Summer 2017

Direct computations of marangoni driven flows using a volume of fluid method

Ivana Seric
New Jersey Institute of Technology

Recommended Citation

Seric, Ivana, "Direct computations of marangoni driven flows using a volume of fluid method" (2017). *Dissertations*. 40.
https://digitalcommons.njit.edu/dissertations/40

ABSTRACT

DIRECT COMPUTATIONS OF MARANGONI DRIVEN FLOWS USING A VOLUME OF FLUID METHOD

by
Ivana Seric

The volume of fluid (VoF) interface tracking methods have been used for simulating a wide range of interfacial flows. An improved accuracy of the surface tension force computation has enabled the VoF method to become widely used for simulating flows driven by the surface tension force. A general methodology for the inclusion of variable surface tension coefficient into a VoF based Navier-Stokes solver is developed. This new numerical model provides a robust and accurate method for computing the surface gradients directly by finding the tangent directions on the interface using height functions. The implementation applies to both temperature and concentration dependent surface tension coefficient, along with the setups involving a large jump in the temperature between the fluid and its surrounding, as well as the situations where the concentration should be strictly confined to the fluid domain, such as the mixing of fluids with different surface tension coefficients. The accuracy and convergence of the surface gradient computation are presented for various geometries, and for a classical problem of the thermocapillary migration of bubbles. The study of several applications of variable surface tension flows is presented, such as the breakup of liquid metal films and filaments, and the coalescence of drops characterized by different surface tension.

DIRECT COMPUTATIONS OF MARANGONI DRIVEN FLOWS USING A VOLUME OF FLUID METHOD

by
Ivana Seric

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
and Rutgers, The State University of New Jersey–Newark
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Mathematical Sciences

Department of Mathematical Sciences, NJIT
Department of Mathematics and Computer Science, Rutgers–Newark

August 2017

APPROVAL PAGE

DIRECT COMPUTATIONS OF MARANGONI DRIVEN FLOWS USING A VOLUME OF FLUID METHOD

Ivana Seric

Dr. Shahriar Afkhami, Dissertation Co-Advisor Date
Associate Professor of Mathematical Sciences, New Jersey Institute of Technology

Dr. Lou Kondic, Dissertation Co-Advisor Date
Professor of Mathematical Sciences, New Jersey Institute of Technology

Dr. Linda J. Cummings, Committee Member Date
Professor of Mathematical Sciences, New Jersey Institute of Technology

Dr. Philip D. Rack, Committee Member Date
Professor of Materials Science and Engineering, University of Tennessee

Dr. David G. Shirokoff, Committee Member Date
Assistant Professor of Mathematical Sciences, New Jersey Institute of Technology

BIOGRAPHICAL SKETCH

Author:	Ivana Seric
Degree:	Doctor of Philosophy
Date:	August 2017

Undergraduate and Graduate Education:

- Doctor of Philosophy in Mathematical Sciences,
  New Jersey Institute of Technology, Newark, NJ, 2017
- Bachelor of Science,
  New Jersey Institute of Technology, Newark, NJ, 2011

Major: Applied Mathematics

Publications:

I. Seric, S. Afkhami, and L. Kondic. Direct numerical simulations of variable surface tension flows using a Volume-of-Fluid method. *Journal of Computational Physics* (under review) (2017).

C. A. Hartnett, I. Seric, K. Mahady, L. Kondic, S. Afkhami, J. D. Fowlkes, and P. D. Rack. Exploiting the Marangoni effect to initiate instabilities and direct the assembly of liquid metal filaments. to appear in: Langmuir, (2017).

I. Seric, S. Afkhami, and L. Kondic, Interfacial instability of thin ferrofluid films under a magnetic field. *Journal of Fluid Mechanics Rapids* **755** (2014).

S. P. Naughton, N. K. Patel, and I. Seric, Instability of gravity driven flow of liquid crystal films. *SIAM Undergraduate Research Online (SIURO)* **5** (2012).

Presentations:

Direct numerical simulations of variable surface tension flows using a Volume-of-Fluid method, University of North Carolina at Chapel Hill, June 8, 2017.

Direct numerical simulations of variable surface tension flows using a Volume-of-Fluid method. *Numerical analysis seminar*, Los Alamos National Laboratory, May 1, 2017.

Direct numerical simulation of variable surface tension flows using Volume-of-Fluid method, *Applied Math Days*, Rensselaer Polytechnic Institute, Troy, NY, April 7-8, 2017.

Direct computations of Marangoni-induced flows using a Volume-of-Fluid method. *SIAM Conference on Computational Science and Engineering*, Atlanta, GA, February 27-March 3, 2017.

Explicit demonstration of the role of Marangoni effect in the breakup of nanoscale liquid filaments. *Complex Fluids and Soft Matter Workshop*, Hoboken, NJ, January 13, 2017.

Explicit demonstration of the role of Marangoni effect in the breakup of nanoscale liquid filaments. *American Physical Society, 69th Annual Meeting of the Division of Fluid Dynamics*, Portland, OR, November 20-22, 2016.

Spreading and mixing of drops on a miscible liquid of different surface tension. *American Physical Society, 69th Annual Meeting of the Division of Fluid Dynamics*, Portland, OR, November 20-22, 2016.

Interfacial instability of thin ferrofluid films under a magnetic field. *Complex Fluids and Soft Matter Workshop*, Newark, NJ, January 16, 2015.

Volume of fluid simulations of liquefied metal nanofilms with Marangoni effects. *American Physical Society, 67th Annual Meeting of the Division of Fluid Dynamics*, San Francisco, CA, November 23-25, 2014.

Numerical simulations of a ferrofluid drop on a substrate under an applied magnetic field. *American Physical Society, 66th Annual Meeting of the Division of Fluid Dynamics*, Pittsburgh, PA, November 24-26, 2013.

Posters:

Breakup of liquid metal filaments, *Dana Knox Student Research Showcase*, New Jersey Institute of Technology, Newark, NJ, April 19, 2017.

Breakup of liquid metal filaments, *Graduate Student Research Day*, New Jersey Institute of Technology, Newark, NJ, November 1, 2016.

Direct computations of Marangoni-induced flows using a volume of fluid method, *Frontiers in Applied and Computational Mathematics*, New Jersey Institute of Technology, Newark, NJ, June 3-4, 2016.

Interfacial instability in thin ferrofluid films under a magnetic field, *Frontiers in Applied and Computational Mathematics*, New Jersey Institute of Technology, Newark, NJ, May 22-23, 2014.

Interfacial instability in thin ferrofluid films under a magnetic field, *Dana Knox Student Research Showcase*, New Jersey Institute of Technology, Newark, NJ, April 16, 2014.

Numerical simulations of a ferrofluid drop on a substrate under an applied magnetic field, *Graduate Student Research Day*, New Jersey Institute of Technology, Newark, NJ, October 31, 2013.

Long-wave approximation of a ferrofluid film under an external magnetic field, *Frontiers in Applied and Computational Mathematics*, New Jersey Institute of Technology, Newark, NJ, May 31 - June 2, 2013.

Missing Page

To Split,
the most beautiful city in the world,
to my parents, Nina and Ivica,
and to Kyle.

ACKNOWLEDGMENT

First and foremost, a very special thank to my advisors Prof. Shahriar Afkhami and Prof. Lou Kondic for their guidance and for always being supportive. They have taught me to pay attention to the details, but at the same time to keep the big picture in mind. They were also the integral part of my decision to pursue a Ph.D. degree by introducing their research to me while I was an undergraduate student.

I would also like to thank my committee members Prof. Linda J. Cummings, Prof. David G. Shirokoff, and Prof. Philip D. Rack, for fruitful discussions and all their advice. In particular, I would like to thank Prof. Cummings for insightful questions during our meetings, which helped deepen my understanding of the physics and improved my presentation skills, and Prof. Rack, for providing interesting applications and the driving motivation for the topic of this dissertation.

Next, I would like to thank all my undergraduate professors, but specifically Prof. Eliza (Z.-H.) Michalopoulou and Prof. Yassine Boubendir, who encouraged me to continue my education after receiving the bachelors degree. I would also like to thank the Department of Mathematical Sciences whose support made this thesis work possible. I would also like to acknowledge partial support by the NSF Grant No. CBET-1604351.

I would like to thank my colleagues and friends at the graduate program Lenka Kovalčinová, Aminur Rahman, Peter Szu-Pei Fu, Casayndra Basarab, and Ensela Mema who made the process fun starting from the discussions about mathematics, countless cups of coffee, through our hiking adventures. I would like to thank my friend, colleague and flatmate Valeria Barra who turned our house into home with music, art and delicious food. I would also like to thank my friend Winnie Yeung, who is always there for me when I need a family away from home. Most of all, I

would like to thank Kyle Mahady for putting up with me through good and bad, and for giving a meaning to it all.

Finally, I would like to thank my family: my grandmother Hedviga Ozretić for all her stories over cappuccino; my parents Nina and Ivica Šerić for their unconditional love and support, and for always knowing a way to make me laugh; my brothers Antonio and Nino Šerić for always being there for me when I needed help; and last but not the least: my nephew Luka Šerić for inspiring me to try to make the world a better place for him.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

LIST OF FIGURES
(Continued)

LIST OF FIGURES
(Continued)

LIST OF FIGURES
(Continued)

CHAPTER 1

INTRODUCTION

Interfacial flows appear in many forms in nature and industry, such as bubbles/drops, jets, waves and films, and on a variety of length scales from nanoscopic to ocean scale. Some industrial applications involve chemical engineering, e.g., bubbly flows are in the core of the chemical reactors and in fuel atomization, or coastal engineering where understanding ocean waves is of great importance. Thin fluid films are important for understanding coating and drying processes in printing and polymer production [6, 7], and many other applications. Particularly in the context of thin films, it is becoming more and more important to accurately and reliably model the interfacial processes on very short scales, where the interfacial effects interact closely with the ones involving fluid/solid interactions (wetting). In this field, a significant amount of research is carried out using asymptotic long wave theory, however, there is an emerging set of problems involving large contact angles where such an approach is not accurate. It is therefore important to be able to compute accurately the interfacial effects by solving directly Navier-Stokes equations coupled with appropriate boundary conditions.

1.1 Volume of Fluid Method

Tracking the interface accurately is a challenge for simulating interfacial flows, due to the jump discontinuity in the physical properties of the two fluids separated by the interface. Some of the methods for tracking interfaces include the front tracking, boundary integral, level set, and volume of fluid (VoF) methods [8, 9, 10]. The VoF method tracks the interface implicitly by tracking fluid volume in each computational cell (control volume), using the volume fraction function, which is advected with the flow. The volume fraction function is a scalar function defined in each computational cell as the ratio of the fluid volume in the cell to the cell volume. The interface is then reconstructed from the volume fractions. The main challenges in the VoF method are advection of the volume fraction, interface reconstruction, and accurate calculation of interfacial geometrical features; most of these difficulties have been resolved in recent years and empowered the VoF method to become a widespread method for modeling flows with interfaces [11, 12].

First volume tracking algorithms are introduced by DeBar [13], Hirt and Nichols [14], and Noh and Woodward [15]. In [14, 15], a simple line interface calculation (SLIC) for interface reconstruction is used, where the interface is represented by a piecewise-constant line in each interfacial cell constructed either vertically or horizontally. In the algorithm by DeBar [13], a piecewise linear interface calculation (PLIC) is used; improved versions of which [16, 17, 18] are still widely used methods in volume tracking algorithms [19]. More accurate interface reconstruction methods can be achieved with least squares VoF interface reconstruction algorithm (LVIRA) [20]. The interface can also be reconstructed using higher order polynomials, e.g., parabolas [21]. More detailed review on interface reconstruction in the VoF method can be found in [17, 22].

The approximation of the surface tension term is one of the main challenges in the VoF methods, due to the difficulty of calculating the surface normals and curvature. In the interface tracking methods, the surface normals and curvature are computed from the gradients of a function representing the interface or interface positions: in the front tracking method the position of the marker points on the interface is known; in the level set method the interface is tracked via a continuous function, whose gradients can be computed directly using finite difference methods. On the other hand, in the VoF methods, the position of the interface is approximated from the volume fractions, i.e., a step functions whose gradients need to be computed with care.

