\varepsilon_\tau = \varepsilon_n = \varepsilon$ in equation (6). In the subsequent work, presented in [7], $\varepsilon_\tau = 0$. The idea was to avoid unnecessary movement of the 0.5–level set in the tangential direction. However, the tangential diffusion is essential for the contact angle to achieve the desired static value. An alternative is to include diffusion in the advection equation (4).

If our modeling is successful, the solution ϕ will exhibit a boundary layer of thickness proportional to the regularization parameter γ. The angle of the interface (the 0.5–level set of ϕ) to the boundary changes from the static contact angle α_s to the apparent contact angle α_a. The apparent contact angle is the angle formed between the boundary and the interface away from the contact point region. Thus, close to the boundary the curvature will be large. It is important that γ is chosen so that the curvature in this region is numerically resolved. The contact point will move with the same speed as the tangential fluid velocity just outside the contact point region. When the two fluids have the same viscosity it can be shown that for small γ the speed is independent of γ, see [9].

2.3 Numerical Treatment

We solve the Navier–Stokes equations using a projection method with an added pressure stabilization term [7]. This method was introduced by Guermond and Quartapelle [15]. The surface tension effect is treated as a force, $\mathbf{f} = \sigma\kappa\mathbf{n}\delta_\Gamma$ added to the Navier–Stokes equations. Since we consider applications where the capillary effects dominate the accuracy of the surface tension force is essential. In general, the accuracy of the curvature estimation and the implementation of the surface tension force are crucial in order to avoid large unphysical oscillations of the velocity close to the interface, so–called spurious velocities. Using the finite element framework enables us to write the equations in weak form. The weak formulation of the equations does not include δ_Γ and therefore no explicit discretization of the delta function is required. By using the Laplace–Beltrami characterization of the curvature we do not need to calculate the curvature.

Define the tangential derivative of a function g along Γ by $\nabla_\Gamma g = P\nabla g$ where P is the orthogonal projection:

$$P = I - \mathbf{n}\mathbf{n}^T. \tag{7}$$

Replace the term $\kappa\mathbf{n}$ by the Laplace–Beltrami operator

$$\nabla_\Gamma \cdot (\nabla_\Gamma id_\Gamma) = \kappa\mathbf{n}, \tag{8}$$

where id_Γ is the identity on Γ and the normal vector $\mathbf{n}$ is calculated as

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}. \tag{9}$$

If $\mathbf{v}$ is a smooth vector valued function on Ω we can integrate by parts to get

$$\begin{aligned} f_\Gamma(\mathbf{v}) &= \frac{1}{Ca}\int_\Omega \kappa\mathbf{n}\cdot\mathbf{v}\delta_\Gamma d\Omega = \frac{1}{Ca}\int_\Gamma \kappa\mathbf{n}\cdot\mathbf{v}d\Gamma = \\ &= \frac{1}{Ca}\int_\Gamma \nabla_\Gamma\cdot(\nabla_\Gamma id_\Gamma)\cdot\mathbf{v}d\Gamma = -\frac{1}{Ca}\int_\Gamma \nabla_\Gamma id_\Gamma\cdot\nabla_\Gamma\mathbf{v}d\Gamma \end{aligned} \tag{10}$$

In our model the boundary conditions for the velocity is no–slip. Hence our test function $\mathbf{v}$ is zero at the contact points and we don't have any boundary terms in the above identity. This technique of reducing the order of differentiation associated with the curvature term has also successfully been applied in the finite element context in [10, 11] and references therein.

At contact points diffusion of ϕ tangential to the interface is essential. In the third application, "Capillary Dominated Channel Flow", we choose to add the diffusion in the reinitialization equation (6) by letting $\varepsilon_\tau = \varepsilon_n$. In the first and the second application, "Droplet on a Flat Plate" and "Rising Droplet", we let $\varepsilon_\tau = 0$ but we add an isotropic diffusion term equal to $5h^2 = 0.0011$ in the advection equation (4). In each time step the reinitialization equation (6) is solved to steady state. As in [7] we use a second order semi–implicit discretization in time with $\Delta\hat{t} = \Delta t$. Typically two steps of reinitialization is performed at each time step.

The numerical simulations were carried out using FemLego (Amberg et al. [16]), a symbolic tool to solve partial differential equations with the finite element method. All equations were discretized in space using piecewise linear functions and the linear systems were solved using conjugate gradient methods.

3. APPLICATIONS

In this section we discuss the numerical treatment and results for three applications in which capillary effects dominate: a droplet on a flat plate, rising droplet and channel flow. All applications involve two immiscible fluids and moving contact lines. The proposed model is used to simulate the contact line dynamics. The results in the application "Droplet on a Flat Plate" is compared with theory and other computations. The two other applications indicate that we can also handle density and viscosity differences.

3.1 Application 1: Droplet on a Flat Plate

We consider here a two–dimensional droplet with diameter l_{ref}, density ρ_{ref}, and viscosity μ_{ref}, lying on a solid surface surrounded by another liquid with the same density and viscosity. The static contact angle is $\alpha_s = 25^\circ$. Capillary effects dominate and gravity is neglected. We will simulate how this droplet wets the surface. This wetting phenomenon was previously studied using a phase field method by Villanueva and Amberg [5]. We will compare our results with data from the phase field simulation and Cox's theory [17].

3.1.1 Computations

The computational domain is in non–dimensional coordinates $\{(x, y) : -2 \leq x \leq 2,\ 0 \leq y \leq 2\}$ and the initial drop is symmetric around $x = 0$ and has non–dimensional radius $r = 0.5$. At initial time the drop is in contact with the solid wall at an angle of 156°. The Reynolds number $Re = \frac{\rho_{\text{ref}} u_{\text{ref}} l_{\text{ref}}}{\mu_{\text{ref}}} = 1$ and the Capillary number $Ca = \frac{2\sqrt{2}\mu_{\text{ref}} u_{\text{ref}}}{3\sigma} = 1$. The simulation of the wetting is performed using a mesh consisting of regular triangles with 270×135 nodes and a time step $\Delta t = 0.0011$. The liquid–liquid interface (the 0.5–level set of ϕ) is advected by solving the advection equation (4) with a diffusion term equal to $5h^2 = 0.0011$ where $h = \Delta x = \Delta y$ and typically two steps of reinitialization. In the reinitialization described in equation (6) the viscosity parameter in the normal direction is set to $\varepsilon_n = 4.5h = 0.0667$ and the viscosity parameter in the tangential direction is set to $\varepsilon_\tau = 0$. The boundary condition for the velocity is no–slip. For the regularized normal vector field, equation (5) the boundary condition is

$$\mathbf{n}|_{y=0} = (-\text{sign}(x)0.4226, -0.9063)$$

so that the contact angle at the wall is equal to the static contact angle 25°. At the other boundaries we use homogeneous Neumann conditions. The regularization parameter in equation (5) $\gamma = 0.1667$. We want to choose γ so that the normal vector varies smoothly from the prescribed equilibrium value at the boundary to the actual value. Thus on one hand, a large γ is needed when the dynamic contact angle is large compared to the static contact angle. The static contact angle determines the value of the normal vector at the boundary. On the other hand, if γ is too large, we get too large wetting speed, especially in the initial steps. This was studied in [9]. When γ is decreased, the grid has to be refined. We believe that compared to [9] we should be able to use a smaller γ since we don't need to calculate the curvature. This has however not yet been tested.

In Fig. 1 the wetting on the solid wall is illustrated for six different snapshots. The liquid–liquid interface immediately forms a foot due to the boundary condition for the normal at the wall described above. The strong curvature in the foot region causes the contact line to move and the drop starts to wet the surface. The contact–line movement is fast in the initial steps but slows down as the drop approaches equilibrium. At $t \approx 300$ equilibrium is reached.

We measure the dynamic contact angle that the liquid–liquid interface makes with the solid surface at the inflexion point of the foot as illustrated in Fig. 2(a). The wetting speed is given by dividing the difference in position of the intersection point (see Fig. 2(a)) at two successive times by the time difference. The dynamic contact angle initially decreases rapidly with time but the decrease slows down as the angle approaches the equilibrium value. In Fig. 2 the dependence of the dynamic contact angle on the Capillary number $Ca^* = \frac{2\sqrt{2}\mu_{\text{ref}}u}{3\sigma} = \frac{u}{u_{\text{ref}}}$, which in this case is equal to the wetting speed, is shown. The results are compared with a phase field simulation[5] and Cox's theory[17]. Our simulation shows qualitative agreement with them all. The velocity for each given angle is, in our simulation, somewhat larger than the predictions provided by Cox's theory with the viscosity ratio $\lambda = 1$ and $\delta = 10^{-4}$. See [9] for the choice of δ.

To investigate convergence we have performed computations for three different meshes with $h_1 = 0.0222$, $h_2 = \frac{2}{3}h_1$ and $h_3 = \frac{4}{9}h_1$. All parameters are fixed, the initial contact angle is now 90° and $\gamma = 0.1$. The error is measured as the sum of absolute differences in the y–coordinates for the interface for 100 x–coordinates divided by the number of points. The order of convergence is shown in Fig. 3 both when applying integration by parts of the Laplace–Beltrami operator on the interface and calculating the curvature and then the line integral.

3.2 Application 2: Rising Droplet

We now consider a two–dimensional droplet with density ρ_D and viscosity μ_D lying on a solid surface at an angle of 90° surrounded by a heavier fluid with density $\rho_L = 10\rho_D$ and viscosity $\mu_L = \mu_D$. We make a simulation of the droplet rising.

3.2.1 Computations

The computational domain is in non–dimensional coordinates $\{(x, y) : -1.5 \leq x \leq 1.5,\ 0 \leq y \leq 3\}$. The initial drop is symmetric around $x = 0$ with non–dimensional radius $r = 0.5$. The Reynolds number and the Capillary number are as in the first application "Droplet on a Flat Plate" set to 1 but the gravity cannot be neglected and the Froude number $Fr = \frac{u_{\text{ref}}}{\sqrt{l_{\text{ref}}g}} = 0.7$. The simulation is performed using a mesh consisting of regular triangles with 180×180 nodes and a time step $\Delta t = 0.0011$. The regularization parameter in equation (5) $\gamma = 0.1$. All other parameters are as in the first application "Droplet on a Flat Plate". In Fig. 4 the interface and the velocities at the interface are plotted at a sequence of different times. The drop wants to wet the solid surface but the surrounding fluid is heavier so it starts to rise. At $t \approx 9$ the droplet breaks into two parts, where one part continues to rise and one part remains attached to the lower wall.

3.3 Application 3: Capillary Dominated Channel Flow

Consider two fluids in a two dimensional channel of non–dimensional width 1 and length 1.5, initially separated by an interface, normal to the channel, without curvature. Thus, the initial apparent contact angle is $\alpha_a = 90^\circ$. We assume that the static contact angle α_s is less than 90°, which corresponds to the left fluid wetting the channel walls. Clearly the initial condition is not a steady state solution. In the absence of an outer pressure gradient or a prescribed flux, the only driving force are the capillary effect at the channel walls, which will move the interface to the right. If the densities and viscosities of the two fluids are of similar size we expect that after an initial, transient process, the interface will move at a constant, or in the case of differing fluid properties, slowly varying speed U. We expect the interface to consist of an interior part with curvature $\kappa_a = 2\cos(\alpha_a)$, and two small sections close to the walls with curvature $\kappa \propto \frac{f(\alpha_a - \alpha_s)}{\gamma}$, where f is some smooth function.

We have performed two sets of computations to investigate the qualitative behavior of our model. In the computations in this section the Capillary number and the Reynolds number are set to unity, and both fluids have the same density. At the channel walls we prescribe zero velocity, i.e. no slip, and the flux of the level set function is set to zero. The normal vector is set so that the contact angle at the wall equals the static contact angle. At the inlet and outlet we use vanishing tangential velocity, vanishing normal derivative of the normal velocity, and Dirichlet conditions for the level set function and the pressure. Throughout this section we use $\varepsilon_\tau = \varepsilon_n = 0.025$, $\gamma = 0.05$, $\Delta x = 0.01$ and $\Delta\hat{t} = \Delta t = 0.04$.

In the first set of computations we investigate the effect of the static contact angle on the interface speed. In these computations the fluids have the same viscosity. In Fig. 5 we show the interface together with local velocity vectors during the initial, transient phase. A region of large curvature of the interface develops near

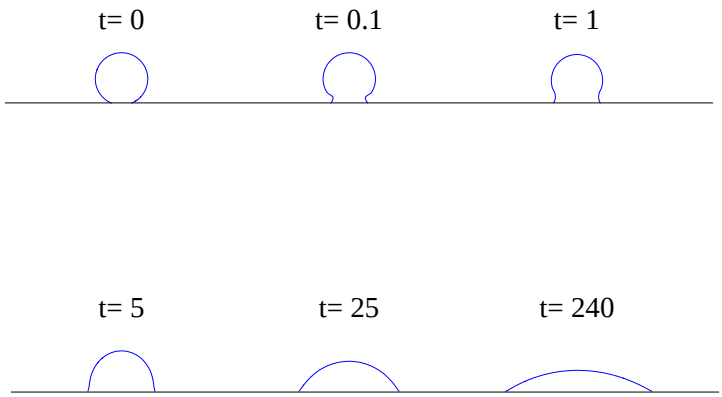
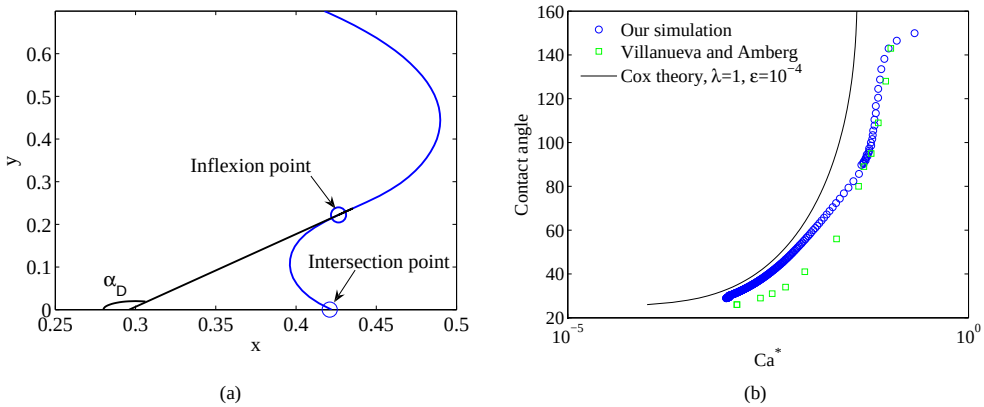


Figure 1: Wetting of a liquid drop on a solid surface. The interface separating the two fluids at times $t = 0, 0.1, 1, 5, 25, 240$ with $Re = 1$, $Ca = 1$. The contact angles at the different times are $\alpha \approx 156, 149, 117, 82, 54, 29$. The regularization parameter $\gamma = 0.1667$.



(a) (b)

Figure 2: (a): The dynamic contact angle is measured by considering the tangent at the inflexion point. The wetting speed is given from the position of the intersection point at different times. (b) : The dynamic contact angle versus the capillary number. Initial angle is $156°$.

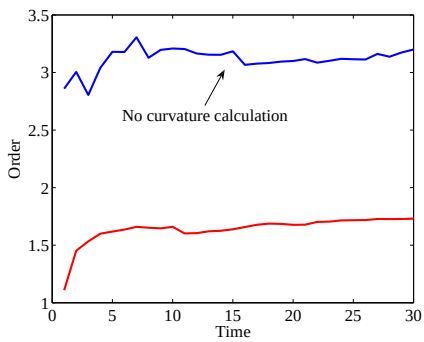
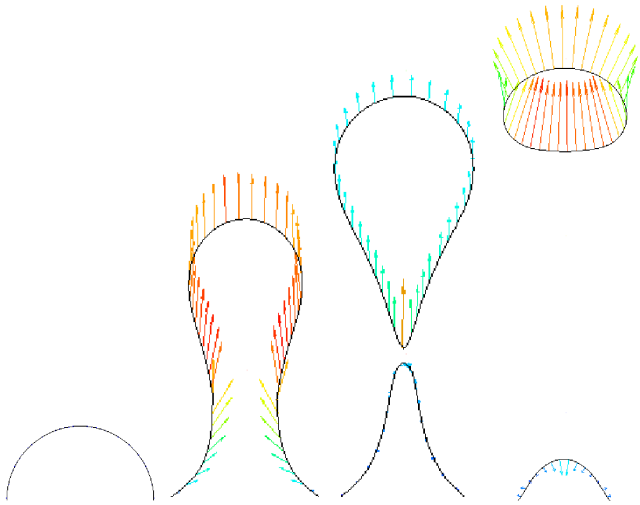


Figure 3: The order of convergence both when applying integration by parts of the Laplace–Beltrami operator on the interface and calculating the curvature and then the line integral. All parameters are fixed, the initial contact angle is $90°$ and $\gamma = 0.1$.



(a) $t = 0$ (b) $t = 7$ (c) $t = 9$ (d) $t = 13$

Figure 4: Rising drop. The location of the interface separating the two fluids at times $t = 0, 7, 9, 13$ with Re=1, Ca=1, Fr=0.7, and $\frac{\rho_L}{\rho_D} = 10$. The arrows show velocities at the interface.

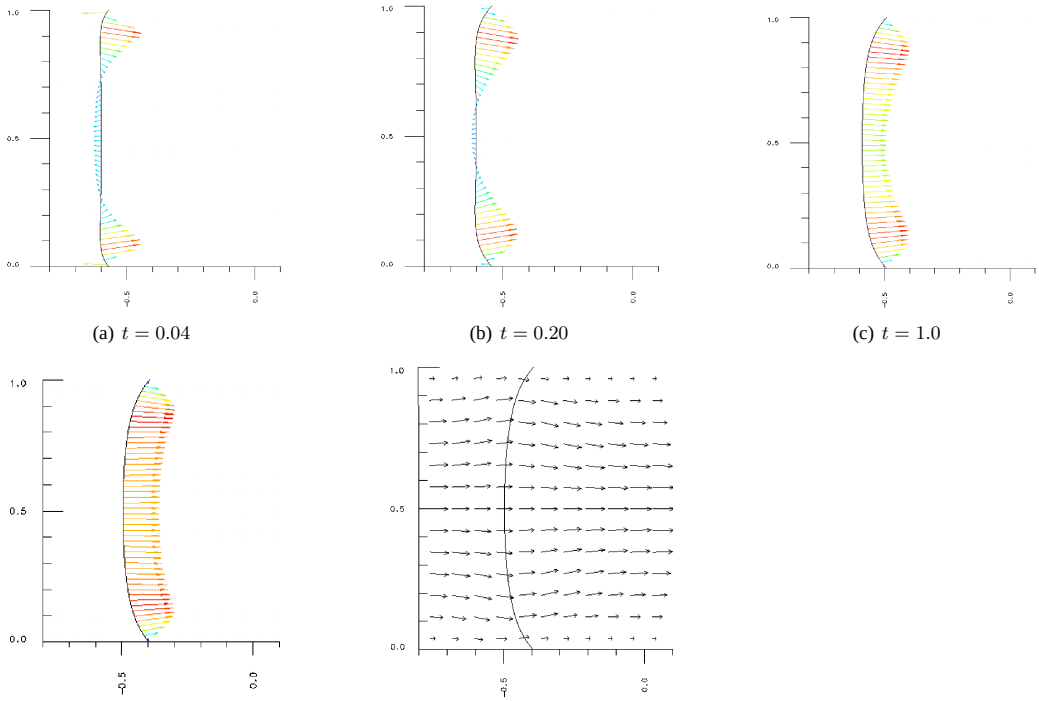
the solid walls. There, the surface tension is strong, resulting in local, large velocities. After $t = 3$ the transient development of the interface is complete, but the curvature and velocities are still large near the wall. Away from the interface the velocity field is close to the standard pipe flow parabola. In Fig. 6 we see the time history of the interface for two different static angles. In both cases an interface shape develops within the first time units, and then propagates with constant speed. As expected, the speed is greater when the static angle is smaller. In the second set of computations we investigate the effect of a difference in fluid viscosity on the interface movement. We let the fluid to the left be more viscous than the fluid to the right. As the interface moves to the right an increasing part of the channel is filled with the more viscous fluid. Thus we expect the interface speed to slow down. In Fig. 7 we see the time history of the interface at an early stage and at a later stage for two different viscosity ratios. It is clear that increasing the difference in viscosity increases the slowing down effect, as expected.

4. CONCLUSIONS

We have developed and presented a conservative level set model for two–phase flow, which can capture contact line dynamics using no–slip boundary conditions for the velocity. In this model, the contact angle at equilibrium is used to induce a movement by diffusion of the contact point. Since the model is formulated in terms of partial differential equations on conservative form, conservation properties are retained by using standard conservative numerical methods.

In our model we have a diffusion parameter and a regularization parameter related to the size of the contact point region. An investigation of where the needed diffusion is best to add, in the advection equation or in the reinitialization, as well as the magnitude of the diffusion term needs to be done. By writing the equations in weak form and using the Laplace–Beltrami operator we have been able to avoid calculation of the curvature. We believe this will enable us to choose a smaller regularization parameter but this remains to be studied. As a further improvement of this model we suggest that the model parameters as well as the new boundary condition should be adaptive and act only locally in small regions close to contact points. Also, adaptive refinement of the mesh in regions with large curvature would be beneficial in order to be able to use smaller values of the parameters. This is especially important for the regularization parameter γ.

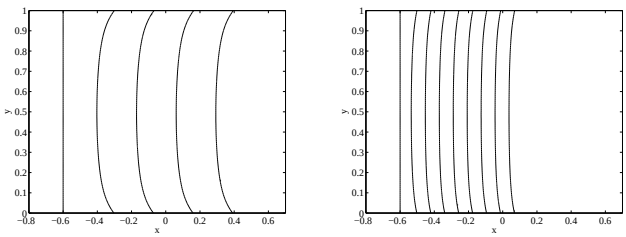
Numerical computations for three different applications demonstrate that qualitative features of contact line dynamics are captured. The results from the calculation of the droplet wetting the wall has shown good



(a) $t = 0.04$ (b) $t = 0.20$ (c) $t = 1.0$

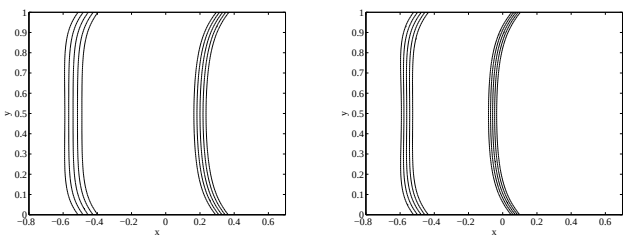
(d) $t = 3.0$ (e) $t = 3.0$

Figure 5: The location of the interface (the 0.5–level set of ϕ) at $t = 0.04, 0.2, 1.0, 3.0$. The arrows show the velocities. The angles at the upper and the lower channel walls are $\alpha = 45°$. The viscosity ratio $\frac{\mu_1}{\mu_2} = 1$.



(a) $\alpha_s = 45°$ (b) $\alpha_{s_1} = 75°$

Figure 6: The location of the interface (the 0.5–level set of ϕ) at $t = 0, 5, 10, 15$, and 20 for two different static contact angles. The viscosity ratio $\frac{\mu_1}{\mu_2} = 1$.



(a) $\frac{\mu_1}{\mu_2} = 4$ (b) $\frac{\mu_1}{\mu_2} = 8$

Figure 7: The location of the interface (the 0.5–level set of ϕ) at $t = 1, 2, 3, 4, 5, 36, 37, 38, 39, 40$. The angles at the upper and the lower channel walls are $\alpha_{s_1} = \alpha_{s_2} = 45°$. The fluid to the left is more viscous than the fluid to the right.

agreement with available theory and we observed better convergence compared to the calculations done in [9]. However, comparison with experiments and a better understanding of how the diffusion parameter and the regularization parameter should be chosen in order to resolve the curvature is needed before the model can be used for qualitative predictions.

ACKNOWLEDGEMENTS

The authors acknowledge Gustav Amberg, Minh Do–Quang, Anna–Karin Tornberg and Emanuel Rubensson for helpful discussions on fluid dynamics, Femlego and numerics. This project has partially been financed by Swedish Foundation for Strategic Research (SSF), Center for Industrial and applied mathematics at KTH (CIAM) and Linne Flow center at KTH.

REFERENCES

[1] P. G. de Gennes (1985). Wetting: statics and dynamics.*Rev. Mod. Phys.* **57**, 827–863.
[2] M. Renardy, Y. Renardy, J. Li (2001). Numerical simulation of moving contact line problems using a volume–of–fluid method. *J. Comput. Phys.* **171**, 243–263.
[3] T. Qian, Z.-P. Wang, P. Sheng (2006). Molecular hydrodynamics of the moving contact line in two–phase immiscible flows. *Comm. Comput. Phys.* **1**, 1–52.
[4] D. Jacqmin (2000). Contact-line dynamics of a diffuse fluid interface. *J. Fluid Mech.* **402**, 57–88.
[5] W. Villanueva, G. Amberg (2006). Some generic capillary-driven flows. *Int. J. Multiphase Flow* **32**, 1072–1086.
[6] E. Olsson, G. Kreiss (2005). A conservative level set method for two phase flow. *J. Comput. Phys.* **210**, 225–246.
[7] E. Olsson, G. Kreiss, S. Zahedi (2007). A conservative level set method for two phase flow II. *J. Comput. Phys.* **225**, 785–807.
[8] G. Kreiss (2007). *Modeling of contact line dynamics for two–phase flow*. ICIAM2007 proceedings.
[9] S. Zahedi, K. Gustavsson, G. Kreiss (2008). *A new model for contact line dynamics*. TRITA–NA, Royal Institute of Technology, Sweden, ISSN 0348–2952.
[10] E. Bänsch (1998). *Numerical methods for the instationary Navier–Stokes equations with a free capillary surface*. Habiliation thesis, University of Freiburg, Germany.
[11] S. Gross, A. Reusken (2007). Finite element discretization error analysis of a surface tension force in two–phase incompressible flows. *SIAM J. Numer. Anal.* **45**, 1679–1700.
[12] J. Sethian (1999). *Level Set Methods and Fast Marching Methods*. Cambridge University Press.
[13] S. Osher, R. Fedkiw (2003). Level Set Methods and Dynamic Implicit Surfaces. Springer–Verlag.
[14] A. Harten (1977). The artificial compression method for computation of shocks and contact discontinuities. I. Single conservation laws. *Comm. Pure Appl. Math.* **30** 611–638.
[15] J.–L. Guermond, L. Quartapelle (1997). Calculation of incompressible viscous flows by an unconditionally stable projection FEM. *J. Comput. Phys.* **132**, 12–33.
[16] G. Amberg, R. Tönhardt, C. Winkler (1999). Finite element simulations using symbolic computing. *Math. Comput. Simulat.* **49**, 257–274.
[17] R. G. Cox (1986). The dynamics of the spreading of liquids on a solid surface. Part1. Viscous flow. *J. Fluid Mech.* **168**, 169–194.

Paper 3

Delta function approximations in level set methods by distance function extension

Sara Zahedi and Anna-Karin Tornberg
Manuscript to be submitted to Journal Computational Physics

Delta Function Approximations in Level Set Methods by Distance Function Extension

Sara Zahedi and Anna–Karin Tornberg
School of Computer Science and Communication,
Royal Institute of Technology, 100 44 Stockholm, Sweden

Abstract

In [J. Comput. Phys. 200 (2004) 462–488], it was shown for simple examples that the then most common way to regularize delta functions in connection to level set methods produces inconsistent approximations with errors that are not reduced with grid refinement. Since then, several clever approximations have been derived to overcome this problem. However, the great appeal of the old method was its simplicity. Here, we show that the old method – a one–dimensional delta function approximation extended to higher dimensions by a distance function – can be made accurate with a different class of one–dimensional delta function approximations. The prize that we pay is a wider support of the resulting delta function approximations.

1 Introduction

The level set method, originally devised by Osher and Sethian [1], is a very popular method for the evolution of interfaces, and it has been implemented for numerous applications. In some of these applications, the question of how to numerically approximate a Dirac delta function arises. For example, in immiscible multiphase problems, Dirac delta functions supported on interfaces separating different fluids are often used in the modeling of the surface tension forces acting on the interfaces. Another example is the problem of evaluating a line integral in two dimensions or a surface integral in three dimensions. This problem can conveniently be reformulated as an integral in 2D or 3D involving a Dirac delta function with support on the line or surface. An approach to approximate such delta functions is to extend a regularized one–dimensional delta function to higher dimensions using a distance function. This has been a common technique in connection to level set methods [2] since the distance function is usually available discretized on a computational

grid. However, care is needed since the extension to higher dimensions using a distance function may lead to $\mathcal{O}(1)$ errors [3]. In [3], it was shown that another extension technique that is based on products of regularized one–dimensional delta functions [4] is consistent. This technique is however only applicable when an explicit representation of the curve or the surface is available. In level set methods the curve or the surface is represented implicitly by a level set [2, 5].

To overcome the lack of consistency that became apparent with the work presented in [3], a number of consistent delta function approximations that can be used with level set methods have been proposed. Engquist et al. [6] proposed two such approximations. The first one is an approximation of the product rule using the distance function and its gradient. The second one is based on the linear hat function but uses a variable regularization parameter. Smereka [7] derived a discrete delta function obtained as the truncation error in solving the Laplacian of the Green's function, which was proven to be second order accurate by Beale [8]. Consistent approximations for which the level set function and its gradient are needed have also been introduced by Towers [9, 10]. The advantage of these methods is that the supports of the delta function approximations are very small. The discrete delta function proposed by Smereka has its support contained within a single mesh cell.

One way to explain the reason for the inconsistency shown in [3] is the following: The one–dimensional delta function approximation is designed to obey certain moment conditions on a uniform grid. The first moment condition is the mass condition that ensures that the delta function approximation sums to one independent of shifts in the grid. Delta function approximations with compact support where the widths of the approximations are fixed to a number of cell widths was considered in [3]. As the one–dimensional delta function is extended to higher dimensions, using the closest distance to the line or surface, the effective width in each coordinate direction relative to the grid size will depend on the slope of the curve or surface. This will in general no longer be within the design of the one–dimensional delta approximation, causing a violation of the mass condition, and hence an $\mathcal{O}(1)$ error that will not vanish with grid refinement. This is further discussed in Section 3. This effect was recognized by Engquist et al. [6] who introduced a first correction to this, by defining the regularization parameter to depend on the gradient of the distance function.

The problem with extending the delta function to higher dimensions using the closest distance to the line or surface is hence that the one–dimensional delta approximation is dilated, and that the moment conditions are no longer valid. One can however construct delta approximations such that the moment conditions do hold for a wide range of dilations. These functions are however not of compact support. One such function was given in [11]. It has compact

support in Fourier space, and decays rapidly enough in real space to lend itself to truncation, but the effective support will be wider than one or two grid points as in the approximations above. In addition, in difference to the delta function approximations discussed above, that are of low regularity, this function is infinitely differentiable.