A breakthrough for simulating surface tension driven flows in the VoF method was made by Brackbill et al. [23] with the continuum surface force method (CSF), where the surface tension force is applied as a volumetric force at the interface. However, the original approach suffered from spurious currents, i.e., flow induced by the discretization of the surface force and the pressure gradient, whose magnitudes do not converge with the mesh refinement. The obstacle of spurious currents creation has been resolved by applying the surface force consistently with the calculations of the pressure gradient in balanced force method [21, 24], and by more accurate calculations of interface normals and curvature using generalized height functions [11, 12, 24, 25, 26, 27]. A summary of these methods is given in Section 2.2.

1.2 Numerical Simulations of Variable Surface Tension Flows

Flows induced by spatial variations in the surface tension, also known as the Marangoni effect [28], can be caused by surfactants, temperature or concentration gradients, or a combination of these effects. Understanding these flows is important since they are relevant in microfluidics [29], heat pipe flows [30], motion of drops or bubbles in materials processing applications that include heating or cooling [31], evolution of metal films of nanoscale thickness melted by laser pulses [32, 33], and in a variety of other thin film flows, see [34, 35] for reviews.

Numerical methods for studying variable surface tension flows include front tracking [36], level set [37], diffuse interface [38], marker particle [39, 40], immersed boundary [41], boundary integral [42], interface-interaction [43], and VoF [44, 45, 2] methods. The VoF method is efficient and robust for tracking topologically complex evolving interfaces. The improvements in recent years in the computation of the surface tension have empowered the VoF method to become a widespread method for modeling interfacial flows [11, 12]. However, an accurate implementation of the variable surface tension in the VoF formulation is still lacking a general treatment.

A challenge for including variable surface tension effects into the VoF method is that the surface tension is not known exactly at the interface – only the value averaged over a computational cell containing the interface is known. To obtain the surface tension at the interface, an approximation from the values near the interface, usually calculated at the center of each adjacent computational cell, is necessary. As we outline below, the approximation of the interface values has been carried out in the literature differently, depending on the physics of the problem studied. An additional major issue concerns computing the surface gradients of the surface tension at the interface.

In [46] and [2], the VoF method is used to study flows involving temperature dependent surface tension. The implementation in [46] solves the heat equation in fluids on both sides of the interface. Then the continuity of the temperature and flux at the interface is imposed, along with the conservation of energy in the cell containing the interface. In the last step, the temperature in the fluid and air in the cell is approximated. These temperature values are then used to calculate surface gradients of the temperature from nearby cells that are not cut by the interface; these gradients are then exponentially extrapolated to the interface. In the work by Ma and Bothe [2], the temperature at the interface is approximated from the temperatures in the liquid and the gas by imposing the continuity of heat flux at the interface. The surface gradients of the temperature are approximated by computing the derivatives in each coordinate direction using finite differences, and then projecting them onto the tangential direction. If the interface is not contained in all cells of the finite difference stencil, then one sided differences are used. Hence, this method requires temperature solution on both sides of the interface and therefore cannot be used for setups involving a large difference in thermal conductivity of the two fluids, since the fluids may have a large temperature difference. Furthermore, both of these methods are not applicable to setups where the surface tension only depends on the concentration, such as mixing of miscible liquids with different surface tension. In the work by James and Lowengrub [45], the VoF method is used to study the flows induced by the surfactant concentration gradient. In their method, the concentration values at the interface are obtained by imposing the condition that the average concentration at the interface is equal to the average concentration in the cell containing the interface. Then, the surface gradients are computed using the cell-center interfacial concentration in the two adjacent cells.

Here, we develop a method that can be applied to both temperature and concentration dependent surface tension, with the surface gradients computed using the cell-center values in the interfacial cells only. We find the tangential gradients directly by computing the tangent directions on the interface using height functions [1]. This method can be applied to setups such that the concentration is confined to the fluid domain, e.g., mixing of liquids with different surface tension coefficients, as well as configurations involving a jump in the temperature between the liquid and the surrounding. Since our method does not depend on whether we consider temperature or concentration gradients, we will use them interchangeably in the remaining part of the paper.

Our numerical method is implemented using GERRIS: an open source adaptive Navier-Stokes solver [47, 1]. The current version includes CSF [23] implementation of the surface tension force with height function algorithm for computing interfacial normal and curvature [1]. Here, we present the method for extending this formulation to include variable surface tension, allowing us to consider the surface force in the direction tangential to the interface. As far as we are aware, this is the first implementation of the variable surface tension combined with the accurate implementation of the CSF method, such that the curvature and interface normals are computed using generalized height functions [1]. Our extension is a step closer to to covering all aspects of variable surface tension flows; the remaining one is the implementation of the surfactant transport and surface tension gradients due to the presence of surfactants.

1.3 Contact Angle Implementation

One of the applications of our method involves the breakup of thin films and the dynamics of drops on a substrate. In order to study both of these applications, a contact angle boundary condition needs to be imposed at the substrate. Implementation of an arbitrary contact angle in three spatial dimensions is also discussed. In particular, we describe an algorithm for imposing a contact angle boundary condition in VoF simulations using the height function method described in [48, 4]. We present an alternative implementation of this method using GERRIS flow solver in 2D, and discuss its extension to 3D.

1.4 Applications of Variable Surface Tension Flows

Directed and self-assembly of nanoparticles are low cost methods for producing arrays of nanoparticles with controlled particle size and spacing [49, 50]. Ensembles of nanoparticles have different properties compared to an individual particle or bulk materials, and can be used in plasmonic light trapping in thin film solar cells [50]. In spherical nanoparticles arranged into hexagonal arrays a collective plasmonic mode with diffractive coupling between the particles can be excited, which along with their fast and inexpensive fabrication procedure, makes them promising for sensing in biomedical applications [51]. Liquid metals directed assembly has been studied in experiments where the metal films are liquefied by a laser pulse. The assembly can be directed by the initial pattern, which dewets and solidifies into arrays of particles on the substrate [52, 53, 54, 55, 56], by varying the thickness of the initial nanostructures [57, 58, 58, 59, 60, 61, 62, 63], or by varying the laser source [64, 65].

Numerically, the breakup of liquid metals has been studied in the long wave regime, where spinodal dewetting drives the breakup [66, 63]. The thermocapillary effect is also considered, where the surface tension depends linearly on the temperature of the film [67, 68], in the parameter regime where the temperature satisfies the steady state heat equation with variation in the out-of-plane direction only. The heat source from the laser is modeled by the film absorption of the laser energy according to [69], with time independent and uniform and spatially dependent laser beam intensity distribution. The breakup of liquid metal films is also studied using VoF simulations where the breakup is induced by the Rayleigh-Plateau instability [70] with a sinusoidal initial film pattern [55], square wave pattern [56, 71], and by dewetting of liquid metal nanostructures [72] where the capillary force drives the flow.

We study the influence of the variable surface tension coefficient on the directed and self-assembly of nanoparticles from liquefied metal films and filaments. First, we investigate a new methodology for directed assembly of metal nanoparticles from bimetal filaments where the breakup dynamics and the resulting particle configurations are determined by the Marangoni force caused by the variation in the metal concentration at the liquid–air interface. Second, we investigate the influence of the variable surface tension flow created by the temperature variations in the film, i.e., the thermocapillary force, on the stability of the interface.

1.5 Summary and Overview

The rest of the dissertation is organized as follows: In Chapter 2.2 we introduce the governing equation of the problem, and give an overview of the VoF method, including the CSF method for the computation of the surface tension, balanced force algorithm, and height function method for computation of curvature and interface normal. Chapter 3 describes in detail the implementation of the variable surface tension coefficient first for a special case where the surface tension can be modeled as a function of the film height in two dimensions in Section 3.1, and second for a general variable surface tension coefficient in two and three dimensions in Section 3.2. Section 3.3 illustrates the performance of our method for various test cases, where we compare the results with exact and asymptotic solutions. Chapter 4 gives an overview of the contact angle algorithm in two dimensions and a guideline for implementation in thee dimensions. Chapter 5 gives various physical applications of the developed numerical method. First, in Section 5.1 we present simulation of the coalescence of droplets with different surface tensions, and comparison of the results with available experiments in the literature. Second, we show the influence of the concentration driven Marangoni force on the breakup of liquid metal filaments in Section 5.2, and on the breakup of films with cylindrical perturbations in Section 5.3. Next we show the influence of the thermocapillary force on the breakup of two dimensional liquid metal films, and three dimensional filaments, in Section 5.4. The dissertation ends with discussion of results and future direction in Chapter 6. Additional details of mathematical models and computational algorithms are given in the Appendices.

CHAPTER 2

MATHEMATICAL MODELS AND OVERVIEW OF THE NUMERICAL METHODS

In this chapter, we present the governing equations describing the two phase fluid flow and the equations governing the concentration or temperature in Section 2.1. We represent the two phase flow by the Navier-Stokes equations, where the material properties are phase dependent. The surface forces at the interface between two fluids are z represented by a body force using the CSF method. The temperature or the concentrations are governed by advection-diffusion equations. In Section 2.2 we give a brief overview of the numerical methods used in Gerris.

2.1 Governing Equations

We consider an incompressible two-phase flow described by Navier-Stokes equations

$$\rho(\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}) = -\nabla p + \nabla \cdot (2\mu D) + \mathbf{F}, \tag{2.1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{2.2}$$

and the advection of the phase-dependent density $\rho\,(\chi)$

$$\partial_t \rho + (\mathbf{u} \cdot \nabla)\rho = 0, \tag{2.3}$$

where $\mathbf{u} = (u, v, w)$ is the fluid velocity, p is the pressure, $\rho(\chi) = \chi\rho_1 + (1-\chi)\rho_2$ and $\mu(\chi) = \chi\mu_1 + (1-\chi)\mu_2$ are the phase dependent density and viscosity respectively, D is the rate of deformation tensor $D = \left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)/2$. Subscripts 1 and 2 correspond to the fluids 1 and 2, respectively (see Figure 2.1). Here, χ is the (continuous) characteristic function, such that $\chi = 1$ in the fluid 1, and $\chi = 0$ in the fluid 2. Note that any body force can be included in $\mathbf{F}$. The characteristic function is advected

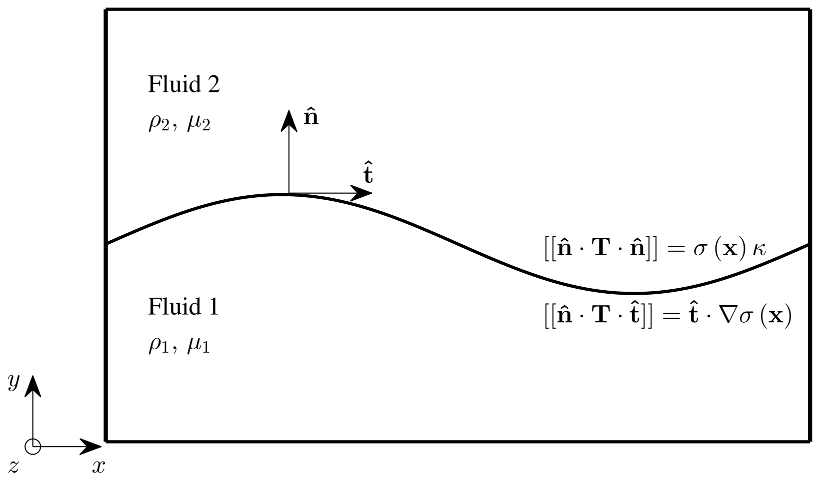


Figure 2.1 Schematic of a system with two immiscible fluids and the corresponding boundary conditions.

with the flow, thus

$$\partial_t\chi + (\mathbf{u}\cdot\nabla)\chi = 0. \tag{2.4}$$

Note that solving Equation (2.4) is equivalent to solving Equation (2.3).