Another way to recognize the problem of extending a compact one–dimensional delta function approximation by the distance rule, is to use the result from the analysis in [12]. Here, the error is split into two parts. The first part is the analytical error due to the approximation of the delta function. The second part is the numerical error due to the approximation of the integral containing the delta function approximation. For the first part, it is continuous moment conditions that are important, and for the second part, it is (in addition to the order of the quadrature rule) the regularity of the delta function approximation that limits the accuracy. This gives an upper limit of the error, but in the case when extending by the distance rule, the error is quite close to this upper limit. The result from this analysis is that the numerical error is of order $\mathcal{O}((h/\varepsilon)^p)$ where p is determined by the regularity of the delta function approximation, which is typically low. This shows that with a choice of $\varepsilon = mh$, which has been the common choice, the numerical error is of $\mathcal{O}(1)$, and the method is inconsistent. Depending on the regularity and continuous moment order of the approximation, there is an optimal $\alpha < 1$ in $\varepsilon \sim h^\alpha$ that results in the best convergence. If we replace the narrow delta function approximation with an infinitely differentiable delta function approximation, the result is quite different. The regularity of the delta function approximation will no longer limit the accuracy of the quadrature rule. In fact, these functions can be considered as periodic functions, since they decay to zero. For these functions, the trapezoidal rule on a uniform grid will converge faster than any power of h in the limit as $h \to 0$. This is often referred to as the superconvergence of the trapezoidal rule, and will yield a very small numerical error.

In this paper, we will consider three different functions that all have these properties. We will now provide a comparison between one of the delta function approximations considered in this paper and the narrow linear hat function which in [3] was shown to give $\mathcal{O}(1)$ errors. Consider the computation of the arc–length of a circle of radius 1 centered at origin by evaluating

$$\int_\Omega \delta_\varepsilon(d(\Gamma, \mathbf{x}))d\Omega,$$

where the computational domain Ω is discretized with a regular mesh with mesh size h. Use the trapezoidal rule for the integration. For the narrow

linear hat function

$$\delta_{2h}^{L}(x) = \begin{cases} \frac{1}{2h}(1 - \frac{|x|}{2h}), & \text{if } |x| \le 2h, \\ 0, & \text{if } |x| > 2h, \end{cases} \tag{1}$$

there is no analytical error in the computation of the arc–length, see [12]. Still there is no convergence as $h \to 0$:

h	0.1	0.05	0.025	0.0125	0.00625
Relative error	$2.2 \cdot 10^{-3}$	$8 \cdot 10^{-4}$	$8 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$4 \cdot 10^{-4}$

Due to the symmetry of the problem and resulting cancellation of errors, these errors are quite small compared to the errors in the examples given in Section 3. We use the same technique with the one–dimensional delta function $\delta_\varepsilon(x) = \delta_{2h}^{FD}(x)$ defined as the derivative of the Fermi–Dirac function

$$\delta_{2h}^{FD}(x) = \partial_x \frac{1}{1 + e^{-x/(2h)}}. \tag{2}$$

This delta function approximation was used in the conservative level set method [13, 14]. For this approximation, the error decreases exponentially down to the relative floating point error:

h	0.1	0.05	0.025	0.0125	0.00625
Relative error	$4.4 \cdot 10^{-3}$	$2 \cdot 10^{-5}$	$7 \cdot 10^{-10}$	$1 \cdot 10^{-14}$	$4 \cdot 10^{-15}$

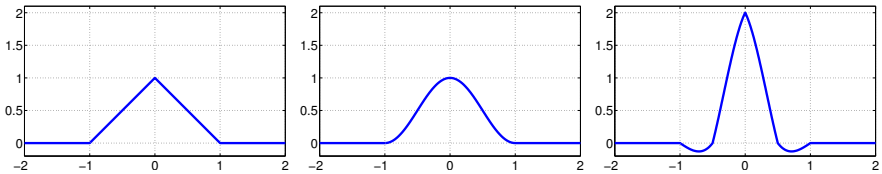
What we see is the superconvergence of the trapezoidal rule for infinitely differentiable periodic functions.

This paper is organized as follows. In Section 2 we define delta function approximations and state conditions for accuracy in one–dimension. In Section 3 we discuss the simple example of computing the length of a line. We show why the compact delta function approximations produce $O(1)$ errors and how large they are. We also show why this does not occur for approximations with compact support in Fourier space. In Section 4 we introduce three different consistent delta function approximations and discuss their properties. In Section 5 we state and prove theorems for the error in both two and three dimensions. We present numerical experiments in Section 6 and summarize our results in Section 7.

2 Regularization

Given a function $\varphi(\xi)$, a delta function approximation can be constructed by

$$\delta_\varepsilon(x) = \frac{1}{\varepsilon}\varphi(x/\varepsilon). \tag{3}$$



(a) $\varphi^L(\xi)$ (b) $\varphi^{\cos}(\xi)$ (c) $\varphi^C(\xi)$

Figure 1: Building blocks $\varphi(\xi)$ for delta function approximations. A linear hat function (a), a cosine approximation (b), and a piecewise cubic function (c).

Examples of such $\varphi(\xi)$ functions with compact support are: the piecewise linear hat function

$$\varphi^L(\xi) = \begin{cases} (1 - |\xi|), & \text{if } |\xi| \leq 1, \\ 0, & \text{if } |\xi| > 1, \end{cases} \tag{4}$$

the cosine approximation

$$\varphi^{\cos}(\xi) = \begin{cases} \frac{1}{2}(1 + \cos(\pi\xi)), & \text{if } |\xi| \leq 1, \\ 0, & \text{if } |\xi| > 1, \end{cases} \tag{5}$$

and the piecewise cubic function

$$\varphi^C(\xi) = \begin{cases} 2 - |2\xi| - 2|2\xi|^2 + |2\xi|^3, & \text{if } 0 \leq |\xi| \leq 1/2, \\ 2 - \frac{11}{3}|2\xi| + 2|2\xi|^2 - \frac{1}{3}|2\xi|^3, & \text{if } 1/2 < |\xi| \leq 1, \\ 0, & \text{if } |\xi| > 1. \end{cases} \tag{6}$$

The functions $\varphi^L(\xi)$, $\varphi^{\cos}(\xi)$, and $\varphi^C(\xi)$ are plotted in Fig. 1.

In the next section we state the conditions the regularized one dimensional delta function must satisfy in order to be accurate.

2.1 Discrete regularization in one–dimension

Assume a regular grid in one–dimension, with grid size h and grid points $x_j = jh$, $j \in \mathbb{Z}$. We introduce the discrete moment conditions:

Definition 2.1. *A function δ_ε satisfies q discrete moment conditions if for all $x^* \in \mathbb{R}$,*

$$M_r(\delta_\varepsilon, x^*, h) = h \sum_{j \in \mathbb{Z}} \delta_\varepsilon(x_j - x^*)(x_j - x^*)^r = \begin{cases} 1, & \text{if } r = 0, \\ 0, & \text{if } 1 \leq r < q, \end{cases} \tag{7}$$

where $x_j = jh$, $h > 0$, $j \in \mathbb{Z}$.

If δ_ε satisfies q moment conditions, we say that it has a moment order q. The first moment condition ensures that the mass of the delta function approximation δ_ε is one, independent of shifts in the grid. It is therefore referred to as the mass condition. The higher moment conditions are important when the delta approximation is multiplied by a non–constant function. The following theorem states that in one–dimension the numerical accuracy of a regularized delta function is determined by the number of discrete moment conditions.

Proposition 2.1. *Assume δ_ε satisfies q discrete moment conditions and has compact support in $[-Mh, Mh]$. Assume also that $f(x) \in C^q(\mathbb{R})$, and that all derivatives of f are bounded, then*

$$E = \left| h \sum_j \delta_\varepsilon(x_j - x^*) f(x_j) - f(x^*) \right| \leq Ch^q \tag{8}$$

and $E = 0$ if f is constant.

A proof based on Taylor expansion of f around $x^* \in \mathbb{R}$ is found in Refs. [15, 3]. The delta function approximations δ_ε in last section have support in $[-\varepsilon, \varepsilon]$. The linear hat function $\delta_{2h}^L(x)$, the cosine approximation $\delta_{2h}^{\cos}(x)$ and the cubic function $\delta_{2h}^C(x)$ all satisfy the mass condition and hence are consistent approximations. The cubic function $\delta_{2h}^C(x)$ is most accurate. It satisfies four discrete moment conditions and is according to Proposition 2.1 a fourth order accurate approximation.

2.2 Extensions to higher dimensions

A Dirac delta measure concentrated on a curve or surface can be approximated by extending a regularized one–dimensional delta function to higher dimensions. Basically two techniques are used. One is the product formula and the other technique is based on a distance function to the curve or the surface.

Let $\Gamma \subset \mathbb{R}^d$ be a $d-1$ dimensional closed, continuous, and bounded surface and let S be surface coordinates on Γ. The product formula yields

$$\delta_\varepsilon(\Gamma, g, \mathbf{x}) = \int_\Gamma \prod_{k=1}^{d} \delta_{\varepsilon_k}(x^{(k)} - X^{(k)}(S)) g(S) dS, \tag{9}$$

where δ_{ε_k} is a one–dimensional regularized delta function, $\mathbf{x} = (x^{(1)}, \cdots x^{(d)})$, $\mathbf{X}(S) = (X^{(1)}(S), \cdots, X^{(d)}(S))$ is a parametrization of Γ and $\varepsilon = (\varepsilon_1, \cdots, \varepsilon_d)$.

Assume that the space $\mathbb{R}^d$ is covered by a regular grid

$$\{\mathbf{x_j}\}_{\mathbf{j}\in\mathbb{Z}^d}, \quad \mathbf{x_j} = (x_{j_1}^{(1)}, \cdots, x_{j_d}^{(d)}),$$
$$x_{j_k}^{(k)} = x_0^{(k)} + j_k h_k, \quad j_k \in \mathbb{Z}, \quad k = 1, \cdots, d. \tag{10}$$

The following theorem was proved by Tornberg and Engquist in Ref. [3].

Theorem 2.1. *Suppose that δ_ε is a one–dimensional delta function approximation with compact support in $[-\varepsilon,\ \varepsilon]$, that satisfies q discrete moment conditions (see Definition 2.1); $g \in C^r(\mathbb{R}^d)$ and $f \in C^r(\mathbb{R}^d)$, $r \geq q$. Then for any rectifiable curve Γ and $\delta_\varepsilon(\Gamma, g, \mathbf{x})$ as defined in equation (9) with $\varepsilon = (mh_1, mh_2 \cdots, mh_d)$, it holds that*

$$E = \left| \left(\prod_{k=1}^{d} h_k \right) \sum_{j\in\mathbb{Z}^d} \delta_\varepsilon(\Gamma, g, \mathbf{x}_j) f(\mathbf{x}_j) - \int_\Gamma g(S) f(\mathbf{X}(S)) dS \right| \leq C h^q \tag{11}$$

with $h = \max_{1\leq k\leq d} h_k$ and $E = 0$ for constant f.

This means that the results from one–dimension carry over to higher dimensions, and that it is still the discrete moment order of the one–dimensional delta function approximation that determines the order of accuracy.

The product formula is easy to use when Γ is explicitly defined. However, in level set methods, Γ is defined implicitly by a level set function $\phi(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$,

$$\Gamma = \{\mathbf{x} \mid \phi(\mathbf{x}) = 0\}. \tag{12}$$

It is therefore preferable to use this function to extend the regularized one–dimensional delta function δ_ε to higher dimensions. In the case when $\phi(\mathbf{x}) = d(\Gamma, \mathbf{x})$, the signed distance function to Γ, where the distance is the Euclidean distance from $\mathbf{x}$ to Γ, the delta function approximation is defined as

$$\delta_\varepsilon(\Gamma, g, \mathbf{x}) = \tilde{g}(\mathbf{x}) \delta_\varepsilon(d(\Gamma, \mathbf{x})), \tag{13}$$

where $\tilde{g}$ is an extension of g to $\mathbb{R}^d$, such that $\tilde{g}(X(S)) = g(S)$. In this paper we focus on the extension to higher dimensions that employs the distance function.

3 Computing the length of a straight line

In this section we study the error made in the computation of the length of a straight line. In the computations a regularized one–dimensional delta function δ_ε is extended to higher dimensions using a distance function, according to equation (13) with $g = 1$.

Consider the problem of calculating the length of a curve Γ:

$$|\Gamma| = \bar{S} = \int_{\mathbb{R}^2} \delta(\Gamma, \mathbf{x}) d\mathbf{x}. \tag{14}$$

In the computation of $|\bar{S}|$, a delta function approximation δ_ε on a regular grid is used:

$$\begin{aligned} &\bar{S}_h = h^2 \sum_{j \in \mathbb{Z}^2} \delta_\varepsilon(d(\Gamma, \mathbf{x_j})), \\ &\mathbf{x_j} = (x_{j_1}, y_{j_2}), \quad x_{j_1} = j_1 h, \quad y_{j_2} = j_2 h, \quad j_l \in \mathbb{Z}, \quad l = 1, 2. \end{aligned} \tag{15}$$

In the following we will let the curve $\Gamma \in \mathbb{R}^2$ be a straight line with slope k, but not a vertical line. In Ref. [3] it was shown that for $\Gamma = \{\mathbf{x},\ x^{(2)} = x^{(1)},\ 0 \le x^{(1)} < \bar{S}/\sqrt{2}\}$, a line with slope $k = 1$ and δ_ε being equal to the narrow linear hat function δ_h^L the relative error $|\bar{S}_h - \bar{S}|/\bar{S}$ is more than 12% as $h \to 0$. This was shown by dividing the sum $\bar{S}_h$ into contributions of M subsegments of Γ, each of length $\sqrt{2}h$. Here, we consider a different approach. We express the error in terms of the first discrete moment of δ_ε (see equation (7) with $r = 0$). We can then show which delta function approximations that will produce $\mathcal{O}(1)$ errors. For simplicity, we consider a line of infinite length so we do not have to worry about contributions from end point terms.

From equation (3) we see that we can write $\delta_\varepsilon(x)$ as

$$\delta_\varepsilon(x) = \beta \delta_{\tilde{\varepsilon}}(\beta x), \quad \tilde{\varepsilon} = \beta\varepsilon. \tag{16}$$

We take

$$\beta = \frac{1}{k}\sqrt{1 + k^2}, \tag{17}$$

where k is the slope of Γ as defined above. Using equation (16) the computed length of a straight line $\bar{S}_h$ (as given in (15)) is

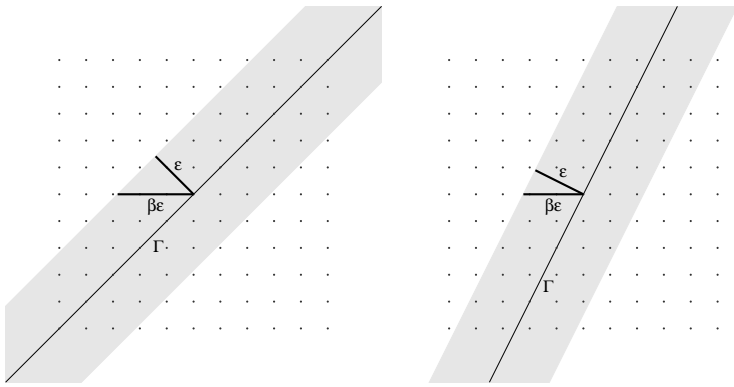
$$\bar{S}_h = h \sum_{j_2 \in \mathbb{Z}} \left(h \sum_{j_1 \in \mathbb{Z}} \delta_\varepsilon(d(\Gamma, x_{j_1}, y_{j_2})) \right) = h \sum_{j_2 \in \mathbb{Z}} \left(\beta h \sum_{j_1 \in \mathbb{Z}} \delta_{\tilde{\varepsilon}}(\beta d(\Gamma, x_{j_1}, y_{j_2})) \right). \tag{18}$$

It can be verified from Fig. 2 that $\beta d(\Gamma, x_{j_1}, y_{j_2}) = x_{j_1} - x^*(y_{j_2})$ where

$$x^*(y_{j_2}) = x_n + ph, \quad 0 \le p < 1, \quad n \in \mathbb{Z} \tag{19}$$

is the x–coordinate of Γ at $y = y_{j_2}$. Using the definition of the first discrete moment condition we can express $\bar{S}_h$ in terms of the first moment of $\delta_{\tilde{\varepsilon}}$ in the x–direction:

$$\bar{S}_h = h \sum_{j_2 \in \mathbb{Z}} \left(\beta h \sum_{j_1 \in \mathbb{Z}} \delta_{\tilde{\varepsilon}}(x_{j_1} - x^*(y_{j_2})) \right) = \beta h \sum_{j_2 \in \mathbb{Z}} M_0(\delta_{\tilde{\varepsilon}}, x^*(y_{j_2}), h). \tag{20}$$



(a) (b)

Figure 2: Γ is a straight line with slope k, $\beta = \frac{1}{k}\sqrt{1+k^2}$, and $\varepsilon = 2h$. In Panel (a) $k = 1$, $\beta = \sqrt{2}$, and x^*, defined in equation (19) is always a grid point, i.e $p = 0$. In Panel (b) $k = 2$, $\beta = \frac{\sqrt{5}}{2}$, and x^* is either a grid point or lies in the middle of two grid points, i.e $p = 0$ and $p = 1/2$ every second time.

For the linear hat function δ_ε^L and the cosine approximation $\delta_\varepsilon^{\cos}$ with $\varepsilon = mh$ we have from Ref. [11] that

$$M_0(\delta_{mh}^L, x^*, h) = \frac{1}{m^2}\left((k_0+k_1+1)m + (k_1-k_0-1)p - \frac{k_0(k_0+1)}{2} - \frac{k_1(k_1+1)}{2}\right) \tag{21}$$

and

$$M_0(\delta_{mh}^{\cos}, x^*, h) = \frac{k_0+k_1+1}{2m} + \frac{\sin((k_0+k_1+1)\pi/2m)\cos(((k_1-k_0)/2-p)\pi/m)}{2m\sin(\pi/2m)}, \tag{22}$$

where

$$k_0 = \lfloor m-p \rfloor, \quad k_1 = \lfloor m+p \rfloor. \tag{23}$$

Here, $\lfloor m \rfloor$ denotes m rounded to the nearest integer towards minus infinity. We recall from Section 2.1 that one–dimensional delta function approximations are consistent if they fulfill the mass condition i.e. $M_0(\delta_{mh}, x^*, h) = 1$ for any shift in the grid. The linear hat function with half width support $\varepsilon = mh$ satisfies the mass condition when m is an integer. The cosine function with half width support $\varepsilon = mh$ satisfies the mass condition when $2m$ is an integer. In this case, when Γ is a straight line, the effective half width support is $\beta\varepsilon$ (see Fig. 2) and βm and $2\beta m$, respectively, must be integers in order for the linear hat function and the cosine function to be consistent approximations.

Using equation (20) together with the formulas (21) and (22) we can evaluate the error in the computation of the length of a straight line with slope k using the linear hat function or the cosine approximation. The line in Fig. 2(a) has slope $k = 1$ and it intersects the grid points, hence $(x^*(y_{j_2}), y_{j_2})$ is always a grid point i.e. $p = 0$. Using the delta approximation $\delta_\varepsilon = \delta_h^L$ as in [3] we get from equation (20) and (21) with $\beta = \sqrt{2}$, $m = \beta$, and $p = 0$

$$\bar{S}_h = \frac{1}{2}\left(3\sqrt{2} - 2\right) \sum_{j_2 \in \mathbb{Z}} \sqrt{2}h = 1.1213 \sum_{j_2 \in \mathbb{Z}} \sqrt{2}h. \tag{24}$$

This indicates a relative error of over 12% independent of the mesh size h. This was also observed in [3]. With the delta approximation $\delta_\varepsilon = \delta_{2h}^{\cos}$ we get from equation (20) and (22) with $\beta = \sqrt{2}$, $m = 2\beta$, and $p = 0$

$$\bar{S}_h = \frac{1}{2m}\left(5 + \frac{\sin(5\pi/(2m))}{\sin(\pi/(2m))}\right) \sum_{j_2 \in \mathbb{Z}} \sqrt{2}h \approx 1.0035 \sum_{j_2 \in \mathbb{Z}} \sqrt{2}h. \tag{25}$$

This shows that a relative error of 0.35% independent of the mesh size h is expected when the approximation $\delta_{2h}^{\cos}$ is used. The line in Fig. 2(b) has slope $k = 2$ and it either intersect a grid point or lies in the middle of two grid points. Hence every second time $p = 1/2$ instead of $p = 0$. Thus, using $\delta_\varepsilon = \delta_h^L$ equation (20) and (21) with $\beta = \frac{\sqrt{5}}{2}$, $m = \beta$ and $p = 0$, $p = 1/2$ every second time gives

$$\bar{S}_h = \left(\frac{3\sqrt{5} - 4}{5} + \frac{2\sqrt{5} - 2}{5}\right) \sum_{j_2 \in \mathbb{Z}} \frac{\sqrt{5}}{2}h \approx 1.0361 \sum_{j_2 \in \mathbb{Z}} \frac{\sqrt{5}}{2}h. \tag{26}$$

This results in a relative error of 3.61% independent of mesh size. For $\delta_{2h}^{\cos}$ a relative error of around 0.013% is expected.

3.1 Numerical validation

In order to validate the results of the previous subsection we consider Γ being two parallel lines of length L at a normal distance $2a$, joined at both ends by a half circle with radius a. The slope of the lines to the x–axis is k. The length of Γ is $\bar{S} = 2L + 2\pi a$. Contours of the distance function $d(\Gamma, \mathbf{x})$ when $k = 2$ are shown in Fig. 3.

In Fig. 4(a) we show the relative error $E = |\bar{S}_h - \bar{S}|/\bar{S}$ where S_h is computed according to equation (15) with $\delta_\varepsilon = \delta_{2h}^{\cos}$ and $k = 1$. We can clearly see that there is no convergence as $h \to 0$. Equation (25) indicates that the relative error for the straight lines is around 0.0035. We can see in the figure that this number is approached when the length of the straight lines is increased and the radius of the half circles is decreased.

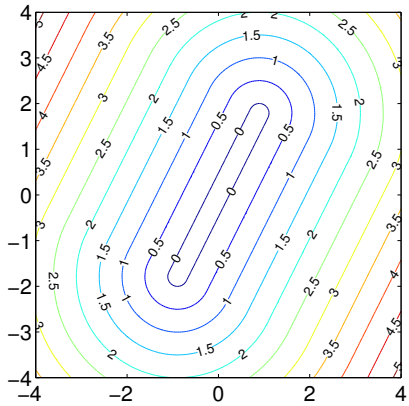


Figure 3: Contours of the distance function $d(\Gamma, \mathbf{x})$. Γ is two parallel lines of length $L = 4$ at a normal distance $2a$, joined at both ends by a half circle of radius $a = 0.48/\sqrt{5}$. The slope of the lines to the x–axis is 2.

In Fig. 4(b) the slope of the parallel lines is 2 and $\delta_\varepsilon = \delta_h^L$ has been used in the computation. There is no convergence. Equation (26) indicates that the relative error for the straight lines is around 0.0361. We see in the figure that as the length of the straight lines is increased or the radius of the half circles is decreased the relative error approaches this number.

3.2 Mass condition reformulated using a Fourier transform

By the use of Poisson's summation formula

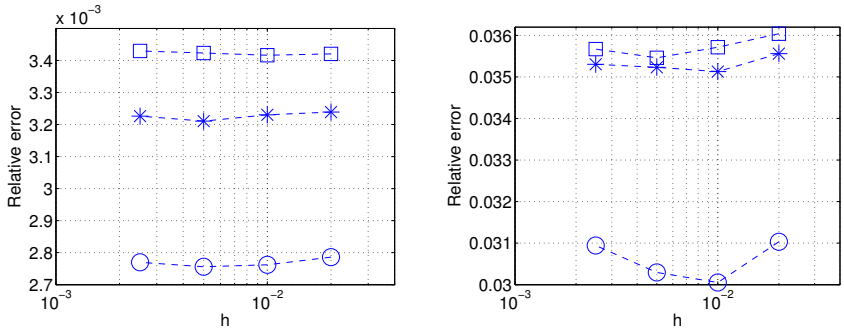
$$\alpha \sum_{j\in\mathbb{Z}} \varphi(\alpha j) = \sum_{j\in\mathbb{Z}} \hat{\varphi}(j/\alpha), \tag{27}$$

the mass condition for a regularized delta function $\delta_\varepsilon = \frac{1}{\varepsilon}\varphi(x/\varepsilon)$ with $\varepsilon = mh$ can be related to the Fourier transform of the function $\varphi(\xi)$

$$\hat{\varphi}(k) = \int_{-\infty}^{\infty} \varphi(\xi) e^{-2\pi i k\xi} d\xi, \tag{28}$$

in the following way:

$$M_0(\delta_{mh}, x^*, h) = \frac{1}{m} \sum_{j\in\mathbb{Z}} \varphi((j-p)/m) = \sum_{j\in\mathbb{Z}} e^{-2\pi i j p} \hat{\varphi}(jm). \tag{29}$$



(a) (b)

Figure 4: The relative error $E = |\bar{S}_h - \bar{S}|/\bar{S}$ where Γ is two parallel lines of length L at a normal distance $2a$, joined at both ends by a half circle of radius a. In Panel (a) the slope of the parallel lines is $k = 1$ and $\delta_\varepsilon = \delta_{2h}^{\cos}$ is used. Circles: $L = 4$, $a = 0.24\sqrt{2}$. Stars: $L = 6$, $a = 0.12\sqrt{2}$. Squares: $L = 6$, $a = 0.03\sqrt{2}$. The predicted relative error for the length of lines is around $3.5 \cdot 10^{-3}$. In Panel (b) the slope of the parallel lines is $k = 2$ and $\delta_\varepsilon = \delta_h^L$ is used. Circles: $L = 4$, $a = 0.48/\sqrt{5}$. Stars: $L = 6$, $a = 0.12/\sqrt{5}$. Squares: $L = 6$, $a = 0.06\sqrt{2}$. The predicted relative error for the length of lines is around 0.0361.

The linear hat function, $\varphi^L(\xi)$, defined as in equation (4), has the Fourier transform

$$\hat{\varphi}^L(k) = \frac{\sin^2(\pi k)}{\pi^2 k^2}. \tag{30}$$

Thus, using equation (29) we have

$$M_0(\delta_\varepsilon^L, x^*, h) = 1 + \sum_{j\in\mathbb{Z}, j\neq 0} e^{-2\pi i j p} \left(\frac{\sin(\pi m j)}{\pi m j}\right)^2, \tag{31}$$

Here, we have used that $\hat{\varphi}^L(0) = 1$. The second term in equation (31) is zero independent of any shift in the grid x^* only when $\sin(\pi m j) = 0$ for all $j \in \mathbb{Z}, j \neq 0$. Therefore the mass condition is satisfied only for integers $m \geq 1$. This result can also be obtained using formula (21).

The cosine approximation $\varphi^{\cos}(\xi)$, defined in equation (5), has the Fourier transform

$$\hat{\varphi}^{\cos}(k) = \begin{cases} \left(\frac{\sin(2\pi k)}{2\pi k}\right)\frac{1}{1-4k^2}, & \text{if } k \neq \pm 1/2, \\ 1/2, & \text{if } k = \pm 1/2. \end{cases} \tag{32}$$

By the same argument as above we have that the mass condition is satisfied only when $\sin(2\pi m j) = 0$ for all $j \in \mathbb{Z}, j \neq 0$ and $m \neq 1/2$. Therefore, the mass condition is satisfied when $m \geq 1$ and $2m$ is an integer.

In equation (20) we needed to evaluate $M_0(\delta_{\tilde{\varepsilon}}, x^*, h)$, where $\tilde{\varepsilon} = \beta m h$. From Poisson's summation formula we have

$$M_0(\delta_{\tilde{\varepsilon}}, x^*, h) = \sum_{j\in\mathbb{Z}} e^{-2\pi i j p} \hat{\varphi}(j\beta m). \tag{33}$$

Thus, in order to not have $\mathcal{O}(1)$ errors in the computation of the length of a straight line, m in $\varepsilon = mh$ must be chosen such that $\beta m \geq 1$ is an integer when $\delta_\varepsilon = \delta_\varepsilon^L$ and $\beta m \geq 1$, $2\beta m$ is an integer when $\delta_\varepsilon = \delta_\varepsilon^{\cos}$.

Remark The regularized one–dimensional delta functions were in this section extended using the distance function. If instead a non distance function $\phi(\mathbf{x})$ is used ε must be chosen differently in order to avoid $\mathcal{O}(1)$ errors. This is due to the fact that $\phi(\mathbf{x})$ does not give the physical distance, and hence a different scaling is needed.

Assume that $\hat{\varphi}(k)$ has compact support on $(-1, 1)$ and that $\hat{\varphi}(0) = 1$. Then, by equation (33), for all $m \geq 1/\beta$, the mass condition is satisfied. Note that if $\phi(\mathbf{x})$ is a distance function, $\beta > 1$. Therefore, for all $m \geq 1$, there is no $\mathcal{O}(1)$ error. However, when $\phi(\mathbf{x})$ is not a distance function $\beta \leq 1$. Thus a harder restriction on m is needed in order to not obtain $\mathcal{O}(1)$ errors.

In the next section, we introduce a class of one–dimensional delta functions for which the $\hat{\varphi}(k)$ functions have compact support. We will see that this type of delta function approximations will satisfy the discrete moment conditions for a wide range of dilations.

4 Approximations with compact Fourier transform

In the last section we saw that the linear hat function δ_ε^L and the cosine approximation $\delta_\varepsilon^{\cos}$ with $\varepsilon = mh$ are consistent approximations in one–dimension only for a discrete set of m–values. Therefore they can lead to inconsistent approximations in higher dimensions. However, it is possible to construct delta function approximations that obey the mass condition for a wide range of dilations. We start by stating a theorem

Theorem 4.1. *Assume a regular grid in one–dimension with grid points $x_j = jh$, $j \in \mathbb{Z}$ and let $x^* = x_n + ph$ where $0 \le p < 1$ and $n \in \mathbb{Z}$. Consider a delta function approximation $\delta_\varepsilon = \frac{1}{\varepsilon}\varphi(x/\varepsilon)$ with $\varepsilon = mh$ where*

$$\varphi(\xi) = \int_{-\infty}^{\infty} \hat{\varphi}(k) e^{2\pi i k \xi} dk \tag{34}$$

and the Fourier transform of φ

$$\hat{\varphi}(k) = \int_{-\infty}^{\infty} \varphi(\xi) e^{-2\pi i k \xi} d\xi. \tag{35}$$

If $\hat{\varphi}(k)$ has compact support on $(-\beta, \beta)$ and

$$\left.\frac{\partial^r \hat{\varphi}(k)}{\partial k^r}\right|_{k=0} = \begin{cases} 1, & \text{for } r = 0, \\ 0, & \text{for } 1 \le r < q. \end{cases} \tag{36}$$

Then for all $m \ge \beta$ the delta function approximation δ_ε satisfies q discrete moment conditions.

The discrete moment conditions are important for the accuracy of the one–dimensional delta function approximation, see Proposition 2.1. The conditions in equation (36) in the theorem are equivalent to the continuous moment conditions i.e.

$$\left.\frac{\partial^r \hat{\varphi}(k)}{\partial k^r}\right|_{k=0} = \begin{cases} 1, & \text{for } r = 0, \\ 0, & \text{for } 1 \le r < q \end{cases} \quad \Leftrightarrow \int_{-\infty}^{\infty} \varphi(x) x^r dx = \begin{cases} 1, & \text{for } r = 0, \\ 0, & \text{for } 1 \le r < q. \end{cases} \tag{37}$$

The continuous moment conditions will be important as we consider the analytical error in higher dimensions.