The presence of an interface gives rise to the stress boundary conditions, see Figure 2.1. The normal stress boundary condition at the interface defines the stress jump [73, 74]

$$[\![\hat{\mathbf{n}}\cdot\mathbf{T}\cdot\hat{\mathbf{n}}]\!] = \sigma(\mathbf{x})\,\kappa, \tag{2.5}$$

where $\mathbf{T} = -p\mathbf{I} + \mu\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)$ is the total stress tensor, $\sigma(\mathbf{x})$ is the surface tension coefficient, κ is the curvature of the interface, and $\hat{\mathbf{n}}$ is the unit normal at the interface pointing out of the fluid 1. The variation of surface tension coefficient results in the tangential stress jump at the interface

$$[\![\hat{\mathbf{n}}\cdot\mathbf{T}\cdot\hat{\mathbf{t}}]\!] = \hat{\mathbf{t}}\cdot\nabla\sigma(\mathbf{x}), \tag{2.6}$$

which drives the flow from regions of low surface tension to those with high surface tension. Here, $\hat{\mathbf{t}}$ is the unit tangent vector in two dimensions (2D); in three dimensions

(3D) there are two linearly independent unit tangent vectors. Using the CSF method [23], the forces resulting from the normal and tangential stress jump at the interface can be included in the body force $\mathbf{F} = \mathbf{F}_{sn} + \mathbf{F}_{st}$, defined as

$$\mathbf{F}_{sn} = \sigma\left(\mathbf{x}\right)\kappa\delta_s\hat{\mathbf{n}}, \tag{2.7}$$

and

$$\mathbf{F}_{st} = \nabla_s\sigma\left(\mathbf{x}\right)\delta_s, \tag{2.8}$$

where δ_s is the Dirac delta function centered at the interface, $\delta_s\hat{\mathbf{n}} = \nabla\chi$, and ∇_s is the surface gradient. Substituting Equations (2.7) and (2.8) in the momentum Equation (2.1) gives

$$\rho(\partial_t\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u}) = -\nabla p + \nabla\cdot(2\mu D) + \sigma\left(\mathbf{x}\right)\kappa\delta_s\hat{\mathbf{n}} + \nabla_s\sigma\left(\mathbf{x}\right)\delta_s. \tag{2.9}$$

We define the nondimensional variables, denoted with a superscript "*", as

$$\mathbf{x}^* = \frac{\mathbf{x}}{a}, \quad t^* = \frac{t}{t_r}, \quad \mathbf{u}^* = \frac{\mathbf{u}}{U_0}, \quad p^* = \frac{p}{p_0},$$
$$\rho^*\left(\chi\right) = \frac{\rho\left(\chi\right)}{\rho_1}, \quad \mu^*\left(\chi\right) = \frac{\mu\left(\chi\right)}{\mu_1}, \quad \sigma^* = \frac{\sigma}{\sigma_0},$$

where the scales a, t_r, U_0, p_0, and σ_0 are chosen based on the problem studied. Hence the dimensionless Equation (2.9) is

$$\begin{aligned}\mathrm{Re}\rho^*(\partial_t^*\mathbf{u}^* + \mathbf{u}^*\cdot\nabla^*\mathbf{u}^*) = -\nabla^*p^* + \nabla^*\cdot(2\mu^*D^*) + \\ + \mathrm{Ca}^{-1}\sigma^*\kappa^*\delta_s^*\hat{\mathbf{n}} + \frac{\sigma_0}{U_0\mu_1}\nabla_s^*\sigma^*\delta_s^*\hat{\mathbf{t}},\end{aligned} \tag{2.10}$$

where Re and Ca are the Reynolds and Capillary numbers respectively, defined as

$$\mathrm{Re} = \frac{\rho_1 U_0 a}{\mu_1}, \quad \mathrm{Ca} = \frac{U_0\mu_1}{\sigma_0}. \tag{2.11}$$

The surface tension is a function of temperature, T, or concentration, C, which satisfy an advection diffusion equation

$$\rho\left(\chi\right) C_p\left(\chi\right)\left(\partial_t T + (\mathbf{u}\cdot\nabla)T\right) = \nabla\cdot\left(k\left(\chi\right)\nabla T\right), \tag{2.12}$$

$$\partial_t C + (\mathbf{u}\cdot\nabla)C = \nabla\cdot\left(\alpha\left(\chi\right)\nabla C\right), \tag{2.13}$$

where $C_p\left(\chi\right)$, $k\left(\chi\right)$ and $\alpha\left(\chi\right)$ are the phase dependent heat capacity, conductivity and diffusivity, respectively. Along with the scales given above, Equations (2.12) and (2.13) are nondimensionalized using the following scales

$$k^*\left(\chi\right) = \frac{k\left(\chi\right)}{k_1},\quad C_p^*\left(\chi\right) = \frac{C_p\left(\chi\right)}{C_{p_1}},\quad T^* = \frac{T}{T_0},\quad \alpha^*\left(\chi\right) = \frac{\alpha\left(\chi\right)}{\alpha_1}, \tag{2.14}$$

where T_0 is chosen based on the physics of the system. Hence the dimensionless equations (2.12) and (2.13) are

$$\mathrm{Ma}\rho^* C_p^*\left(\partial_t^* T^* + (\mathbf{u}^*\cdot\nabla^*)T^*\right) = \nabla^*\cdot\left(k^*\nabla^* T^*\right), \tag{2.15}$$

$$\mathrm{Ma}\left(\partial_t^* C^* + (\mathbf{u}^*\cdot\nabla^*)C^*\right) = \nabla^*\cdot\left(\alpha^*\nabla^* C^*\right), \tag{2.16}$$

where Ma is the Marangoni number defined as

$$\mathrm{Ma} = \frac{U_0 a}{\alpha_1}. \tag{2.17}$$

The diffusivity, α_1, in the heat equation is $\alpha_1 = k_1/(\rho_1 C_{p_1})$. Surface tension can have linear or nonlinear dependence on temperature or concentration. In many applications the surface tension depends on the temperature linearly, i.e.,

$$\sigma = \sigma_0 + \sigma_T\left(T - T_R\right), \tag{2.18}$$

where σ_0 is the surface tension at a reference temperature T_R, and σ_T is a constant. Then, we can write

$$\nabla_s\sigma = \sigma_T\nabla_s T, \tag{2.19}$$

and compute $\nabla_s T$ in the same manner as $\nabla_s \sigma$. Using the scales given above, the dimensionless Equation (2.18) is

$$\sigma^* = 1 + \frac{\sigma_T T_0}{\sigma_0} \left(T^* - T_R^* \right). \tag{2.20}$$

In the following section, we describe a method for computing $\nabla_s \sigma$ in general, regardless of the dependence on the temperature or concentration.

2.2 Overview of the Numerical Methods

The system given in Equation (2.1) is discretized using a time-splitting projection method [75]

$$\rho_{n+1/2} \left[\frac{\mathbf{u}_* - \mathbf{u}_n}{\Delta t} + \mathbf{u}_{n+1/2} \cdot \nabla \mathbf{u}_{n+1/2} \right] = \nabla \cdot \left[\mu_{n+1/2} (D_n + D_*) \right] + \mathbf{F}_{n+1/2}, \tag{2.21}$$

$$\frac{\chi_{n+1/2} - \chi_{n-1/2}}{\Delta t} + \nabla \cdot (\chi_n \mathbf{u}_n) = 0, \tag{2.22}$$

$$\mathbf{u}_{n+1} = \mathbf{u}_* - \frac{\Delta t}{\rho_{n+1/2}} \nabla p_{n+1/2}, \tag{2.23}$$

$$\nabla \cdot \mathbf{u}_{n+1} = 0. \tag{2.24}$$

First the intermediate (auxiliary) velocity, $\mathbf{u}_*$, is calculated from Equation (2.21). The approximation of the velocity advection term $\mathbf{u}_{n+1/2} \cdot \nabla \mathbf{u}_{n+1/2}$ is carried out using the Bell-Colella-Glaz second-order unsplit upwind scheme [1, 47, 76]. Then the pressure, $p_{n+1/2}$, is found by solving the Poisson equation

$$\nabla \cdot \left(\frac{\Delta t}{\rho_{n+1/2}} \nabla p_{n+1/2} \right) = \nabla \cdot \mathbf{u}_*, \tag{2.25}$$

which is solved using the multilevel solver described in [47]. Surface tension force is applied to the auxiliary face- and cell-centered velocity field, denoted by superscripts

f and c respectively, via the balanced force algorithm [21, 24], as follows

$$\mathbf{u}_*^f = \mathbf{u}_*^f + \frac{\Delta t\, \sigma\, \kappa_{n+1/2}^f}{\rho_{n+1/2}} \nabla \chi_{n+1/2}^f, \tag{2.26}$$

$$\mathbf{u}_*^c = \mathbf{u}_*^c + \left| \frac{\Delta t \sigma\, \kappa_{n+1/2}^f}{\rho_{n+1/2}^f} \nabla^f \chi_{n+1/2} \right|^c . \tag{2.27}$$

This approach, along with an accurate curvature computation, decreases the spurious currents at the interface for a case of a stationary drop in equilibrium [21, 24]. Equation (2.21) gives a Crank-Nicholson discretization of viscous terms and is second order accurate and unconditionally stable [1]. The $\mathbf{u}_{n+1/2} \cdot \nabla \mathbf{u}_{n+1/2}$ term is approximated using the Bell-Colella-Glaz second order unsplit upwind scheme [76]. This scheme is stable for Courant-Friedrichs-Lewy (CFL) number ($|u|\Delta t/\Delta x + |v|\Delta t/\Delta y + |w|\Delta t/\Delta z$) smaller than one [1]. Note that in GERRIS, the grid is discretized with $\Delta x = \Delta y = \Delta z \equiv \Delta$. The surface tension gives rise to another time step constraint: $\Delta t \leq \sqrt{\rho_1 \Delta^3/\pi\sigma}$ [1]. After solving Equation (2.25), a correction is applied to the intermediate velocity at the cell face

$$\mathbf{u}_{n+1}^f = \mathbf{u}_*^f - \frac{\Delta t}{\rho_{n+1/2}^f} \nabla^f p_{n+1/2}, \tag{2.28}$$

which results in the exactly divergence free face-centered velocity. The cell-centered velocity is approximately divergence-free and given by

$$\mathbf{u}_{n+1}^c = \mathbf{u}_*^c - \left| \frac{\Delta t}{\rho_{n+1/2}^f} \nabla^f p_{n+1/2} \right|^c , \tag{2.29}$$

where $|.|^c$ denotes the cell-centered value obtained from averaging over all cell faces.

Generalized height functions are used for computing normals and curvature of the interface [1]. The accuracy of the normal and curvature computation using height functions is comparable to the normal and curvature computation using the derivatives of the level set functions [25]. The height function value in each interfacial cell is a sum of the volume fractions in the fluid column in the given stencil. Figure

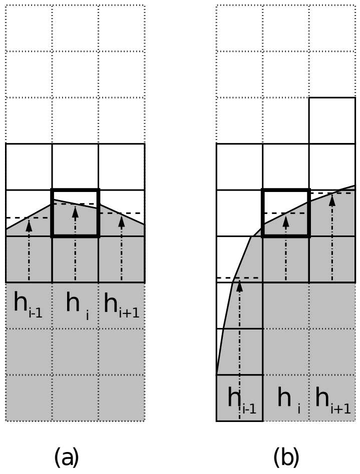


Figure 2.2 Height function computation with compact stencil constructed independently for each column. (a) Stencil for a slightly curved interface. (b) Asymmetric stencil, where each column requires different number of cells. *Source: [1].*

2.2 shows a compact stencil used for calculating height functions implemented in Gerris solver [1], where the solid lines show the stencil size required to calculate the curvature in the cell marked by the bold lines. The height function is then

$$\mathcal{H}_i = \sum_k \chi_{i,k}, \tag{2.30}$$

where the range of index k depends on the stencil size. The stencil size is determined by visiting the cells in the column, starting from the interfacial cell until a full cell is found in direction in the fluid – where $\chi = 1$, and an empty cell is found in the direction outside of the fluid – where $\chi = 0$. In Figure 2.2(a) only three cells are required for calculating the height function in each column. Figure 2.2(b) shows an asymmetric stencil where each column requires visiting a different number of cells.