Proof of Theorem 4.1. There is no restriction in taking $n = 0$, such that $x^* = ph$, with $0 \le p < 1$. Let $f_{r,\varepsilon}(x) = \frac{1}{\varepsilon}\varphi(x/\varepsilon)x^r$. We have

$$\begin{aligned} M_r(\delta_\varepsilon, x^*, h) &= h\sum_{j\in\mathbb{Z}} \delta_\varepsilon(x_j - x^*)(x_j - x^*)^r \\ &= h\sum_{j\in\mathbb{Z}} \frac{1}{\varepsilon}\varphi((x_j - x^*)/\varepsilon)(x_j - x^*)^r = \\ &= h\sum_{j\in\mathbb{Z}} f_{r,\varepsilon}((j-p)h). \end{aligned} \tag{38}$$

Since the Fourier transform of $f_{r,\varepsilon}(x)$ is

$$\hat{f}_{r,\varepsilon}(k) = \frac{1}{(-2\pi i)^r}\frac{\partial^r}{\partial k^r}\hat{\varphi}(\varepsilon k) \tag{39}$$

and the Fourier transform of $f_{r,\varepsilon}(x - x^*)$ is $e^{-2\pi i k x^*}\hat{f}_{r,\varepsilon}(k)$ we have from Poisson's summation formula that

$$M_r(\delta_\varepsilon, x^*, h) = \sum_{k\in\mathbb{Z}} e^{-2\pi i k p}\frac{1}{(-2\pi i)^r}\frac{\partial^r}{\partial k^r}\hat{\varphi}(\varepsilon k/h). \tag{40}$$

With $\varepsilon = mh$ we get

$$M_r(\delta_\varepsilon, x^*, h) = \frac{1}{(-2\pi i)^r}\left.\frac{\partial^r\hat{\varphi}(mk)}{\partial k^r}\right|_{k=0} + \frac{1}{(-2\pi i)^r}\sum_{k\in\mathbb{Z}, k\neq 0} e^{-2\pi i k p}\frac{\partial^r}{\partial k^r}\hat{\varphi}(mk). \tag{41}$$

If $\hat{\varphi}(k)$ has compact support in $(-\beta, \beta)$ the second term in the above equation vanishes for all $m \ge \beta$. Hence if the condition in equation (36) is satisfied for $r = 0, 1, \cdots, q-1$ then $\delta_\varepsilon = \frac{1}{\varepsilon}\varphi(x/\varepsilon)$ satisfies q discrete moment conditions. □

An example of a delta function approximation with the function $\hat{\varphi}(k)$ having compact support is the delta function introduced in Ref. [11]

$$\delta_\varepsilon^{TE}(x) = \frac{1}{\varepsilon}\varphi^{TE}(x/\varepsilon) \tag{42}$$

with

$$\varphi^{TE}(\xi) = \int_{-\infty}^{\infty}\hat{\varphi}^{TE}(k)e^{2\pi i k\xi}dk, \quad \hat{\varphi}^{TE}(k) = \begin{cases} e^{\frac{1}{d\beta^2}}e^{\frac{1}{d(k^2-\beta^2)}} & \text{if } |k| < \beta, \\ 0 & \text{if } |k| \ge \beta, \end{cases} \tag{43}$$

$d = 0.1$ and $\beta = 1$. Note that

$$\hat{\varphi}^{TE}(0) = 1, \quad \left.\frac{\partial \hat{\varphi}^{TE}(k)}{\partial k}\right|_{k=0} = 0, \quad \left.\frac{\partial^2 \hat{\varphi}^{TE}(k)}{\partial k^2}\right|_{k=0} = \frac{-2}{d\beta^4} \neq 0. \tag{44}$$

Hence from Theorem 4.1 we have that $\delta_{mh}^{TE}(x)$ is of moment order 2 for all $m \geq 1$. Thus, it is possible to construct one–dimensional delta function approximations that obey the discrete moment conditions for a wide range of dilations. These delta function approximations will not have compact support since their Fourier transforms have compact support. However, if an approximation is decaying rapidly it can in practice be truncated.

It is computationally demanding to evaluate the approximation from its Fourier transform. Therefore, we would like to have an explicit expression for the approximation. In the following we give explicit expressions for two delta function approximations which have Fourier transforms that decay rapidly. Theorem 4.1 can then be used to find m–values for which $\delta_{mh}(x)$ satisfies the discrete moment conditions within a given error tolerance.

4.1 The derivative of the Fermi–Dirac function

Define a delta approximation as the derivative of the Fermi–Dirac or the sigmoid function

$$\delta_\varepsilon^{FD}(x) = \partial_x \frac{1}{1 + e^{-x/\varepsilon}} = \frac{1}{\varepsilon} \frac{e^{-x/\varepsilon}}{(1 + e^{-x/\varepsilon})^2}. \tag{45}$$

Let then $\delta_\varepsilon^{FD}(x) = \frac{1}{\varepsilon}\varphi^{FD}(x/\varepsilon)$, where

$$\varphi^{FD}(\xi) = \frac{e^{-\xi}}{(1 + e^{-\xi})^2}. \tag{46}$$

The Fourier transform of $\varphi^{FD}(\xi)$ is

$$\hat{\varphi}^{FD}(k) = 1 - 4\pi k \Im \left(\sum_{j=1}^{\infty} \frac{(-1)^j}{j + 2\pi i k} \right), \tag{47}$$

where $\Im$ represents the imaginary part. This was obtained by differentiating the Fourier transform of the Fermi–Dirac function given in Ref. [16]. We have

$$\hat{\varphi}^{FD}(0) = 1, \quad \left.\frac{\partial^r \hat{\varphi}^{FD}(k)}{\partial k^r}\right|_{k=0} = 0 \tag{48}$$

for r odd. However, the second derivative

$$\left.\frac{\partial^2 \hat{\varphi}^{FD}(k)}{\partial k^2}\right|_{k=0} = 16\pi^2 \left(\sum_{j=1}^{\infty} \frac{(-1)^j}{j^2} \right) \neq 0. \tag{49}$$

Hence from Theorem 4.1 we have that $\delta_{mh}^{FD}(x)$ is of moment order 2 for all $m \geq \beta$ provided that $\hat{\varphi}^{FD}(k)$ has compact support in $(-\beta, \beta)$. Since $\hat{\varphi}^{FD}(k)$ does not have compact support, as was the case for $\hat{\varphi}^{TE}(k)$ we will always have a mass error but for $m = 2$ this error will be of order 10^{-16} which usually is the order of rounding errors. Therefore we consider $\hat{\varphi}^{FD}(k)$ to have compact support in $(-2, 2)$ and thus by Theorem 4.1 $\delta_{mh}^{FD}(x)$ is of moment order 2 for all $m \geq 2$. In Fig. 5(b) (solid line) we can see that for $k = \pm 1$, $\hat{\varphi}^{FD}(k)$ is of order 10^{-7}. This implies that taking $m = 1$ will typically give a mass error that is of order 10^{-7}.

The advantage of the approximation $\delta_{\varepsilon}^{FD}(x)$ compared to $\delta_{\varepsilon}^{TE}(x)$ is that it is given explicitly and is less computationally demanding to evaluate compared to $\delta_{\varepsilon}^{TE}(x)$.

4.2 The Gaussian function

A third example is the Gaussian function. The Fourier transform of a Gaussian is another Gaussian. Let

$$\varphi^{G}(\xi) = \sqrt{\frac{\pi}{9}} e^{-\pi^2 \xi^2 / 9}. \tag{50}$$

Then,

$$\hat{\varphi}^{G}(k) = e^{-9k^2}. \tag{51}$$

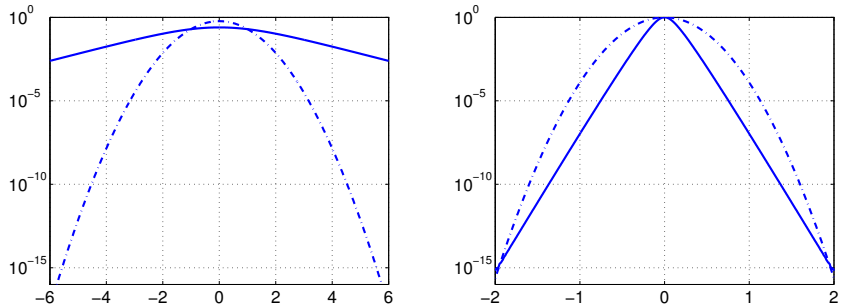
Also, for this function we have that

$$\hat{\varphi}^{G}(0) = 1, \quad \left. \frac{\partial^r \hat{\varphi}^{G}(k)}{\partial k^r} \right|_{k=0} = 0 \tag{52}$$

for all odd r but

$$\left. \frac{\partial^2 \hat{\varphi}^{G}(k)}{\partial k^2} \right|_{k=0} \neq 0. \tag{53}$$

Hence from Theorem 4.1 we have that $\delta_{mh}^{G}(x)$ is of moment order 2 for all $m \geq \beta$ provided that $\hat{\varphi}^{G}(k)$ has compact support in $(-\beta, \beta)$. Just as $\hat{\varphi}^{FD}(k)$, the function $\hat{\varphi}^{G}(k)$ is never zero. Therefore, it does not, strictly speaking, have compact support. However, since the function decreases exponentially to zero, it can in practice be regarded as zero whenever smaller than some tolerance. In Fig. 5(b) we see the function $\hat{\varphi}^{G}(k)$ (dash–dotted line). For $k = \pm 1$, $\hat{\varphi}^{G}(k)$ is of order 10^{-5} and for $k = \pm 2$ it is of order 10^{-16} just as $\hat{\varphi}^{FD}(k)$. With an error tolerance of 10^{-16} we consider $\hat{\varphi}^{G}(k)$ to be of moment order 2 for all $m \geq 2$. We can see from Fig. 5(a) that the support of $\varphi^{G}(\xi)$ is much smaller than $\varphi^{FD}(\xi)$. Since, in practical computations, a narrow support of $\varphi(\xi)$ is desired the Gaussian approximation seems to be preferable.



(a) $\varphi(\xi)$ (b) $\hat{\varphi}(k)$

Figure 5: The functions $\varphi(\xi)$ (Panel (a)) used to define the regularized delta functions $\delta_\varepsilon(x) = \frac{1}{\varepsilon}\varphi(x/\varepsilon)$, and their Fourier transforms $\hat{\varphi}(k)$ (Panel(b)). Solid lines: $\varphi(\xi) = \varphi^{FD}(\xi)$ and $\hat{\varphi}(k) = \hat{\varphi}^{FD}(k)$. Dash–dotted lines: $\varphi(\xi) = \varphi^{G}(\xi)$ and $\hat{\varphi}(k) = \hat{\varphi}^{G}(k)$.

In the next section we state and prove theorems about the error when the distance function is used to extend the one–dimensional delta approximations to higher dimensions.

5 Error analysis

Let Ω be the domain of integration. We assume throughout this section that the approximation $\delta_\varepsilon(d(\Gamma, \mathbf{x}))$ has compact support in

$$\Omega_\omega = \{\mathbf{x} : |d(\Gamma, \mathbf{x})| \leq \omega\}, \tag{54}$$

where $d(\Gamma, \mathbf{x})$ is the signed distance function and ω is small. We want to integrate

$$I_{\Gamma,F} = \int_\Omega \delta(d(\Gamma, \mathbf{x}))F(\mathbf{x})d\mathbf{x}. \tag{55}$$

The total error in the integration of the function $\delta(d(\Gamma, \mathbf{x}))F(\mathbf{x})$ approximated by $\delta_\varepsilon(d(\Gamma, \mathbf{x}))F(\mathbf{x})$ is

$$E_{tot,F}(\delta_\varepsilon) = \int_\Omega \delta(d(\Gamma, \mathbf{x}))F(\mathbf{x})d\mathbf{x} - \text{quad}(\delta_\varepsilon(d(\Gamma, \mathbf{x}))F(\mathbf{x})), \tag{56}$$

where quad denotes the quadrature rule used to approximate the integral. The quadrature rule we consider is the trapezoidal rule. We split the total

error into two parts: the analytical error made when replacing the integrand with its approximation

$$E_{\omega,F}(\delta_\varepsilon) = \int_{\Omega_\omega} \delta(d(\Gamma,\mathbf{x}))F(\mathbf{x})d\mathbf{x} - \int_{\Omega_\omega} \delta_\varepsilon(d(\Gamma,\mathbf{x}))F(\mathbf{x})d\mathbf{x} \tag{57}$$

and the numerical error made in the integration of this approximation using the trapezoidal rule

$$E_{\text{quad},F}(\delta_\varepsilon) = \int_{\Omega_\omega} \delta_\varepsilon(d(\Gamma,\mathbf{x}))F(\mathbf{x})d\mathbf{x} - \text{quad}(\delta_\varepsilon(d(\Gamma,\mathbf{x}))F(\mathbf{x})).$$

5.1 Analytical error

Definition 5.1. *A function δ_ε with compact support in $[-\omega,\omega]$ satisfies α continuous moment conditions if*

$$\int_{-\omega}^{\omega} \delta_\varepsilon(t)t^r dt = \begin{cases} 1, & \text{if } r = 0, \\ 0, & \text{if } 1 \le r < \alpha. \end{cases} \tag{58}$$

We now state two theorems for the analytical error. The first theorem provides an expression for the analytical error in two dimensions, and the second theorem in three dimensions.

Theorem 5.1. *Let δ_ε be a continuous function with support in $[-\omega,\omega]$, $\omega = p\varepsilon$ that satisfies α continuous moment conditions, see Definition 5.1. Assume that Γ, the zero level set of $d(\Gamma,\mathbf{x})$, is a curve in $\mathbb{R}^2$ of class C^2 that can be parametrized by $\Gamma = (x(s), y(s))$, $x, y \in C^2[s1, s2]$ with the curvature $\kappa(s)$ defined by*

$$\kappa(s) = \frac{x'(s)y''(s) - x''(s)y'(s)}{q(s)^3}, \quad q(s) = \sqrt{x'(s)^2 + y'(s)^2} \neq 0. \tag{59}$$

Assume also that

$$\omega \max_s |\kappa(s)| < 1, \tag{60}$$

and that $F(\mathbf{x})$ is a smooth function. Then, the analytical error for the integration of $\delta(d(\Gamma,\mathbf{x}))F(\mathbf{x})$ made when replacing $\delta(d(\Gamma,\mathbf{x}))$ by $\delta_\varepsilon(d(\Gamma,\mathbf{x}))$ is given by

$$E_{\omega,F}(\delta_\varepsilon) = -\varepsilon^\alpha C_{\alpha,F} \int_{-p}^{p} \varphi(\xi)\xi^\alpha d\xi + \mathcal{O}(\varepsilon^{\alpha+1}), \tag{61}$$

with

$$C_{\alpha,F} = \frac{1}{\alpha!}\int_{s_1}^{s_2} q(s)f_{\alpha t}(s,0)ds - \frac{1}{(\alpha-1)!}\int_{s_1}^{s_2} q(s)\kappa(s)f_{(\alpha-1)t}(s,0)ds. \tag{62}$$