In 2D, the height function can be calculated in either x or y direction, depending on the orientation of the interface. Here, we will present the discretization for height function for columns in the y direction (as in Figure 2.2). For heights collected in the x direction, the equations are equivalent with the derivatives of the height function with respect to x replaced by the derivatives with respect to y. The curvature of the interface is calculated from the height functions as

$$\kappa = \frac{\mathcal{H}_{xx}}{(1+\mathcal{H}_x^2)^{3/2}}, \tag{2.31}$$

and the normal vector at the interface is

$$\hat{\mathbf{n}} = \frac{1}{\sqrt{1+\mathcal{H}_x^2}}\left(\mathcal{H}_x, -1\right), \tag{2.32}$$

where the derivatives of the height functions in Equations (2.31) and (2.32) are calculated using a second order central difference

$$\mathcal{H}_x = \frac{\mathcal{H}_{i+1} - \mathcal{H}_{i-1}}{2\Delta} \tag{2.33}$$

$$\mathcal{H}_{xx} = \frac{\mathcal{H}_{i+1} - 2\mathcal{H}_i + \mathcal{H}_{i-1}}{\Delta^2}. \tag{2.34}$$

where Δ is the cell size. In 3D, e.g., if the columns are computed in the z direction, the curvature is

$$\kappa = \frac{\mathcal{H}_{xx} + \mathcal{H}_{yy} + \mathcal{H}_{xx}\mathcal{H}_y^2 + \mathcal{H}_{yy}\mathcal{H}_x^2 - 2\mathcal{H}_{xy}\mathcal{H}_x\mathcal{H}_y}{\left(1+\mathcal{H}_x^2+\mathcal{H}_y^2\right)^{3/2}}. \tag{2.35}$$

and the normal vector at the interface is

$$\hat{\mathbf{n}} = \frac{1}{\sqrt{1+\mathcal{H}_x^2+\mathcal{H}_y^2}}\left(\mathcal{H}_x, \mathcal{H}_y, -1\right). \tag{2.36}$$

For the case where consistent height functions cannot be constructed to compute the derivatives given in Equations (2.33), Gerris uses a so called generalized height function method [1] to compute the curvature and normals. In this case, interface height positions are collected and used for fitting a parabola in 2D or paraboloid in 3D, whose derivatives are subsequently used to compute the curvature and normal vectors.

CHAPTER 3

VARIABLE SURFACE TENSION IMPLEMENTATION

This chapter gives the details of the numerical implementation of the surface gradients of the variable surface tension coefficient, $\nabla_s \sigma$. The surface gradients are computed in two ways. First in Section 3.1 we present a method for a fluid film formulation in 2D, where the surface tension depends on the temperature linearly, and the temperature is known at the interface as a function of the film height. Second, in Section 3.2 the surface gradient is computed in a general manner for any scalar field defined on the interface, regardless of the interfacial geometry.

3.1 Marangoni Stress with a Given Temperature Field in 2D

As a first simplified case, we implement a method for computing the surface gradients of the surface tension coefficient, $\nabla_s \sigma$, where σ depends on the temperature linearly, such as specified by Equation (2.18), and the temperature is known as a function of the film height, h. These assumptions follow from an application to liquid metal films melted by a pulsed laser, explained in more detail in Section 5.4.1. Note that the approach presented here can be used only when the interface is a single valued function of x or y. Recall from Section 2.1 that the surface gradients of σ can be computed as in Equation (2.19). In the 2D case, we can write the surface gradient as $\nabla_s \ = \partial \ /\partial s \, \hat{\mathbf{t}}$. Following the above assumptions and simplifications, the surface gradients of σ can be computed as

$$\nabla_s \sigma = \frac{\sigma_T \frac{\partial T}{\partial h} \frac{\partial h}{\partial x}}{\mathrm{d}s} \, \hat{\mathbf{t}}, \tag{3.1}$$

where $\mathrm{d}s = \sqrt{1 + (\partial \mathcal{H}/\partial x)^2}$ is the arc length, and $\mathcal{H}$ is the height function described in Section 2.2. The gradient of the temperature with respect to the film height,

$\partial T/\partial h$, is computed analytically, and evaluated at each interfacial cell. Note that $\partial h/\partial x$ in Equation (3.1), is equivalent to $\partial \mathcal{H}/\partial x$, i.e., the derivative of the height function with columns in the y direction. Furthermore, $\partial \mathcal{H}/\partial x$ can be found from the height function with columns in the x direction, denoted $\mathcal{W}$ here, as $(\partial \mathcal{W}/\partial y)^{-1}$. Finally, the surface force at the interface is added as a body force using the CSF method (see Equation (2.9)).

We test the implementation by comparing a growth rate of the perturbations for a fluid film in the simulations with the prediction from the linear stability analysis (LSA) derived from the thin film approximation [77]. Consider a thin fluid film lying on a planar horizontal substrate. The dynamics can be described by applying a long wave approximation, which reduces the Navier-Stokes equations to a single nonlinear partial differential equation for the film height $h\,(x,t)$ [77]

$$3\eta \frac{\partial h}{\partial t} = -\nabla \cdot \left(\sigma_0 h^3 \nabla \nabla^2 h + \frac{3}{2} h^2 \nabla \sigma \right), \tag{3.2}$$

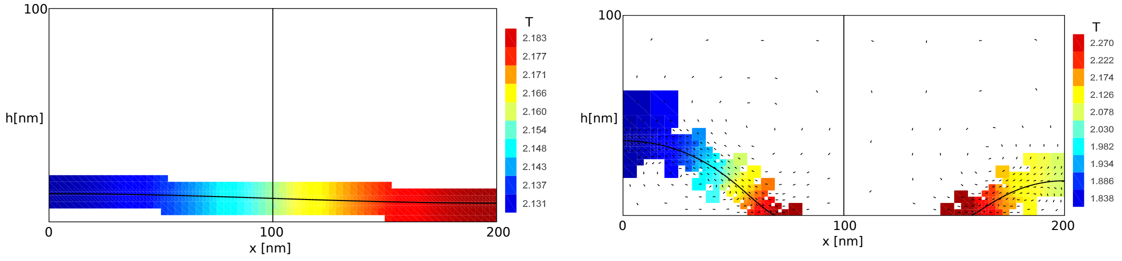
where the gradient in the 2D case is $\nabla \equiv \partial\ /\partial x$. Next we use the LSA to obtain the dispersion relation. Assume that the film is perturbed about the equilibrium film thickness h_0, as $h(x,t) = h_0 + \varepsilon h_0 e^{\beta t + kx}$ with a small parameter ε, growth rate β and wave number k. Using this expression for h in Equation (3.2), and keeping only leading order terms in ε, gives the dispersion relation

$$\beta = -\frac{h_0^2 k^2}{3\eta} \left(\sigma_0 h_0 k^2 - \frac{3}{2} \sigma_T \frac{\partial T}{\partial h} \right). \tag{3.3}$$

Here, we assume the surface tension depends on the temperature T linearly via (2.18). From Equation (3.3) we can see that the unstable modes ($\beta < 0$) are possible only if $\partial T/\partial h < 0$. We assume linear temperature dependence on the film thickness h

$$T(h) = T_R + \frac{\partial T}{\partial h}\, h,$$

where T_R is a reference temperature, and $\partial T/\partial h$ is constant.



(a) (b)

Figure 3.1 Temperature of the interface at the initial (a) and final time step (b). The film is initially perturbed around the equilibrium film thickness $h_0 = 11$ nm. The temperature is rescaled by the melting temperature of Nickel (T_R). Arrows show the direction of the velocity field.

For the purpose of the illustration of the test simulation setup, Figure 3.1 shows the interface configuration and the temperature of the interface for initial and final times of the simulation. In this example, the interface is perturbed with the most unstable wave number, $k_{max} \approx 0.0314\,\mathrm{nm}^{-1}$, i.e., the wave number corresponding to the maximum growth rate, β_{max}, from Equation (3.3). The material parameters used correspond to liquid Nickel: density $\rho = 7905.0\,\mathrm{kg\ m}^{-3}$, viscosity $\eta = 5.5 \times 10^{-3}\,\mathrm{Pa\ s}$, and in the expression (2.19) for the surface tension, $\sigma_0 = 1.778\mathrm{N\ m}^{-1}$, $\sigma_T = -0.38 \times 10^{-3}\mathrm{N\ m}^{-1}\mathrm{K}^{-1}$ and $T_R = 1727\mathrm{K}$. Symmetry boundary conditions are applied to the left left, right and top boundary, and no-slip and no-penetration conditions on the bottom boundary. A contact angle of 45° is imposed at the bottom boundary.

Figure 3.2 shows the comparison of the growth rate predicted by Equation (3.3), and measured from the numerical simulations. The measured points are not exactly on the analytically-predicted curve, but they follow the same trend. Note that the exact agreement is not expected here since long wave theory assumes small contact angles.

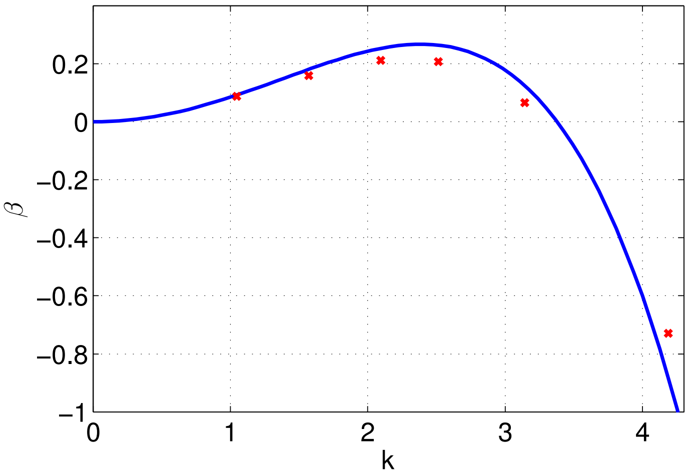


Figure 3.2 Comparison of the growth rate predicted by the linear stability analysis with the growth rate measured from the numerical simulations. The solid line shows the growth rate predicted by Equation (3.3), and red symbols show the growth rate measured from the simulations.

3.2 General Method

The proposed numerical method is implemented into GERRIS, which numerically solves Equations (2.1) to (2.3) using the VoF interface tracking method with an implicit treatment of the viscous forces [1, 47, 78]. The GERRIS code uses octree (3D) and quadtree (2D) grids, allowing to adaptively refine the grid in the immediate neighborhood of the interface. While we describe our implementation of the variable surface tension coefficient for uniform meshes, the extension to adaptively refined meshes is straightforward, following the implementation details described by [1, 47].

The surface gradient of any scalar field Q is defined as the projection of the gradient onto the surface, i.e.,

$$\nabla_s Q = \nabla Q - \hat{\mathbf{n}}\left(\hat{\mathbf{n}} \cdot \nabla Q\right) \tag{3.4}$$

where $\hat{\mathbf{n}}$ is the unit normal vector at the surface. However, this definition of the surface gradient can result in inaccuracies when implemented in the VoF method for general variable surface tension coefficient for two reasons. First, the discontinuities of the material properties across the interface can result in Q having a large jump across the interface: for example, in the case of surface tension coefficient dependence on the temperature where the fluids on each side of the interface have a large difference in the conductivity. The second reason is that, in general, the surface tension coefficient can depend on the concentration: for example, in the case of the mixing of two liquids with different surface tension coefficients, or in the case of surface tension coefficient dependent on the surfactant concentration.

Here, we propose a numerical method for implementing a general variable surface tension coefficient. We compute the surface gradient as

$$\mathbf{F}_{st} = \frac{\partial \sigma}{\partial s_1} \delta_s \hat{\mathbf{t}}_1 + \frac{\partial \sigma}{\partial s_2} \delta_s \hat{\mathbf{t}}_2, \tag{3.5}$$

where $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$ are the unit tangent vectors at the interface, pointing in the s_1 and s_2 directions, respectively. In our approach, we first define the surface tension coefficient values at the interface, then compute the derivatives of σ along the interface, and finally project the derivatives onto the tangent space defined by $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$. In the following sections we present the details of the implementation. In Section 3.2.1, we show how to approximate the value of the surface tension coefficient on the interface using the cell-center values. Then in Section 3.2.2, we show how $\partial \sigma / \partial s_d$, for $d = 1, 2$, are evaluated, along with the choice of the tangent vectors and addition of the tangential surface force using the CSF method.