A proof of this theorem for $p = 1$ is given in Ref. [12] but a generalization is straightforward. The parametrization of $\Gamma = (x(s), y(s))$ and the normal vector of the curve defined by

$$\mathbf{n} = \frac{(-y'(s), x'(s))}{q(s)} \tag{63}$$

are used to parametrize the integration domain Ω_ω. Introducing the parametrization $X^j(s,t) = x(s) + tn^1(s)$, $Y^j(s,t) = y(s) + tn^2(s)$. The integration can be performed over $[r_1, r_2] \times [s_1, s_2] \times [-\omega, \omega]$ when the condition in equation (60) is fulfilled. The function f in equation (62) is defined by

$$f(s,t) = F(X^j(s,t), Y^j(s,t)). \tag{64}$$

Theorem 5.2. *Let δ_ε be a continuous function with support in $[-\omega, \omega]$, $\omega = p\varepsilon$ that satisfies α continuous moment conditions, see Definition 5.1. Assume that Γ, the zero level set of $d(\Gamma, \mathbf{x})$, is a 2–manifold in $\mathbb{R}^3$ of class C^2. Suppose that $P^j(r,s) = (x(r,s), y(r,s), z(r,s)) : (r_1, r_2) \times (s_1, s_2) \to V_j$ is a coordinate patch on Γ of class C^2 and Γ is covered by the disjoint union of the open sets $V_1, \cdots, V_l$ and a set of measure zero in Γ. Let κ_1 and κ_2 be the principal curvatures of Γ on V_j. Assume also that*

$$\omega \max(\max_{r,s} |\kappa_1(r,s)|, \max_{r,s} |\kappa_2(r,s)|) < 1, \quad q(r,s) = ||P_r^j \times P_s^j|| \neq 0, \tag{65}$$

and that $F(\mathbf{x})$ is a smooth function. Then, the analytical error for the integration of $\delta(d(\Gamma, \mathbf{x}))F(\mathbf{x})$ made when replacing $\delta(d(\Gamma, \mathbf{x}))$ by $\delta_\varepsilon(d(\Gamma, \mathbf{x}))$ is given by

$$E_{\omega,F}(\delta_\varepsilon) = -\varepsilon^\alpha \int_{-p}^{p} \varphi(\xi)\xi^\alpha d\xi \sum_{j=1}^{l} C_{\alpha,F}^j + \mathcal{O}(\varepsilon^{\alpha+1}), \tag{66}$$

with

$$\begin{aligned} C_{1,F}^j = &\int_{r_1}^{r_2} \int_{s_1}^{s_2} f_t(r,s,0) q(r,s) ds dr \\ &- \int_{r_1}^{r_2} \int_{s_1}^{s_2} f(r,s,0) q(r,s) (\kappa_1(r,s) + \kappa_2(r,s)) ds dr \end{aligned} \tag{67}$$

and for $\alpha \geq 2$

$$\begin{aligned} C_{\alpha,F}^j = &\frac{1}{\alpha!} \int_{r_1}^{r_2} \int_{s_1}^{s_2} q(r,s) f_{\alpha t}(r,s,0) ds dr \\ &- \frac{1}{(\alpha-1)!} \int_{r_1}^{r_2} \int_{s_1}^{s_2} q(r,s)(\kappa_1(r,s) + \kappa_2(r,s)) f_{(\alpha-1)t}(r,s,0) ds dr \\ &+ \frac{1}{(\alpha-2)!} \int_{r_1}^{r_2} \int_{s_1}^{s_2} q(r,s)(\kappa_1(r,s)\kappa_2(r,s)) f_{(\alpha-2)t}(r,s,0) ds dr. \end{aligned} \tag{68}$$

A proof can be found in the Appendix. In order to perform the integration over Ω_ω we parametrize this region using the local parametrization of Γ and the normal vectors defined as

$$\mathbf{n} = (n^1, n^2, n^3) = \frac{1}{q(r,s)}(P_r^j \times P_s^j). \tag{69}$$

Introducing the parametrization $X^j(r,s,t) = x(r,s) + tn^1(r,s)$, $Y^j(r,s,t) = y(r,s) + tn^2(r,s)$, and $Z^j(r,s,t) = z(r,s) + tn^3(r,s)$ we can cover the domain Ω_ω by disjoint union of open sets $M_1, \cdots, M_l$ and a set of measure zero in Ω_ω, where

$$\begin{aligned} M_j = \{(x,y,z) : x = X^j(r,s,t),\ y = Y^j(r,s,t),\ z = Z^j(r,s,t), \\ r \in (r_1, r_2),\ s \in (s_1, s_2),\ t \in [-\omega, \omega]\}. \end{aligned} \tag{70}$$

The condition in equation (65) guarantees that this parametrization is nonsingular. The integration can then be performed over $[r_1, r_2] \times [s_1, s_2] \times [-\omega, \omega]$. For ω small one can Taylor expand

$$f(r,s,t) = F(X^j(r,s,t), Y^j(r,s,t), Z^j(r,s,t)) \tag{71}$$

around $(r,s,0)$ and express the analytical error in terms of the moments of the function δ_ε.

In the next section we analyze the numerical error made using the trapezoidal rule for integration.

5.2 Numerical error

The following theorem gives the error of the trapezoidal rule in one dimension.

Theorem 5.3. *Let*

$$x_n = a + nh, \quad n = 0, \cdots N, \quad h = \frac{b-a}{N} \tag{72}$$

be a decomposition of the interval $[a,b]$ and $T_h(a,b,h,\psi)$ be the trapezoidal sum

$$T_h(a,b,h,\psi) = \sum_{n=0}^{N} w_n \psi(x_n), \tag{73}$$

where

$$w_n = \begin{cases} 1/2, & \text{for } n = 0, \text{ and } n = N, \\ 1, & \text{otherwise.} \end{cases} \tag{74}$$

Assume that $\psi(x) \in C^{2r+2}(a,b)$. Then

$$T_h - \int_a^b \psi(x)dx = R_T(a,b,h,\psi(x)) \tag{75}$$

with

$$R_T(a,b,h,\psi(x)) = \sum_{k=1}^{r} \frac{B_{2k}h^{2k}}{(2k)!}\ \psi^{2k-1}(x)\Big|_{x=a}^{b} + R_{2r+2}(a,b,h,\psi), \qquad (76)$$

where B_j are the Bernoulli numbers and $R_{2r+2}(a,b,h,\psi)$ is $\mathcal{O}(h^{2r+2})$.

For a proof see Ref. [17] p. 298. Note that when the function $\psi(x) \in C^\infty$ for $x \in \mathbb{R}$ and ψ has $[a,b]$ as an interval of periodicity, then

$$\psi^{(k)}(b) = \psi^{(k)}(a), \quad k = 0, 1, 2, \cdots . \qquad (77)$$

Hence,

$$|R_T(a,b,h,\psi)| = \mathcal{O}(h^{2r+2}) \qquad (78)$$

for arbitrary r. Therefore, we have that for periodic infinite differentiable functions the trapezoidal error tends to zero faster than any power of h, as $h \to 0$. This is referred to as superconvergence. In Ref. [18] another proof is given. It is shown by using the Poisson summation formula that the error

$$R_T = \sum_{n=0}^{N} w_n \psi(x_n) - \int_a^b \psi(x)dx. \qquad (79)$$

decreases as $\hat{\psi}(1/h)$, with

$$\hat{\psi}(\omega) = \int_{-\infty}^{\infty} \psi(x) e^{-2\pi i k x} dx. \qquad (80)$$

If $\psi \in C^r[\mathbb{R}]$ and periodic, then $\hat{\psi}(1/h) = \mathcal{O}(h^r)$, as $h \to 0$. Thus, for $\psi \in C^\infty[\mathbb{R}]$ the trapezoidal rule converges faster than any power of h.

In higher dimensions we use the notion of a product rule. For simplicity we do the analysis here in two dimensions. The analysis in three dimensions is similar. Let $\Omega = [a,b] \times [c,d]$, and $\psi(x,y) \in C^{2r+2}(\Omega)$. Introduce a uniform grid

$$\begin{aligned} x_j &= a + jh_x, \quad j = 0, \cdots M, \quad h_x = \frac{b-a}{N}, \\ y_n &= c + nh_y, \quad n = 0, \cdots N, \quad h_y = \frac{d-c}{N}. \end{aligned} \qquad (81)$$

Denote by Q the quadrature scheme obtained by using the trapezoidal rule in both x and y directions with step size h_x and h_y. We can write

$$I = \int\int_\Omega \psi(x,y)dxdy = \int_c^d g(y)dy, \qquad (82)$$

where

$$g(y) = \int_a^b \psi(x, y)dx. \tag{83}$$

Using the trapezoidal rule to integrate in the y–variable (see, equation (73) and (74)) gives

$$I = h_y \sum_{n=0}^{N} w_n \int_a^b \psi(x, y_n)dx + R_T(c, d, h_y, g(y)), \tag{84}$$

where R_T is the quadrature error. Using also the trapezoidal method in the x–direction with step size h_x yields

$$\begin{aligned} I = h_y \sum_{n=0}^{N} w_n \left(h_x \sum_{j=0}^{M} w_j \psi(x_j, y_n) + R_T(a, b, h_x, \psi(x, y_n)) \right) \\ + R_T(c, d, h_y, g(y)). \end{aligned} \tag{85}$$

Simplifying the expression we get

$$\begin{aligned} I = h_y h_x \sum_{n=0}^{N} \sum_{j=0}^{M} w_n w_j \psi(x_j, y_n) + \sum_{n=0}^{N} h_y w_n R_T(a, b, h_x, \psi(x, y_n)) \\ + R_T \left(c, d, h_y, \int_a^b \psi(x, y)dx \right). \end{aligned} \tag{86}$$

Hence

$$\begin{aligned} \left| I - h_y h_x \sum_{n=0}^{N} \sum_{j=0}^{M} w_n w_j \psi(x_j, y_n) \right| \leq (d - c) \max_{y_n} |R_T(a, b, h_x, \psi(x, y_n))| \\ + (b - a) \max_{x \in [a,b]} |R_T(c, d, h_y, \psi(x, y))|, \end{aligned} \tag{87}$$

and we see that the convergence results from one dimension extend to two dimensions. Thus, the superconvergence of the trapezoidal rule also applies in two dimensions. We are now able to formulate the following theorem.

Theorem 5.4. *Let $\Omega = [a, b] \times [c, d]$, and Q be the quadrature scheme obtained by using the trapezoidal rule in both x and y directions with step size h_x and h_y. Suppose δ_ε has compact support in $[-\omega, \omega]$ and $\Omega_\omega \subset \Omega$ where*

$$\Omega_\omega = \{\mathbf{x} : |d(\Gamma, \mathbf{x})| \leq \omega\}. \tag{88}$$

Suppose further that $\delta_\varepsilon(d(\Gamma,\mathbf{x}))F(\mathbf{x}) \in C^\infty(\Omega)$. *Then the numerical error*

$$E_{Q,F}(\delta_\varepsilon) = \left| \int_\Omega \delta_\varepsilon(d(\Gamma,\mathbf{x}))F(\mathbf{x})d\mathbf{x} - Q(\delta_\varepsilon(d(\Gamma,\mathbf{x}))F(\mathbf{x})) \right| \tag{89}$$

decreases faster than any power of $h = \max(h_x, h_y)$.

When δ_ε has compact support in $[-\omega,\omega]$ the approximation $\delta_\varepsilon(d(\Gamma,\mathbf{x}))$ has compact support in Ω_ω. As long as $\Omega_\omega \subset \Omega$, $\delta_\varepsilon(d(\Gamma,\mathbf{x}))$ is a periodic function on Ω. Since the integrand is $C^\infty(\Omega)$ the superconvergence of the trapezoidal rule gives the result of the theorem.

The same result also holds in three dimensions.

5.3 Practical considerations

The theorems in the previous section are applicable to delta function approximations with compact support in $[-\omega,\omega]$. In order to obtain the superconvergence of the trapezoidal rule discussed in the previous section, the delta function approximations must also be infinitely differentiable. All the delta function approximations presented in Section 4 are infinitely differentiable but do not have compact support. To be able to apply the theorems of the previous section, we will truncate the delta function approximation and set it to zero outside some $[-\omega,\omega]$ interval. This is motivated by the fact that the delta function approximations in question decay exponentially fast. The truncation results in an integration error. In the following, we show that it is possible to select an ω so that this integration error is negligibly small. For such ω, the delta function approximation essentially has compact support.

For the analytical error to be small and the conditions in Theorems 5.1 and 5.2 to be satisfied ω needs to be small. Also, note that since the conditions in equation (36), Theorem 4.1 is equivalent to the continuous moment conditions i.e.

$$\left.\frac{\partial^r \hat{\varphi}(k)}{\partial k^r}\right|_{k=0} = \begin{cases} 1, & \text{for } r=0, \\ 0, & \text{for } 1 \le r < q \end{cases} \Leftrightarrow \int_{-\infty}^{\infty} \varphi(x) x^r dx = \begin{cases} 1, & \text{for } r=0, \\ 0, & \text{for } 1 \le r < q. \end{cases} \tag{90}$$

all the delta function approximations in Section 4 satisfy two continuous moment conditions. Therefore we expect them to be second order accurate.

Split the integration over the whole domain into three parts:

$$\int_{-\infty}^{\infty} \delta_\varepsilon(t)f(t)dt = \int_{-\omega}^{\omega} \delta_\varepsilon(t)f(t)dt + \int_{-\infty}^{-\omega} \delta_\varepsilon(t)f(t)dt + \int_{\omega}^{\infty} \delta_\varepsilon(t)f(t)dt. \tag{91}$$

Let $\omega = p\varepsilon$. Then the error we make by not having a delta function approximation with compact support in $[-\omega,\omega]$ is the sum of

$$I_1 = \int_{\omega}^{\infty} \delta_\varepsilon(t)f(t)dt = \int_{p}^{\infty} \varphi(t)f(\varepsilon t)dt \tag{92}$$

and

$$I_2 = \int_{-\infty}^{\omega} \delta_\varepsilon(t) f(t) dt = \int_{-p}^{\infty} \varphi(t) f(\varepsilon t) dt. \tag{93}$$

Since $\delta_\varepsilon^{FD}(t)$ decays slower then the other approximations introduced in Section 4, see Fig. 5(a), we obtain the largest values for I_1 and I_2 when $\delta_\varepsilon^{FD}(t)$ is used. Therefore, we do the following analysis only for $\delta_\varepsilon = \delta_\varepsilon^{FD}(t)$. We have that

$$|I_1| \leq \left| \int_p^{\infty} e^{-t} f(\varepsilon t) dt \right| \tag{94}$$

and

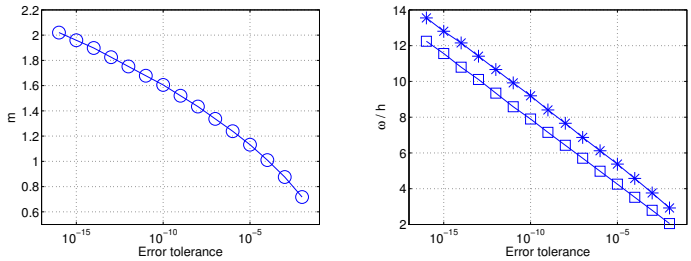
$$|I_2| \leq \left| \int_{-\infty}^{-p} \frac{f(\varepsilon t)}{1 + e^{-t}} dt \right|. \tag{95}$$

We assume that $f(t)$ is either a bounded function or a polynomial. For functions $f(t)$ that increase exponentially the delta function approximations considered in this paper may fail to give accurate results. They are generally not able to cancel such $f(t)$ outside the interval. For $f(t) = t^n$ we get

$$\left| \int_p^{\infty} e^{-t} (\varepsilon t)^n dt \right| = \left| \sum_{j=0}^{n} c_j \varepsilon^n p^j e^{-p} \right| \leq M e^{-p}, \tag{96}$$

where M is a bounded constant and depends on ε^n. Now, given M one can choose $\omega = p\varepsilon$ to be such that Me^{-p} is within some error tolerance. However, the computational domain must be large enough so that $[-\omega, \omega]$ is included in the domain.