3.2.1 Approximation of Interfacial Values of the Surface Tension Coefficient

The algorithm for implementing $\nabla_s \sigma (\mathbf{x})$ in the VoF method starts with the approximation of the interfacial values of the surface tension in each cell containing an interface segment. More precisely, we use the idea of constructing the columns of cells inspired by the computation of interfacial curvature and normals using height functions [1] (see Section 2.2). We should note however that to obtain a consistent height-function approximation, the interface must be sufficiently resolved. While we

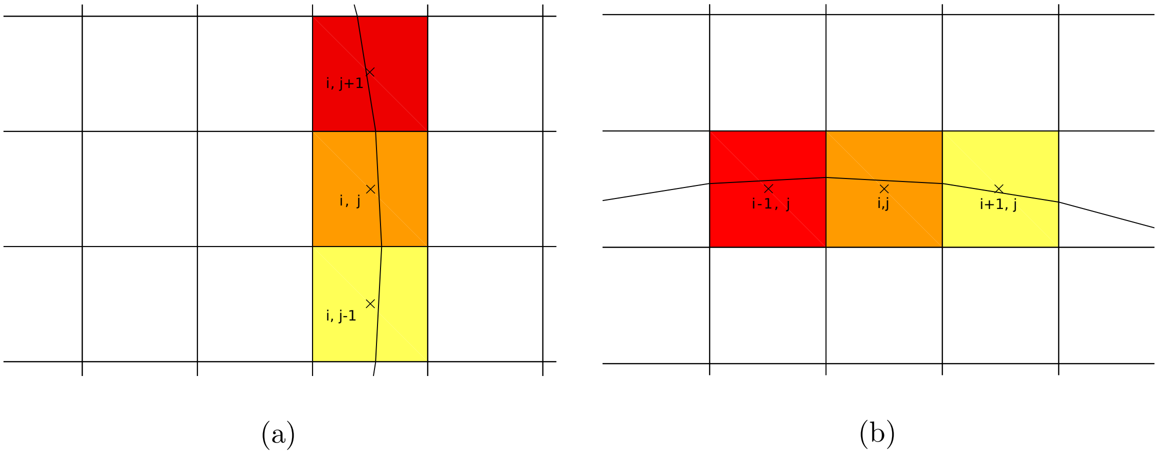
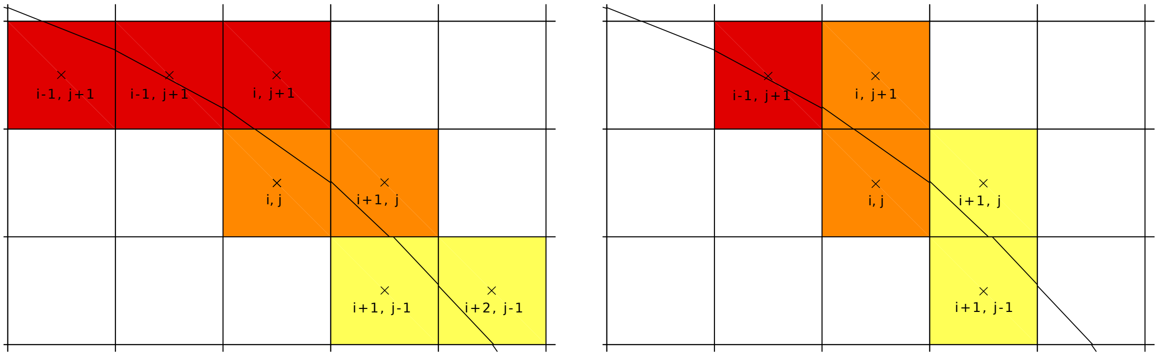


Figure 3.3 An example of the interface orientation, where columns in the x (a) and the y (b) direction for computing $\tilde{\sigma}^c(\mathbf{x})$ contain one interfacial cell. Each color shows a different column in interfacial cells $\mathcal{C}$.

cannot expect reasonable accuracy for the computation of the surface tension gradient in the tangential direction at very low resolutions, when the interface becomes very poorly resolved, one should resort to interpolation methods described by Popinet [1]. We show via examples in Section 3.3.3, the required spatial resolution below which a consistent height-function approximation is no longer possible.

Let $\sigma(\mathcal{C})$ be the surface tension coefficient evaluated from the temperature or concentration at the center of all interfacial cells $\mathcal{C}$, with the volume fraction $\chi(\mathcal{C})$. The surface tension coefficient in each column, denoted by $\tilde{\sigma}^c(\mathbf{x})$, is defined so that it has only one value in each column, regardless of how many interfacial cells are contained in that column. For illustration, Figure 3.3 shows columns that contain only one interfacial cell, and Figure 3.4 shows columns that contain more than one interfacial cell, where the same color denotes cells in the same column. The superscript, $c = x, y, z$, represents the column direction. For simplicity, here we show examples of the implementation in 2D, however, the algorithm extends trivially to 3D.

For columns with only one interfacial cell (see Figure 3.3(a) and (b) for the columns in the x and y directions, respectively), the surface tension of the interfacial cells, $\tilde{\sigma}^c$, is equal to the surface tension $\sigma(\mathcal{C})$ in the same cells. If there is more than



(a) (b)

Figure 3.4 An example of the interface orientation where columns in the x (a) and y (b) direction for computing $\tilde{\sigma}^c$ contain more than one interfacial cell. The cells with the same color belong to the same column.

one interfacial cell in the column, then $\tilde{\sigma}^c$ is approximated by the volume weighted average of the $\sigma(\mathcal{C})$ values. In Figure 3.4, the cells labeled with the cell indices will be used for computing $\tilde{\sigma}^c$ for columns in the x and y directions – Figures 3.4(a) and (b) respectively. For example, in Figure 3.4(a), the $\tilde{\sigma}^x$ is computed using the columns in the x direction, and the value of $\tilde{\sigma}^x$ in the column containing cell $\mathcal{C}_{i,j}$, denoted $\tilde{\sigma}^x_j$, is

$$\tilde{\sigma}^x_j = \frac{\chi_{i,j}\sigma_{i,j} + \chi_{i+1,j}\sigma_{i+1,j}}{\sum \chi_i}. \tag{3.6}$$

Note that the cells in the same column, in this particular example cells $\mathcal{C}_{i,j}$ and $\mathcal{C}_{i+1,j}$, have the same value of $\tilde{\sigma}^x$. For the columns in the y direction, as in Figure 3.4(b), $\tilde{\sigma}^y$ in the column containing cell $\mathcal{C}_{i,j}$, denoted $\tilde{\sigma}^y_i$, is computed as

$$\tilde{\sigma}^y_i = \frac{\chi_{i,j}\sigma_{i,j} + \chi_{i,j+1}\sigma_{i,j+1}}{\sum \chi_i}. \tag{3.7}$$

Again, the cells in the same column, in this case $\mathcal{C}_{i,j}$ and $\mathcal{C}_{i,j+1}$ have the same value of $\tilde{\sigma}^y$.

In our implementation, we first define $\tilde{\sigma}^c$ for all c in all interfacial cells. For certain interface orientations, it is possible to define $\tilde{\sigma}^c$ for columns in more than one direction, e.g., the interface in Figure 3.4. However, this is not always the case, e.g., in Figure 3.3(a) we can only compute $\tilde{\sigma}^x$, and in Figure 3.3(b) we can only compute $\tilde{\sigma}^y$. For the former case, in the following sections we describe how the direction of the columns is chosen along with the discussion of the computation of the surface forces.

3.2.2 The Computation of the Surface Forces

The next step in the variable surface force implementation is the evaluation of the derivatives along the interface, $\partial\sigma/\partial s_d$, where $d = 1, 2$, in equation (3.5). In 2D, we only need to compute the derivative in one direction, since the basis for a tangent line consists of only one vector. However, in 3D, we need two tangent vectors to form a basis for the tangent space, hence we need to evaluate the derivative in two directions. We now discuss the implementation of the method for 2D and 3D.

The Surface Force in 2D In 2D, Equation (3.5) simplifies to

$$\mathbf{F}_{st} = \frac{\partial\sigma}{\partial s^c}\delta_s\hat{\mathbf{t}}, \tag{3.8}$$

since we only have one tangential direction. In this case, $c = x, y$, denotes the direction of the columns (see Section 3.2.1). The derivative of the surface tension coefficient along the interface, $\partial\sigma/\partial s^c$, is approximated by the derivative of the interfacial value, $\tilde{\sigma}^c$ in the column which is formed in the direction c. The choice of the direction, c, is based on the interface orientation: c is chosen to be the same as the largest component of the normal vector to the interface. The same choice is made for computing curvature and the interface normal using the height functions [1].

In each interfacial cell, we compute the derivative along the interface using the center difference, i.e., the finite difference of the $\tilde{\sigma}^c$ in the two neighboring columns. For example, in Figure 3.3(a) and 3.4(a), the derivative is computed with respect to the y direction, as

$$\left(\frac{\partial\sigma}{\partial s^x}\right)_{i,j} = \frac{\tilde{\sigma}^x_{j+1} - \tilde{\sigma}^x_{j-1}}{\mathrm{d}s}. \tag{3.9}$$

As a reminder, $\tilde{\sigma}^x_j$ is the interfacial value of the surface tension in the column j constructed in the x direction. The arc length, $\mathrm{d}s$, is computed from the height function in the same direction as $\partial\sigma/\partial s^c$. For the example given in equation (3.9), the arc length is

$$\mathrm{d}s = 2\Delta\sqrt{1 + \mathcal{H}_y}, \tag{3.10}$$

where $\mathcal{H}_y$ is the derivative of the height function with respect to the y direction (see Section 2.2) and Δ is the cell size.

The next part of the surface gradient implementation is the choice of the tangent vector, $\hat{\mathbf{t}}$, which is computed so that it satisfies $\hat{\mathbf{t}} \cdot \hat{\mathbf{n}} = 0$, where $\hat{\mathbf{n}}$ is found using the Mixed Young's Center (MYC) method by [79]. We use the MYC method mainly due to its computational efficiency when compared to the height function method for computing the normals. This advantage is particularly noticeable in 3D. The direction of $\hat{\mathbf{t}}$ depends on the direction used for computing $\partial\sigma/\partial s^c$: $\hat{\mathbf{t}}$ points in the direction of the positive component orthogonal to the c direction. For example, $\hat{\mathbf{t}}$ points in the positive x direction if we construct columns in the y direction.

We have defined $\partial\sigma/\partial s^c$ in the interfacial cells. However, in the CSF method [23], we need to know $\partial\sigma/\partial s^c$ in the cells around the interface, i.e., in all the cells where $\delta_s = ||\nabla\chi||_2$ is nonzero. Consider an intermediate value of the surface force, $\mathbf{G}$, which includes the correct sign of the surface force, but excludes the magnitude of the tangential vector, i.e.,

$$G_x = \frac{\partial\sigma}{\partial s^c}\mathrm{sgn}(t_x), \tag{3.11}$$

$$G_y = \frac{\partial\sigma}{\partial s^c}\mathrm{sgn}(t_y). \tag{3.12}$$

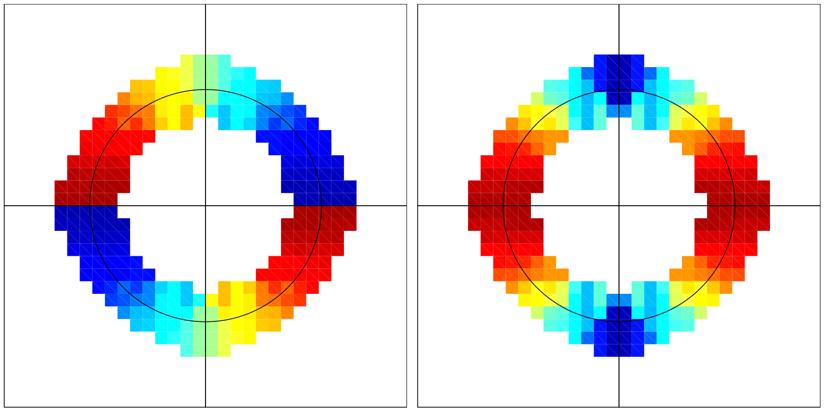

(a) (b)

Figure 3.5 The x (a) and y (b) component of $\mathbf{G}$, with values around the interfacial cells defined by averaging the neighboring cells.