For the delta function approximations δ_ε^{FD} and δ_ε^{G} that we introduced in Section 4 we suggest here a way to select $\varepsilon = mh$ and the half width support where the functions can be truncated so that the approximations have compact support and are second order accurate down to a specified error tolerance. Given a tolerance C, choose m such that $\hat{\varphi}(m) = C$. This gives a delta function approximation δ_{mh} that is second order accurate down to the specified error tolerance. We then define the half width support ω of this delta approximation so that $\delta_\varepsilon(\omega) = C$ with $\varepsilon = mh$. In Figs. 6 and 7 we show m and ω/h for different error tolerances for δ_ε^{G} and δ_ε^{FD} respectively. The widths for the different functions is in agreement with their decay behaviors, see Fig. 5. The Gaussian function decay much faster than the derivative of the Fermi–Dirac function. For example for a tolerance of 10^{-6} the optimal m for the Gaussian approximation is around 1.25. The function δ_{mh}^{G} then has a half width ω around $6h$ for an error tolerance of 10^{-6}. For the same tolerance, m is somewhat smaller than 1 for the derivative of the Fermi–Dirac function but the half width of δ_{mh}^{FD} is $25h$. The difference between the width of the Gaussian approximation and the derivative of the Fermi–Dirac function



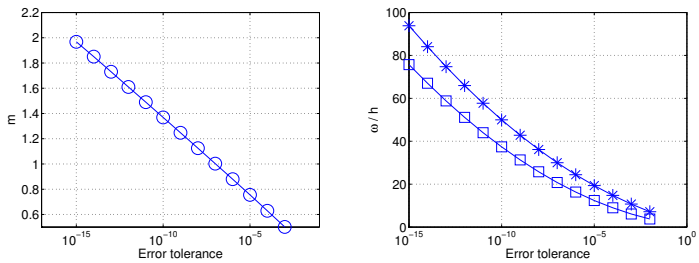
(a) (b)

Figure 6: For the delta function δ^G_{mh} we show in Panel (a) m as a function of error tolerance. We show the smallest m such that $\hat{\varphi}^G(m)$ is below the given error tolerance. This means that δ^G_{mh} is a second order accurate approximation of the delta function down to the specified error tolerance. In Panel (b) we show the half width support of δ^G_{mh} divided by h when m from Panel (a) is used. The support is computed with the same error tolerance that was used to find the appropriate m–value. This calculation was made for two different values of h, $h = 10^{-6}$ (stars) and $h = 10^{-2}$ (squares) and gives an idea of where the delta function approximation can be truncated. For a tolerance of 10^{-6} the optimal m is, for example, around 1.25. The width of the delta approximation down to the same tolerance is around $6h$, while for a requested tolerance of 10^{-16}, $m \approx 2$ and the width of the approximation is around $14h$.

is larger for smaller error tolerances. In practical computations the Gaussian function is preferable.

6 Numerical results

In this section, we present three numerical examples using the distance function to extend the one–dimensional regularized delta functions δ^{FD}_ε, δ^G_ε, and δ^{TE}_ε to higher dimensions. We also show results when non–distance functions are used. Here, we study the rate of convergence numerically by mesh refinement. In all the examples in this section we have integrands with non–vanishing second derivatives and $\varepsilon = mh$. Since the approximations δ^{TE}_ε, δ^{FD}_ε, and δ^G_ε all are of continuous moment order 2, we expect to have a second order analytical error in both two and three dimensions, according to theorems 5.1 and 5.2.



(a) (b)

Figure 7: The delta function δ^{FD}_{mh} is used. This figure should be compared with Fig. 6. In Panel (a) we show the smallest m such that $\hat{\varphi}^G(m)$ is below the given error tolerance. This means that δ^{FD}_{mh} is a second order accurate approximation of the delta function (since 2 moment conditions are satisfied) down to the specified error tolerance. In Panel (b) we show the half width support divided by h of δ^{FD}_{mh} when m from Panel (a) is used. The support is computed with the same error tolerance that was used to find the appropriate m–value. This calculation was made for two different values of h, $h = 10^{-6}$ (stars) and $h = 10^{-2}$ (squares) and gives an idea of where the delta function approximation can be truncated. Here, for a tolerance of 10^{-6} the optimal m is somewhat smaller than 1. The width of the delta approximation up to the same tolerance is around $25h$. Note that this is much larger than the width of the Gaussian approximation which is around $6h$ for the same tolerance, see Fig. 6.

Example 1 Consider the problem of computing the line integral

$$I = \int_\Gamma 3x^2 - y^2 ds = 2\pi, \tag{97}$$

where Γ is a circle of radius 1 centered at the origin. This problem has previously been considered by Smereka, see Table 3 in Ref. [7]. We cover the domain $\Omega = \{\mathbf{x} = (x, y) \mid |x| \leq 2,\ |y| \leq 2\}$ with a regular grid

$$x_i = -2 + ih, \quad i \in \mathbb{Z}, \tag{98}$$

$$y_j = -2 + jh, \quad j \in \mathbb{Z}, \tag{99}$$

and approximate the line integral I by

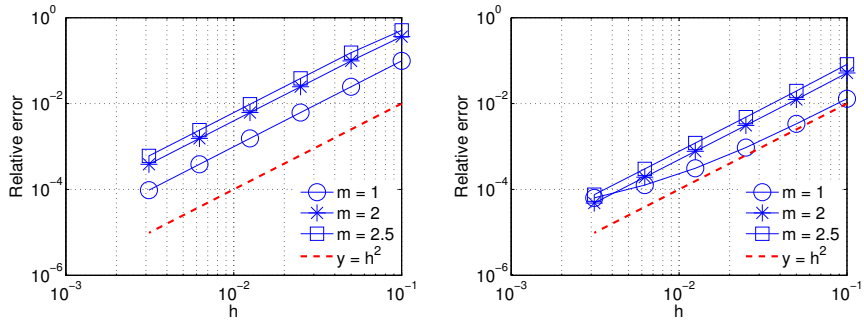
$$I_h = h^2 \sum_{j\in\mathbb{Z}} \sum_{i\in\mathbb{Z}} (3x_i^2 - y_j^2)\delta_{mh}^{FD}(\phi(\Gamma, (x_i, y_j))). \tag{100}$$

In Fig. 8 the relative error $E = |I_h - I|/I$ is shown for $m = 1$, 2, and 2.5. In Fig. 8(a) the level function $\phi(\Gamma, \mathbf{x}) = d(\Gamma, \mathbf{x})$ and we see second order convergence. The mass error in the case when $m = 1$ is of order 10^{-7} (see Fig. 6) and cannot be seen in the plot. We have also used the level set function $\phi(\Gamma, \mathbf{x}) = x^2 + y^2 - 1$ as in Ref. [7], which is not a signed distance function. The results are shown in Fig. 8(b) and indicate second order convergence for $m \geq 2$. In the case of a non–distance function the mass error increases as expected (see the Remark in Section 3). For $m = 1$ the curve representing the error (circles in Fig. 8(b)) flattens out as h decreases since the total error is then dominated by the mass error.

Example 2 Here, we consider the computation of the surface integral:

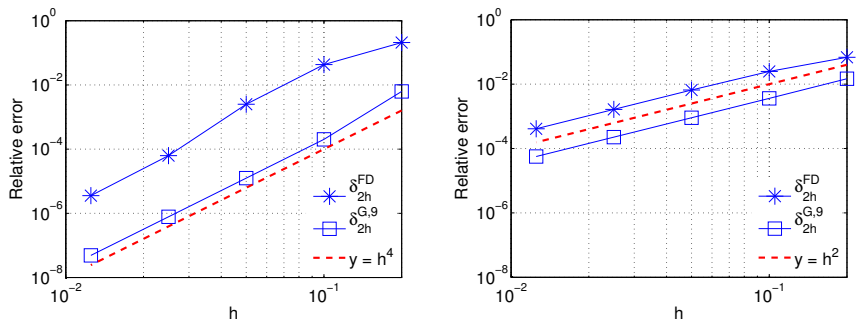
$$I = \int_\Gamma (4 - 3x^2 + 2y^2 - z^2) dA = \frac{40\pi}{3}, \tag{101}$$

where Γ is a sphere of radius 1 centered at the origin. In Refs. [7, 9] the level function $u(\Gamma, \mathbf{x}) = x^2 + y^2 + z^2 - 1$ is used to extend one–dimensional regularized delta functions to three dimensions. The relative error using $\delta_{2h}^{FD}(u(\Gamma, \mathbf{x}))$ and $\delta_{2h}^{G}(u(\Gamma, \mathbf{x}))$ is shown in Fig. 9(b), where second order convergence can be seen. In Fig. 9(a) we use the distance function and observe that the convergence is faster than second order. Using spherical coordinates to parametrize the sphere Γ one can show that the constant $C_{2,F}$ in the analytical error defined in Theorem 5.2 (equation (68)) with $F = 4 - 3x^2 + 2y^2 - z^2$ is zero. Further, since both delta approximations used here satisfy all the odd moment conditions, the convergence is of fourth order, in accordance with Theorem 5.2. The results are similar when the center of the sphere is shifted. The relative error is smaller when $\delta_{2h}^{G}(u(\Gamma, \mathbf{x}))$ is used compared to $\delta_{2h}^{FD}(u(\Gamma, \mathbf{x}))$. This is also in accordance with Theorem 5.2 since the width ω of δ_{2h}^{G} is much smaller than the width of δ_{2h}^{FD} (see Section 5.3). Consequently, the constant in the analytical error is smaller.



(a) (b)

Figure 8: The relative error $E = |I_h - I|/I$ where I and I_h are defined in equation (97) and (100), respectively. We have used δ_{mh}^{FD} with $m = 1$ (circles), $m = 2$ (stars), and $m = 2.5$ (squares). The dashed line is $y = h^2$. In Panel (a) the level function is the signed distance function. In Panel (b) the level function $\phi(\Gamma, \mathbf{x}) = x^2 + y^2 - 1$ is used.



(a) (b)

Figure 9: The relative error in the computation of the surface integral in equation (101) when Γ is a sphere of radius 1 centered at the origin. The results are for $\delta_{2h}^{FD}(u(\Gamma, \mathbf{x}))$ (stars) and $\delta_{2h}^{G}(u(\Gamma, \mathbf{x}))$ (squares). In Panel (a) the level function is the signed distance function. In Panel (b) the level function $u(\Gamma, \mathbf{x}) = x^2 + y^2 + z^2 - 1$ is used.

6.1 Partial differential equations

We consider now the differential equation,

$$\begin{aligned} Lu &= \delta(\Gamma, g, \mathbf{x}), \quad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \\ Bu &= r(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega. \end{aligned} \tag{102}$$

The solution can be written as

$$u(\mathbf{x}) = \int_\Omega G(\mathbf{x}, \mathbf{y}) \delta(\Gamma, g, \mathbf{y}) d\mathbf{y} + R(\mathbf{x}), \tag{103}$$

where $G(\mathbf{x}, \mathbf{y})$ is Green's function and $R(\mathbf{x})$ represents the contribution from the boundary conditions. In the case of homogeneous boundary conditions $R(\mathbf{x}) = 0$. In the computations the delta function is approximated by a regularized delta function $\delta_\varepsilon(\Gamma, g, \mathbf{x})$ with half width support ω. Assume that Green's function $G(\mathbf{x}, \mathbf{y})$ is regular away from $\mathbf{x} = \mathbf{y}$ for all $\mathbf{y} \in \Gamma$. Then, for all $\mathbf{x}_j$ for which $|\mathbf{x}_j - \mathbf{x}| > \omega$ for all $\mathbf{x} \in \Gamma$,

$$|u_j - u(\mathbf{x}_j)| \leq C h^{\min(p,q)}, \tag{104}$$

where q is the order of accuracy of the delta function approximation and p is the order of accuracy for the discretization of the differential operator L. For a proof see Ref. [3].

Example 3 Let us consider the Poisson equation in $\mathbb{R}^2$

$$\begin{aligned} -\Delta u &= \delta(\Gamma, \mathbf{x}), \quad \mathbf{x} \in \Omega \subset \mathbb{R}^2 \\ u(\mathbf{x}) &= v(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega \end{aligned} \tag{105}$$

where $\Omega = \{\mathbf{x} = (x^{(1)}, x^{(2)});\ |x^{(1)}| \leq 1, |x^{(2)}| \leq 1\}$, $\Gamma = \{\mathbf{x}, |\mathbf{x} - \hat{\mathbf{x}}| = 1/2\}$, and $v(\mathbf{x}) = 1 - \log(2|\mathbf{x} - \hat{\mathbf{x}}|)/2$. The solution of this equation is

$$u(\mathbf{x}) = \begin{cases} 1 & |\mathbf{x} - \hat{\mathbf{x}}| \leq 1/2, \\ 1 - \log(2|\mathbf{x} - \hat{\mathbf{x}}|)/2, & |\mathbf{x} - \hat{\mathbf{x}}| > 1/2, \end{cases} \tag{106}$$

see Fig. 10. We introduce a uniform grid, with step size $h = 2/N$ in both $x^{(1)}$ and $x^{(2)}$ direction. The delta function approximations δ_ε^{FD}, δ_ε^{G}, δ_ε^{TE}, and δ_ε^{C} are tested for $\varepsilon = mh$. We use a fourth order stencil D_2^4 to approximate the differential operator. See [3] for the definition of D_2^4. The error,

$$||u - u_h|| \tag{107}$$

is measured in both the maximum norm and the L1–norm. Here, u is the exact solution given in equation (106) and u_h is the numerical solution. In Figs. 11 and 12 we show the error when the circle Γ is centered in $\hat{x} = (0, 0)$.

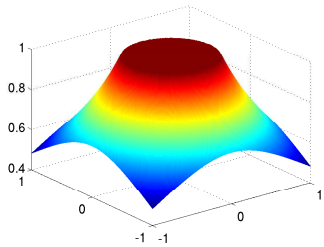


Figure 10: The exact solution $u(\mathbf{x})$ given by equation (106) for $\hat{\mathbf{x}} = 0$.

In Fig. 11 the maximum norm measured over the whole domain Ω when δ_{mh}^{FD} and δ_{mh}^{G} are used is shown for $m = 1$, $m = 2$, and $m = 3$. We have first order convergence since we measured the error close to Γ. To measure the error away from Γ, we introduce the sub–domain

$$\widetilde{\Omega} = \{\mathbf{x} : \mathbf{x} \in \Omega,\ |d(\Gamma, \mathbf{x})| > \eta\}. \tag{108}$$

Since $\delta_{\varepsilon}^{FD}$, δ_{ε}^{G}, and $\delta_{\varepsilon}^{TE}$ are all second order accurate we expect from equation (104) to see second order convergence when the error is measured away from Γ. We recall that analytical results suggested that $\eta \geq \omega$ is needed to obtain the convergence order of the delta function approximation, see equation (104). However, our numerical simulations indicate that $\eta = \varepsilon$ is sufficient. In Fig. 12 the maximum norm and the $L1$-norm of the error is shown for δ_{2h}^{FD}, δ_{2h}^{G}, δ_{2h}^{TE}, and δ_{2h}^{C}. For the regularized delta function δ_{2h}^{C} of order h^4 in one dimension there is no convergence, neither in the maximum norm nor in the L1–norm. Note that the error using δ_{2h}^{G} is almost identical to the error we obtain using δ_{2h}^{TE}.

We obtained similar results for other values of $\hat{x}$ away from the boundary of the computational domain.

7 Conclusions

We have introduced delta function approximations that are convenient to use for delta functions with support on a curve or a surface, represented implicitly by a level set. The framework is based on the "old" method with a one–dimensional delta function approximation extended to higher dimensions by a distance function.