The intermediate force $\mathbf{G}$ can be defined in the cells around the interface by using the same approach as for defining the curvature in the cells around the interface [78], i.e., the values in the cells neighboring the interfacial cells are defined by averaging the values in the direct neighbors that already have the curvature value defined. This procedure is repeated twice, insuring that the curvature values for the corner neighbors to the interfacial cells are defined as well. The x and y components of $\mathbf{G}$ in the cells around the interface are subsequently used to compute the surface force, defined in Equation (3.8), as

$$F_x = G_x |t_x| \, \delta_s, \tag{3.13}$$

$$F_y = G_y |t_y| \, \delta_s. \tag{3.14}$$

Figure 3.5 shows the result of this procedure for an example of the surface gradient where we impose a positive uniform gradient of the surface tension in the y direction.

Surface Force in 3D The implementation of the surface gradient in 3D extends the 2D implementation by considering the second tangential direction as stated in Equation (3.5). Equivalently as in 2D, we first define the column values $\tilde{\sigma}^c$ of the

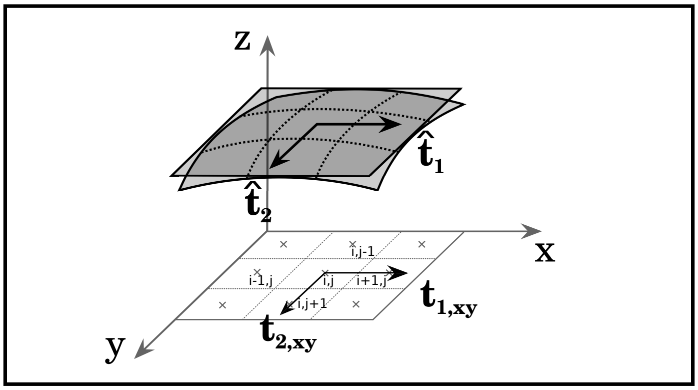


Figure 3.6 A stencil used for computing surface gradient in the column containing the cell $\mathcal{C}_{i,j}$, with a tangent plane defined by the vectors $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$. Vectors $\hat{\mathbf{t}}_{1,xy}$ and $\hat{\mathbf{t}}_{2,xy}$ are the projections of $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$ onto the xy-plane respectively.

surface tension coefficient σ. This part of the algorithm is identical to the 2D part, with the addition of one more direction. After the column values, $\tilde{\sigma}^c$, are defined, we compute the gradients along the two components orthogonal to the columns: for example, if the columns are constructed in the z direction, see Figure 3.6, then the derivatives along the interface are computed in the x and y directions as

$$\begin{aligned} \left(\frac{\partial \sigma}{\partial s_1^z}\right)_{i,j} &= \frac{\tilde{\sigma}^z_{i+1,j} - \tilde{\sigma}^z_{i-1,j}}{2\Delta\sqrt{1+\mathcal{H}_x^2}}, \\ \left(\frac{\partial \sigma}{\partial s_2^z}\right)_{i,j} &= \frac{\tilde{\sigma}^z_{i,j+1} - \tilde{\sigma}^z_{i,j-1}}{2\Delta\sqrt{1+\mathcal{H}_y^2}}. \end{aligned} \tag{3.15}$$

As previously discussed in 2D, the direction, c, in which the columns are constructed, is chosen based on the interface orientation, where c is the same as the direction of the largest component of the interface normal vector.

The next part of the surface gradient computation is the choice of the tangent vectors, $\hat{\mathbf{t}}_d$, which are computed so that they satisfy $\hat{\mathbf{t}}_d \cdot \hat{\mathbf{n}} = 0$. Among all the possibilities for $\hat{\mathbf{t}}_d$, we choose the two whose projections onto the coordinate plane, defined by all points with c coordinate equal to zero, are parallel to the axes. Figure 3.6 illustrates this procedure by an example where the columns are constructed in the z direction and the projections of the tangent vectors $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$ onto the x-y plane are parallel to the x and y axes and denoted by $\hat{\mathbf{t}}_{1,xy}$ and $\hat{\mathbf{t}}_{2,xy}$, respectively. In this particular example, the tangent vectors will be of the form

$$\hat{\mathbf{t}}_1 = (t_{1x}, 0, t_{1z}), \tag{3.16}$$

$$\hat{\mathbf{t}}_2 = (0, t_{2y}, t_{2z}). \tag{3.17}$$

The signs of the components of the tangential vectors are chosen so that their projections onto the coordinate plane point in the positive direction of the coordinate axes (see e.g., Figure 3.6).

Finally, we compute the surface force, $\mathbf{F}_{st} = (F_x, F_y, F_z)$. In the case such that the columns are constructed in the z direction, the components of $\mathbf{F}_{st}$ are

$$F_x = \frac{\partial\sigma}{\partial s_1^z} t_{1x}, \tag{3.18}$$

$$F_y = \frac{\partial\sigma}{\partial s_2^z} t_{2y}, \tag{3.19}$$

$$F_z = \frac{\partial\sigma}{\partial s_1^z} t_{1z} + \frac{\partial\sigma}{\partial s_2^z} t_{2z}. \tag{3.20}$$

Similarly as in the 2D case, in order to use the CSF formulation, the components of the tangential force need to be defined in the cells around the interface. This is done equivalently as in 2D, using the neighbor averaging procedure, see Section 3.2.2. However, in 3D, there is one extra step due to one of the components containing an addition of two terms, e.g., as in Equation (3.20). In order to illustrate this, consider the general form of the x component of the tangential force

$$F_x = \begin{cases} (\partial\sigma/\partial s_1^x) t_{1x} + (\partial\sigma/\partial s_2^x) t_{2x} & \text{if } c = x, \\ (\partial\sigma/\partial s_1^y) t_{1x} & \text{if } c = y, \\ (\partial\sigma/\partial s_1^z) t_{1x} & \text{if } c = z. \end{cases} \tag{3.21}$$

Similarly as in 2D, differences in the sign of the derivatives, $\partial\sigma/\partial s_d^c$, may arise from the choice of column directions. We proceed by defining the intermediate value of the surface force, $\mathbf{G}$. The components of $\mathbf{G}$ are computed equivalently as in 2D, except for the c component which is defined as

$$G_c = \frac{(\partial\sigma/\partial s_1^c)t_{1c} + (\partial\sigma/\partial s_2^c)t_{2c}}{\sqrt{t_{1c}^2 + t_{2c}^2}}, \tag{3.22}$$

where c is the direction of the columns. Now we can carry out the averaging procedure for each component of $\mathbf{G}$. Finally, the component of the force in the direction c is

$$F_c = G_c|t_{1c}|\delta_s + G_c|t_{2c}|\delta_s. \tag{3.23}$$

The other components are computed equivalently as in the 2D case.

3.3 Validation Results

3.3.1 Flat Film Geometry

We first present the validation of our methodology for computing the surface gradient in 2D geometry where we can compute the gradient exactly. The simplest geometry that we consider is a flat perturbed interface, i.e., let the interface be a function of x, defined as

$$h\left(x\right) = h_0 + \varepsilon\cos(2\pi x). \tag{3.24}$$

Let the surface tension coefficient be a function of the interface position, defined as

$$\sigma\left(h\right) = 1 + \sigma_h h\left(x\right). \tag{3.25}$$

Figure 3.7 shows the interface profile and surface tension coefficient at the interface, for $h_0 = 0.5$, $\varepsilon = 0.05$, and $\sigma_h = -10^{-2}$ in a computational domain of 1×1, with the symmetry boundary conditions imposed on all sides. In this case, apart from using the definition of the surface gradient given in Equation (3.4), we can also compute the exact surface gradient using the chain rule as

$$\nabla_s\sigma = \frac{\sigma_h\mathcal{H}_x}{\sqrt{1+\mathcal{H}_x^2}}\,\hat{\mathbf{t}}, \tag{3.26}$$

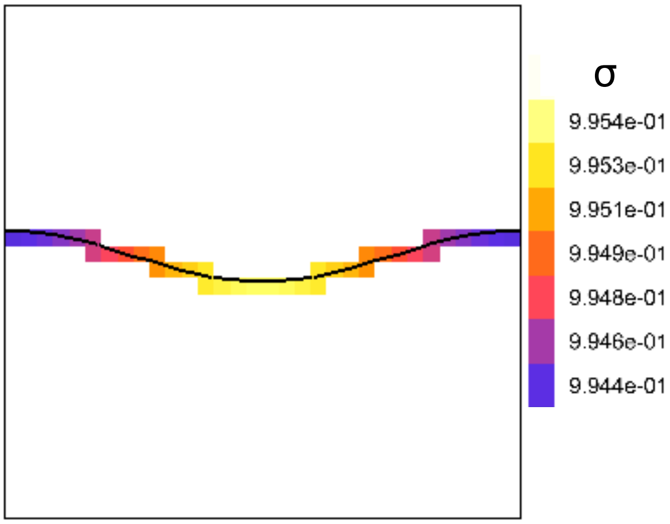


Figure 3.7 The setup of the perturbed interface with a surface tension coefficient dependent on the interface profile $h\,(x) = h_0 + \varepsilon \cos(2\pi x)$. The color represents the surface tension coefficient at the interface, with dark red and dark blue being the maximum and minimum values respectively.

where the unit tangent vector, $\hat{\mathbf{t}}$, is defined to point in the positive x direction as

$$\hat{\mathbf{t}} = \frac{(1, -\mathcal{H}_x)}{\sqrt{1+\mathcal{H}_x^2}}. \tag{3.27}$$

Note that the numerator in Equation (3.26), $\sigma_h \mathcal{H}_x$, is equivalent to the numerator of Equation (3.9); hence, we can compare their computed values to the exact ones directly. We present the errors associated with computing $\sigma_h \mathcal{H}_x$ and $\mathcal{H}_x$, separately, as well as each component of the surface gradient in Equation (3.26). We test the convergence as a function of the mesh size, Δ, using L_1 and L_∞ norms to define E_1 and E_∞ errors respectively as

$$E_1(f) = \frac{\sum_j^N |f_{\text{approx}} - f_{\text{exact}}|}{N}, \tag{3.28}$$

$$E_\infty(f) = \max |f_{\text{approx}} - f_{\text{exact}}|, \tag{3.29}$$

where the summation is over all N interfacial cells. The interface position in each cell can influence the errors obtained in constructing the columns for the computation

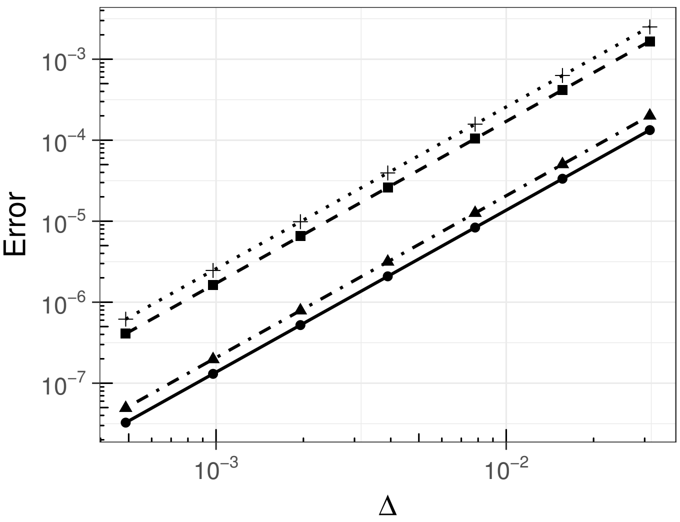


Figure 3.8 The computed errors for $E_1(\sigma_h \mathcal{H}_x)$ (●), $E_\infty(\sigma_h \mathcal{H}_x)$ (▲), $E_1(\mathcal{H}_x)$ (■), and $E_\infty(\mathcal{H}_x)$ (+). The order of convergence is 2 for $\sigma_h \mathcal{H}_x$ for both E_1 (–) and E_∞ (-·-) errors and for h_x for both E_1 (- -) and E_∞ (···) errors. The symbols represent the errors from the computations and the lines show the linear fits.

of both surface gradients, $\partial\sigma/\partial s^y$, and the derivative of the height function, $\mathcal{H}_x$. To avoid this error bias, we average the errors from 100 simulations where h_0 was modified to $\tilde{h_0} = h_0 + h_r$, where h_r is a random number in the interval $[0, \Delta]$ with uniform distribution.

We initialize the surface tension coefficient, σ, directly as a function of x, i.e., $\sigma\,(h) = 1 + \sigma_h(\tilde{h_0} + \varepsilon\cos(2\pi x))$. Figure 3.8 shows the convergence of the computed $\sigma_h \mathcal{H}_x$ as a function of the mesh refinement. As shown, the order of convergence is 2 for both E_1 and E_∞ errors. In this test case, the interfacial value of the surface tension coefficient $\tilde{\sigma}^y$ is computed in the y direction for all cells. Figure 3.8 also shows the order of convergence of $\mathcal{H}_x$, computed using height functions (see Section 2.2), that is also 2 for E_1 and E_∞ errors. We note that in order to eliminate errors in initializing the volume fractions, we prescribe the initial shape with a more refined mesh.

Next we investigate the accuracy of the computed surface gradient

$$\nabla_s \sigma\,(\mathbf{x}) = (\nabla_s \sigma)_x\,\hat{\imath} + (\nabla_s \sigma)_y\,\hat{\jmath},$$

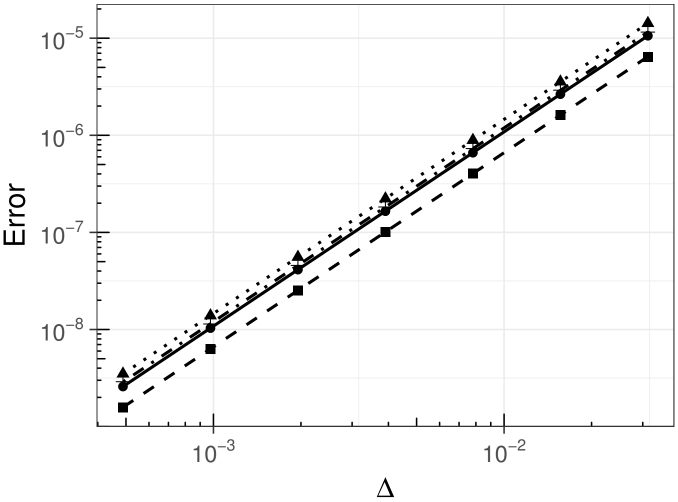


Figure 3.9 The computed errors for $E_1((\nabla_s\sigma)_x)$ (●), $E_\infty((\nabla_s\sigma)_x)$ (▲), $E_1((\nabla_s\sigma)_y)$ (■), and $E_\infty((\nabla_s\sigma)_y)$ (+). The order of convergence is 2 for $(\nabla_s\sigma)_x$ for both E_1 (–) and E_∞ (···) errors and for $(\nabla_s\sigma)_y$ for both E_1 (--) and E_∞ (-··-) errors. The symbols represent the errors from the computations and the lines show the linear fits.

where $((\nabla_s\sigma)_x, (\nabla_s\sigma)_y) = (G_x|t_x|, G_y|t_y|)$. Figure 3.9 compares the x and y components of the surface gradient with the exact solution. As shown, E_1 and E_∞ errors for both x and y components converge with second order.

3.3.2 Circular Geometry

Next we test the convergence for a more general interfacial geometry where the interfacial values of $\tilde{\sigma}^c$ are computed using columns in both x and y directions (see Section 3.2). We consider a circle of radius $a = 0.25$ positioned at $(0.5, 0.5)$ in a 1×1 domain with an imposed temperature distribution

$$T(x,y) = \Delta T\,(x+y)\,, \tag{3.30}$$

where ΔT is a constant. We assume that the thermal diffusivity is equal for the fluid inside and outside of the circle, i.e., $k_1 = k_2, C_{p1} = C_{p2}, \rho_1 = \rho_2, \mu_1 = \mu_2$, where the subscripts 1 and 2 denote the surrounding fluid, and the fluid inside of the drop, respectively. Figure 3.10 shows the setup with color representing the temperature field. Here we choose $\Delta T = 0.1$, $k_1 = 1$, $C_{p1} = 1$ and $\rho_1 = 1$. For simplicity, we assume that the surface tension coefficient is a linear function of temperature, i.e., $\sigma(T) = 1 + \sigma_T T$, where we let $\sigma_T = -0.1$. We set the velocity to zero, and knowing that the interface is exactly circular, we can compute the exact surface gradient from the definition

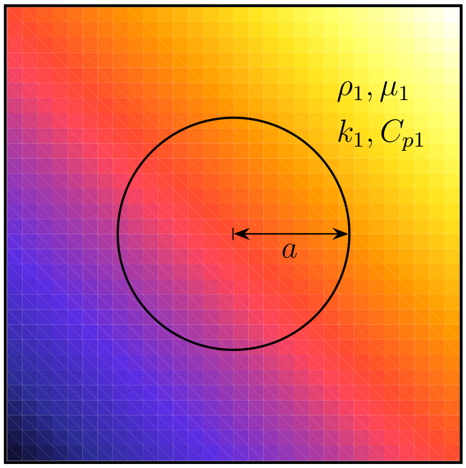


Figure 3.10 The initial setup of a circular drop with an imposed uniform temperature gradient. The color shows the temperature with dark blue and dark red being the minimum and maximum values respectively.

$$\begin{aligned}
\nabla_s \sigma &= \nabla \sigma - \hat{\mathbf{n}} \left(\hat{\mathbf{n}} \cdot \nabla \sigma \right), && (3.31)\\
&= \frac{\sigma_T \Delta T}{a^2} \left\langle a^2 - x^2 \mp x\sqrt{a^2 - x^2}, x^2 \mp x\sqrt{a^2 - x^2} \right\rangle, && (3.32)\\
&= \frac{\sigma_T \Delta T}{a^2} \left\langle y^2 \mp y\sqrt{a^2 - y^2}, a^2 - y^2 \mp y\sqrt{a^2 - y^2} \right\rangle. && (3.33)
\end{aligned}$$

Equations (3.32) and (3.33) give the surface gradient as a function of x and y, respectively.

We initialize the temperature following two approaches, and discuss their performance. The first approach is to define the interface as a function of x and y depending on the more favorable interface orientation as follows

$$T(x,y) = \begin{cases} \Delta T \left(x \pm \sqrt{a^2 - x^2} \right) & \text{if } |x| < |y|, \\ \Delta T \left(y \pm \sqrt{a^2 - y^2} \right) & \text{otherwise,} \end{cases} \qquad (3.34)$$

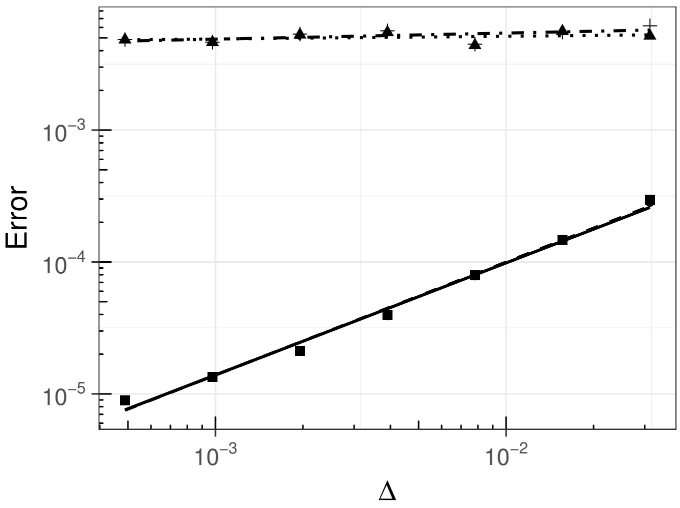


Figure 3.11 The computed error for $E_1((\nabla_s\sigma)_x)$ (●), $E_\infty((\nabla_s\sigma)_x)$ (▲), $E_1((\nabla_s\sigma)_y)$ (■), and $E_\infty((\nabla_s\sigma)_y)$ (+). The order of convergence for $(\nabla_s\sigma)_x$ is 0.85 and 0.021 E_1 (–) and E_∞ (···) errors respectively, and the order of convergence for $(\nabla_s\sigma)_y$ is 0.856 and 0.046 for E_1 (- -) and E_∞ (-··) errors, respectively. The symbols represent the errors from the computations, and the lines show the linear fit of those points.

where x and y are coordinates of the cell centers. The second approach is to use positions of the centroid of the interface contained in each cell to initialize the temperature by Equation (3.30). We show below that the second approach leads to more accurate results.

We compare the computed surface gradient with the exact solution by considering E_1 and E_∞ errors defined in Equations (3.28) and (3.29), respectively. Similarly as in the previous example, in order to eliminate the dependence of the errors on the interface position in the cell, the center of the drop is positioned randomly in the interval $[0, \Delta] \times [0, \Delta]$, and the errors are averaged over 100 random realizations. Figure 3.11 shows the convergence to the exact solution for the x and y components of the gradient. The convergence of the E_1 error is found to be of order 0.85 and 0.86 for the x and y components, respectively. The slow convergence of the E_∞ error observed in the figure is due to the errors in initializing the temperature at the lines $|x| = |y|$ from Equation (3.34), demonstrated later in this section.

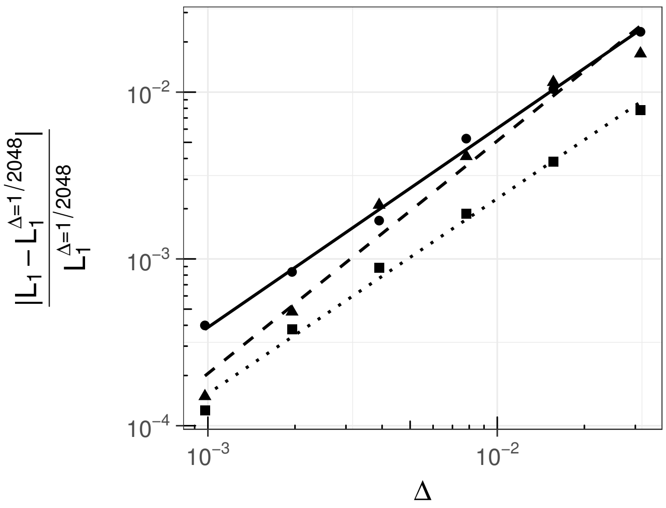


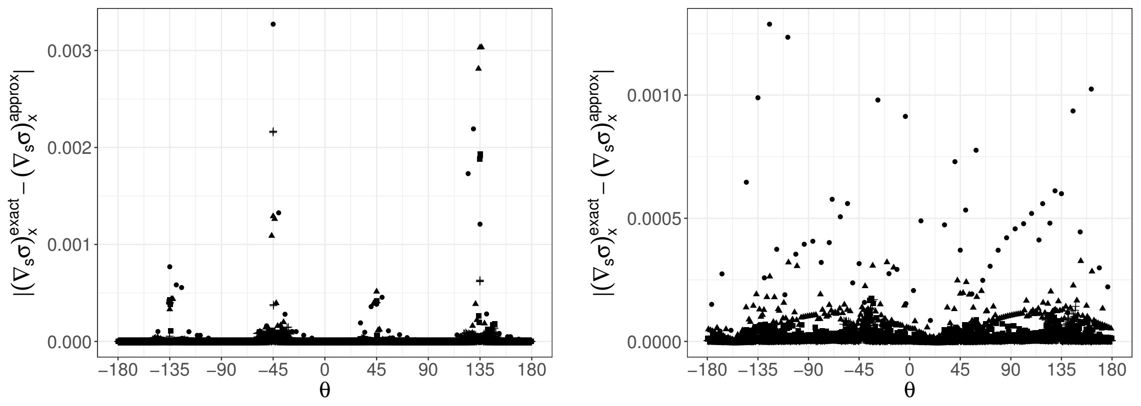
Figure 3.12 The computed errors for $L_1((\nabla_s\sigma)_x)$ (●), $L_1((\nabla_s\sigma)_y)$ (▲), and $L_1(T)$ (■). The order of convergence for $(\nabla_s\sigma)_x$ and $(\nabla_s\sigma)_y$ is 1.2 (–) and 1.4 (- -), respectively. The order of convergence of $T(x, y)$ at the interface is 1.2 (···). The symbols represent the errors from the computations and the lines show the linear fit of those points.