This method was in [3] shown to be inconsistent when different compact one–dimensional delta function approximations were used. In this paper, we

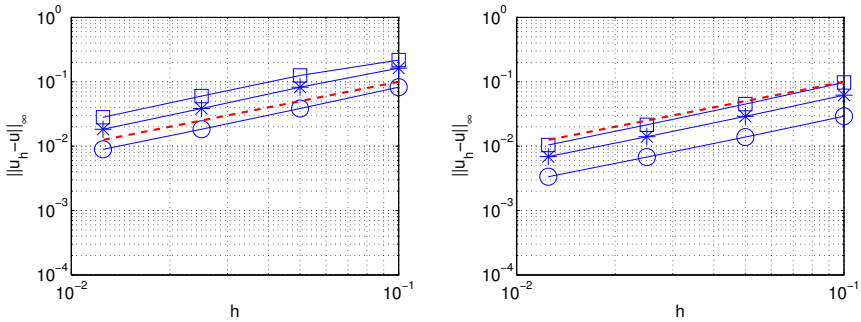


Figure 11: The maximum norm of the error measured over Ω is shown for δ_{mh}, $m = 1$ (circles), $m = 2$ (stars), and $m = 3$ (squares). A fourth order discretization for the differential operator has been used. The dashed line is $y = h$. In Panel (a) $\delta_{mh} = \delta^{FD}_{mh}$ is used. In Panel (b) $\delta_{mh} = \delta^{G}_{mh}$ is used. As expected, we see a first order error, since we have measured the error also close to Γ.

have shown that this can be understood from the fact that these compact functions cannot both satisfy the discrete mass condition for all shifts in the grid *and* for a range of dilations of the support. This is however possible if one is basing the approximation on functions that have compact support instead in Fourier space, or in practice, that are smaller than some tolerance outside a given interval. For such functions, we have proven in both two and three dimensions that the error can be bounded by the sum of the analytical and the numerical error. The analytical error is determined by the moment order of the one–dimensional approximation and the numerical error tends to zero faster than any power of h in the limit as $h \to 0$, due to the superconvergence of the trapezoidal rule. All the three functions we have discussed have second order analytical errors. When we are to compute an integral over the delta function itself, yielding length of curve or surface area, or over the delta function multiplied by a linear function, there is no analytical error for any of the approximations that we have introduced, since they are all of moment order 2.

A function that is compact or decays rapidly in Fourier space will not produce delta function approximations with compact support. This means that in practice, we need to truncate these approximations. For the Gaussian function δ^G_ε, that was introduced in Section 4.2, the accuracy of this procedure was discussed in conjunction with Fig. 6. It was concluded that to achieve an

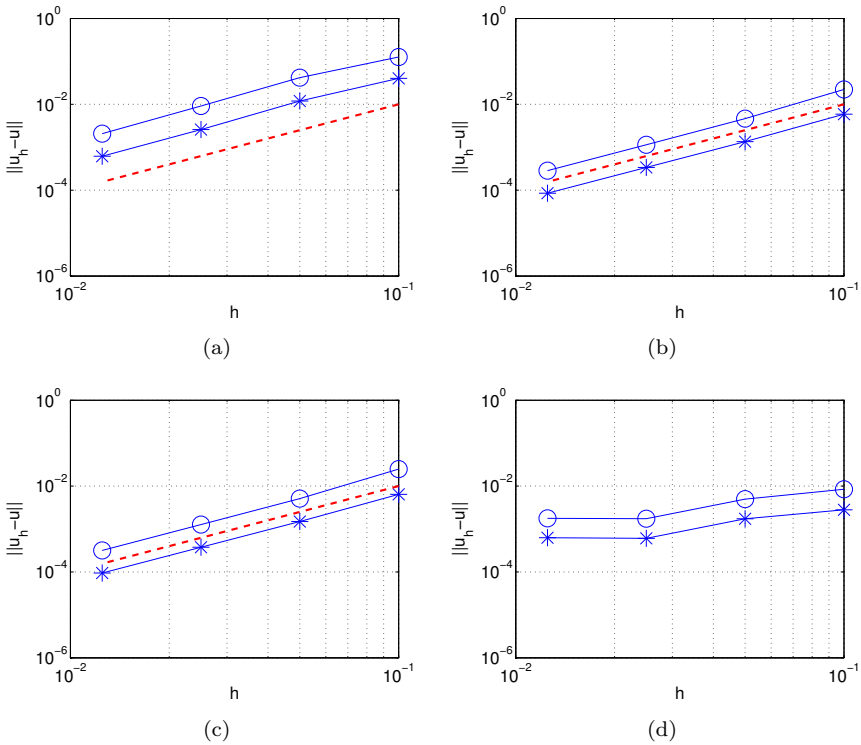


Figure 12: The error measured over $\widetilde{\Omega}$ (this domain is defined in (108) with $\eta = 0.2$) using a fourth order discretization of the differential operator. The max–norm of the error (circles) and the L1–norm (stars) are shown. The dashed line is $y = h^2$. In Panel (a) $\delta_\varepsilon = \delta_{2h}^{FD}$. In Panel (b) $\delta_\varepsilon = \delta_{2h}^{G}$. In Panel (c) $\delta_\varepsilon = \delta_{2h}^{TE}$. In Panel (d) $\delta_\varepsilon = \delta_{2h}^{C}$. The second order convergence is now obtained in Panel (a), (b), and (c), as we are excluding the region closest to Γ, as explained in the text.

error around 10^{-6} we need $\varepsilon \geq mh$, $m = 1.25$. For $m = 1.25$ the Gaussian function can be truncated so that the half width support becomes around $6h$. To get an error around 10^{-16}, we need $m = 2$ and a half width support of around $14h$. When we have non–vanishing second derivatives of the function F, see (55), there will be a second order analytical error which will dominate the numerical error, and for moderate grid sizes, there is no point in using a wider support than $6h$.

In this paper, we also discussed the function δ_ε^{TE}, as defined in (42). This function yields very similar results to the Gaussian, but is more computationally expensive, since the function is not given explicitly but must be computed from its Fourier transform. We found that, compared to the other approximations, the approximation δ_ε^{FD}, has a slower decay in real space and hence a larger support. We therefore recommend to use the Gaussian approximation.

Acknowledgement

S. Z. thanks Gunilla Kreiss and Emanuel Rubensson for valuable comments and helpful discussions during the preparation of the manuscript.

8 Appendix

We start by stating a definition and a theorem from Ref. [19] that will be used in the study of the analytical error in three dimensions.

Definition 8.1. *Let $k > 0$. A k–manifold in $\mathbb{R}^n$ of class C^r is a subspace M of $\mathbb{R}^n$ having the following property: For each $p \in M$, there is an open set V of M containing p, a set U that is open in either $\mathbb{R}^k$ or the upper half–space in $\mathbb{R}^k$ and a continuous map $P : U \to V$ carrying U onto V in one–to–one fashion, such that :*

1. *P is of class C^r .*
2. *$P^{-1} : V \to U$ is continuous.*
3. *$DP(\mathbf{x})$, the Jacobian matrix of P, has rank k for each $\mathbf{x} \in U$.*

The map P is called a coordinate patch on M about p.

Theorem 8.1. *Let M be a compact k–manifold in $\mathbb{R}^n$, of class C^r. Let $f : M \to \mathbb{R}$ be a continuous function. Suppose that $P_i : A_i \to M_i$, for $i = 1, \cdots, N$ is a coordinate patch on M, such that A_i is open in $\mathbb{R}^k$ and M is*

the disjoint union of the open sets $M_1, \cdots, M_N$ *of* M *and a set* K *of measure zero in* M*. Then*

$$\int_M f dV = \sum_{i=1}^{N} \int_{A_i} (f \circ P_i) V(DP_i) \tag{109}$$

This theorem states that $\int_M f dV$ can be evaluated by separately evaluating the integral over local parametrized parts of the manifold and then summing up all the contributions. A proof of the theorem can be found in Ref. [19]. It is assumed that the support of the integrand f lies in M.

Proof of Theorem 5.2. In three dimensions we cannot expect to have a global parametrization but a local exist. By the assumption Γ is a 2–manifold that can be covered by an union of disjoint open sets $V_1, \cdots, V_l$ and a set K of measure zero in Γ. It has been proven that such sets can be constructed using polygonal charts sets, see [20]. Further, we have assumed that a coordinate patch $P^j = (x(r,s), y(r,s), z(r,s)) : (r_1, r_2) \times (s_1, s_2) \to V_j$ of class C^2 on Γ exists.

The normal of Γ at V_j is defined as $\mathbf{n} = (n^1, n^2, n^3) = \frac{1}{q(r,s)}(P^j_r \times P^j_s)$, where $q(r,s) = ||P^j_r \times P^j_s|| \neq 0$.

The integration is over Ω_ω. This is a compact 3-manifold of class C^2. The integrand $\delta_\varepsilon F : \Omega_\omega \to \mathbb{R}$, is a continuous function. The domain Ω_ω can be covered by the disjoint union of open sets $M_1, \cdots, M_l$ and a set of measure zero in Ω_ω. The open set M_j can be given by the following parametrization

$$\begin{aligned} M_j = \{(x,y,z) : x = X^j(r,s,t),\ y = Y^j(r,s,t),\ z = Z^j(r,s,t), \\ r \in (r_1, r_2),\ s \in (s_1, s_2),\ t \in [-\omega, \omega]\}, \end{aligned} \tag{110}$$

where $X^j(r,s,t) = x(r,s) + tn^1(r,s)$, $Y^j(r,s,t) = y(r,s) + tn^2(r,s)$, and $Z^j(r,s,t) = z(r,s) + tn^3(r,s)$. Let $A_j = (r_1, r_2) \times (s_1, s_2) \times [-\omega, \omega]$, and

$$\beta^j = (X^j(r,s,t), Y^j(r,s,t), Z^j(r,s,t)). \tag{111}$$

Then it follows from Theorem 8.1 that

$$I_{\Omega_\omega}(\delta_\varepsilon F) = \sum_{j=1}^{l} I_{A_j} = \sum_{j=1}^{l} \int_{A_j} (\delta_\varepsilon \circ \beta^j)(F \circ \beta^j) V(D\beta^j), \tag{112}$$

where $V(D\beta^j) = |\det(J(r,s,t))| dt ds dr$. The Jacobian determinant for this transformation from (x,y,z) to (r,s,t) is

$$\begin{aligned} \det(J(r,s,t)) &= (X^j_r, Y^j_r, Z^j_r) \times (X^j_s, Y^j_s, Z^j_s) \cdot (X^j_t, Y^j_t, Z^j_t) \\ &= (P^j_r \times P^j_s + t(\mathbf{n}_r \times P^j_s + P^j_r \times \mathbf{n}_s) + t^2(\mathbf{n}_r \times \mathbf{n}_s)) \cdot \mathbf{n} \\ &= ||P^j_r \times P^j_s||(1 - t(\kappa_1 + \kappa_2) + t^2 \kappa_1 \kappa_2) \\ &= q(r,s)(1 - t\kappa_1(r,s))(1 - t\kappa_2(r,s)). \end{aligned} \tag{113}$$

Here we have used that the coordinate patch, P^j is C^2 hence $P^j_{rs} = P^j_{sr}$. This transformation is non–singular because of the assumption in equation (65)

Note that $d(\Gamma, \mathbf{x}) = t$ and denote

$$f(r,s,t) = F(X^j(r,s,t), Y^j(r,s,t), Z^j(r,s,t)). \tag{114}$$

We have

$$I_{A_j} = \int_{A_j} (\delta_\varepsilon \circ \beta^j)(F \circ \beta^j) V(D\beta^j) =$$
$$\int_{r_1}^{r_2} \int_{s_1}^{s_2} \int_{-\omega}^{\omega} \delta_\varepsilon(t) f(r,s,t) q(r,s)(1 - t\kappa_1(r,s))(1 - t\kappa_2(r,s)) dt ds dr. \tag{115}$$

The assumption that $F(\mathbf{x})$ is a smooth function yields that $f(r,s,t)$ has $N+1$ bounded derivatives with respect to t. Since $t \in [-\omega, \omega]$ we can for ω small Taylor expand $f(r,s,t)$ around $(r,s,0)$

$$f(r,s,t) = \sum_{i=0}^{N} \frac{t^i}{i!} f_{it}(r,s,0) + \mathcal{O}(t^{N+1}). \tag{116}$$

The index i in f_{it} denotes the number of partial derivatives with respect to t. Define the moments of the function $\delta_\varepsilon(t)$ as

$$M_\alpha(\delta_\varepsilon(t)) = \int_{-\omega}^{\omega} \delta_\varepsilon(t) t^\alpha dt. \tag{117}$$

Replacing $f(r,s,t)$ in equation (115) with its Taylor expansion we obtain

$$\begin{aligned} I_{A_j} &= M_0(\delta_\varepsilon(t)) \int_{r_1}^{r_2} \int_{s_1}^{s_2} f(r,s,0) q(r,s) ds dr \\ &+ M_1(\delta_\varepsilon(t)) \left(\int_{r_1}^{r_2} \int_{s_1}^{s_2} f_t(r,s,0) q(r,s) ds dr \right. \\ &\left. - \int_{r_1}^{r_2} \int_{s_1}^{s_2} f(r,s,0) q(r,s)(\kappa_1(r,s) + \kappa_2(r,s)) ds dr \right) \\ &+ \sum_{\alpha=2}^{N} C^j_{\alpha,F} M_\alpha(\delta_\varepsilon(t)) + \mathcal{O}(M_{N+1}(\delta_\varepsilon(t))), \end{aligned} \tag{118}$$

where the constant $C^j_{\alpha,F}$ is given in equation (68).

By the change of variable $t/\varepsilon = \xi$ and since $\omega = p\varepsilon$ we get

$$M_\alpha(\delta_\varepsilon(t)) = \int_{-\omega}^{\omega} \frac{1}{\varepsilon} \varphi(t/\varepsilon) t^\alpha dt = \varepsilon^\alpha \int_{-p}^{p} \varphi(\xi) \xi^\alpha dt. \tag{119}$$

By summing up contributions from all A_j we obtain the theorem. □

References

[1] S. Osher, J. A. Sethian., Fronts propagating with curvature dependent speed: Algorithms based on hamilton–jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.

[2] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer-Verlag, 2003.

[3] A.-K. Tornberg, B. Engquist, Numerical approximations of singular source terms in differential equations, J. Comput. Phys. 200 (2004) 462–488.

[4] C. S. Peskin, The immersed boundary method, Acta Numer. 11 (2002) 479–517.

[5] J. Sethian, Level Set Methods and Fast Marching Methods, Cambridge University Press, 1999.

[6] B. Engquist, A.-K. Tornberg, R. Tsai, Discretization of dirac delta functions in level set methods, J. Comput. Phys. 207 (2005) 28–51.

[7] P. Smereka, The numerical approximation of a delta function with application to level set methods, J. Comput. Phys. 211 (2006) 77–90.

[8] J. T. Beale, A proof that a discrete delta function is second–order accurate, J. Comput. Phys. 227 (2008) 2195–2197.

[9] J. D. Towers, Two methods for discretizing a delta function supported on a level set, J. Comput. Phys. 220 (2007) 915–931.

[10] J. D. Towers, A convergence rate theorem for finite difference approximations to delta functions, J. Comput. Phys. 227 (2008) 6591–6597.

[11] A.-K. Tornberg, B. Engquist, Regularization techniques for numerical approximation of pdes with singularities, J. Scient. Comp. 19 (2003) 527–552.

[12] A.-K. Tornberg, Multi–dimensional quadrature of singular and discontinuous functions, BIT 42 (2002) 644–669.

[13] E. Olsson, G. Kreiss, A conservative level set method for two phase flow, J. Comput. Phys. 210 (2005) 225–246.

[14] E. Olsson, G. Kreiss, S. Zahedi, A conservative level set method for two phase flow ii, J. Comput. Phys. 225 (2007) 785–807.

[15] R. P. Beyer, R. J. Leveque, Analysis of a one–dimensional model for the immersed boundary method, SIAM J. Numer. Anal. 29 (2) (1992) 332–364.

[16] B. E. Segee, Using spectral techniques for improved performance in artificial neural networks 500–505.

[17] G. Dahlquist, Åke Björck, Numerical Methods in Scientific Computing, Siam, 2008.

[18] P. J. Davis, P. Rabinowitz, Numerical Integration, Blaisdell publishing company, 1967.

[19] J. R. Munkres, Analysis on Manifolds, Westview Press, 1991.

[20] J. R. Munkres, Elementary differential topology, Princeton University press, 1963.