In order to reduce the influence of the initialization of T on the convergence, we also compute the convergence of the L_1 norm of each component of the surface gradient

$$L_1(f) = \frac{1}{N}\sum_i^N |f_i|, \tag{3.35}$$

where f_i is the x or y component of the surface gradient, and N is the number of interfacial points. Figure 3.12 shows the convergence of the L_1 norm for the x and y components of the surface gradient and the convergence of the temperature $T(x, y)$ in the interfacial points as a function of the mesh size, Δ. We find the order of convergence of the x and y components of the surface gradient to be 1.2 and 1.4. The order of convergence of $T(x, y)$ along the interface is 1.2. This indicates that the order of convergence of the surface gradient is limited by the order of convergence of the initial temperature at the interface.

Figure 3.13(a) shows the distribution of errors at the circular interface for one random realization. The largest errors appear around the lines $|x| = |y|$. Based



(a) (b)

Figure 3.13 Errors of the x component of the surface gradient at the interfacial cells for $\Delta = a/8$ (●), $a/32$ (▲), $a/128$ (■), and $a/512$ (+). Initializing the temperature using (a) Equation (3.34) and (b) Equation (3.36). θ is defined to be zero at the positive x axis and increasing counterclockwise.

on this we conclude that the lack of convergence of the E_∞ error is caused by the initialization of the temperature which changes the dependence on x or y variable at the lines $|x| = |y|$.

In order to initialize the temperature more accurately at the interface we use the centroid of the interface segment contained in each cell, (x_c, y_c). Then the initial temperature is given by

$$T(x, y) = \Delta T\,(x_c + y_c)\,. \tag{3.36}$$

This reduces the errors from initializing the temperature at the lines $|x| = |y|$ compared to using Equation (3.34). Here, we also explore a different way of approximating interfacial temperature, and use surface area weighted average instead of volume fraction weighted average (see Section 3.2). The volume weighted average gives the temperature at the center of the mass of the fluid phase in the column, whereas the surface area weighted average gives the temperature at the center of the interface in the column. Hence, the latter is consistent with the initialization of the temperature using Equation (3.36).

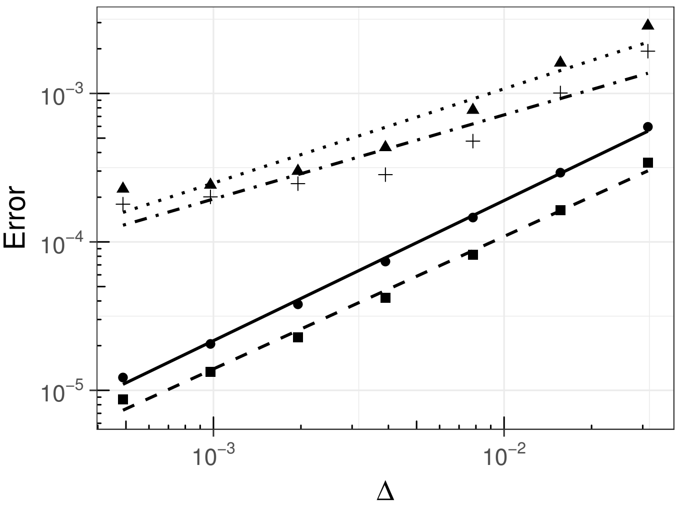


Figure 3.14 The computed errors for $E_1((\nabla_s\sigma)_x)$ (●), $E_\infty((\nabla_s\sigma)_x)$ (▲), $E_1((\nabla_s\sigma)_y)$ (■), and $E_\infty((\nabla_s\sigma)_y)$ (+). The order of convergence for $(\nabla_s\sigma)_x$ is 0.94 (–) and 0.63 (- -) for E_1 and E_∞ errors, respectively, and the order of convergence for $(\nabla_s\sigma)_y$ is 0.89 (···) and 0.57 (-··-), respectively. The symbols represent the errors from the computations, and the lines show the linear fit of those points.

Figure 3.13(b) shows the errors of the x component of the surface gradient at the interfacial cells if the temperature is initialized using Equation (3.36). The errors are still largest around $|x| = |y|$, however those are the usual "weak" spots of the height function construction [1]. Figure 3.14 shows the improvement in the convergence to the exact solution using L_1 and L_∞ norms for the x and y components of the surface gradient as a function of mesh refinement. The order of convergence for the L_1 norm is 0.94 and 0.89 for the x and y components of the surface gradient, respectively, and the order of convergence for the L_∞ norm is 0.63 and 0.57 for the x and y components of the surface gradient, respectively. Hence, the second approach of initializing the temperature (using Equation (3.36)) improves the convergence of the L_∞ norm significantly.

3.3.3 Drop Migration

We further test our numerical implementation using a classical problem of the thermocapillary drop migration (see the reviews [31, 80]). A drop or a bubble placed in a fluid with an imposed temperature gradient moves due to the variation in the surface tension coefficient as a function of temperature. Several authors have used this problem for benchmarking their numerical algorithms for a temperature dependent surface tension coefficient [2, 81, 3]. We show the comparison of our numerical results with the available work in the literature. We also show the comparison with the analytical solution of the drop terminal velocity by Young et al. [82]. Young et al. show that the nondimensional velocity of the drop in an unbounded domain for an axysimmetric geometry in the limit of small Ma and Ca numbers can be approximated as

$$v^*_{ygb} = \frac{\mu_1}{\sigma_T a \Delta T} \frac{2}{(2 + k_r)(2 + 3\mu_r)}, \tag{3.37}$$

where μ_1 is the viscosity of the surrounding fluid, σ_T is the (constant) gradient of the surface tension coefficient with respect to the temperature, a is the drop/bubble radius, ΔT is the imposed temperature gradient, and k_r and μ_r are the thermal conductivity and viscosity ratios, respectively, for the drop/bubble compared to the surrounding fluid.

In what follows, we choose a combination of Re, Ma, and Ca numbers to compare our results with those in [2, 81, 3], as well arbitrarily small Re, Ca, and Ma, while still computationally manageable, when comparing with the theoretical prediction above (Equation 3.37) which is valid in the limit of Re, Ma → 0.

Figure 3.15 shows the considered setup: a drop or a bubble of radius a is placed in an ambient fluid, with a linear temperature gradient imposed in the y direction. The temperature at the top and the bottom boundaries is set to constant values and a zero heat flux boundary condition is imposed at the left and right boundaries. The boundary conditions for the flow are no-slip and no penetration at the top and bottom boundaries and symmetry at the left and right boundaries.

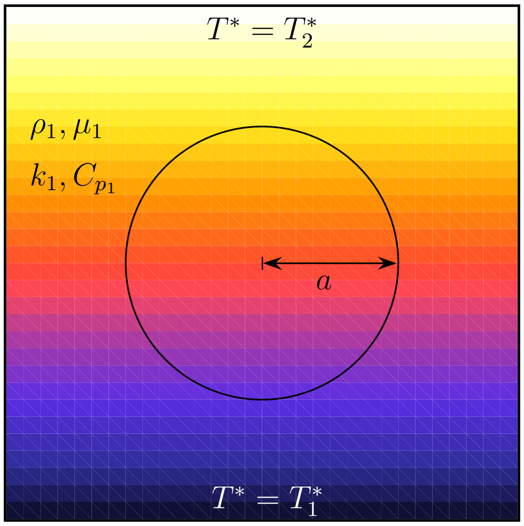


Figure 3.15 The initial setup of the drop migration problem. The color represents the linear temperature distribution with imposed temperatures T_1^* and T_2^* at the horizontal boundaries.

We solve Equations (2.10) and (2.15) and consider the following scales

$$p_0 = \frac{\mu_1 U_0}{a}, \quad t_r = \frac{a}{U_0}, \quad U_0 = \frac{\sigma_T a \Delta T}{\mu_1} \quad T_0 = a\Delta T,$$

where the subscript 1 denotes the properties of the ambient fluid. The surface tension at the interface between the drop and the ambient fluid is assumed to depend linearly on the temperature as given by Equation (2.20), which with the above scales yields

$$\sigma^* = 1 + \mathrm{Ca}\,(T^* - T_R^*)\,. \tag{3.38}$$

Next we present the comparison of our results with the available studies in the literature.

We start by comparing our results with those of Ma & Bothe [2]. The material properties are $\rho_1 = 500\,\mathrm{kg\ m^{-3}}$, $\mu_1 = 0.024\,\mathrm{Pa\ s}$, $\sigma_0 = 10^{-2}\,\mathrm{N\ m^{-1}}$, $\sigma_T = 2 \times 10^{-3}\,\mathrm{N\ m^{-1}K^{-1}}$, $k_1 = 2.4 \times 10^{-6}\,\mathrm{W\ m^{-}1K^{-1}}$, $C_{p_1} = 10^{-4}\,\mathrm{J\ kg^{-1}K^{-1}}$, $\Delta T = 200\,\mathrm{K\ m^{-1}}$, $T_2 = 290\,\mathrm{K}$, $a = 1.44 \times 10^{-3}\,\mathrm{m}$. The ratio of the material properties between the ambient fluid and the drop is 2. These physical properties give the nondimensional parameters Re = Ma = 0.72, Ca = 0.0576, and the velocity scale

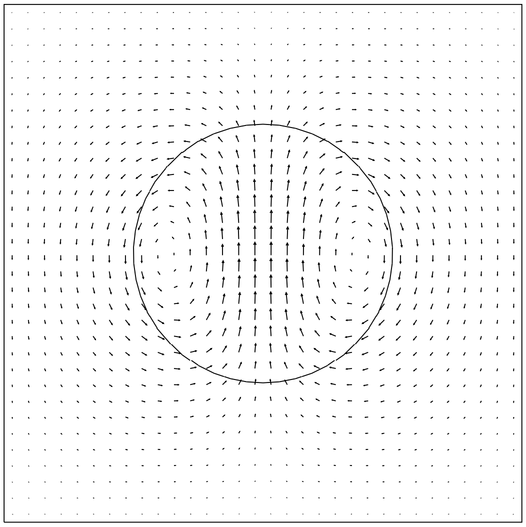


Figure 3.16 The velocity field in the drop and the surrounding fluid.

$U_0 = 0.024\,\mathrm{m\ s^{-1}}$. Figure 3.16 shows the computed velocity field in the drop and the surrounding fluid. The gradient of the surface tension coefficient drives the flow from regions of low (top) to high (bottom) surface tension. This creates the flow inside the drop and as a result the drop moves in the positive y direction. The drop velocity is computed using the following definition of the centroid velocity

$$v_c^* = \frac{\sum\limits_{i,j} v_{i,j}^* \chi_{i,j} \, (\Delta_{i,j}^*)^2}{\sum\limits_{i,j} \chi_{ij} \, (\Delta_{i,j}^*)^2}$$

where $v_{i,j}^*$ is the y component of the cell-center velocity.

Figure 3.17 shows the computed velocity of the drop compared to the results in [2]. In this test case, the computational domain is a square box with a side length equal to four times the drop radius; the drop is initially placed at the center of the domain. As shown, a good agreement is obtained with the result in [2], corresponding to $\Delta = 1/32$ in our results. We also note a decay in the migration velocity at a later time. This effect is due to fact that the drop is getting closer to the top boundary. This decay also exists in the simulation of [2], although it is less pronounced. We have checked and noticed that the migration velocity does not exhibit a decay when

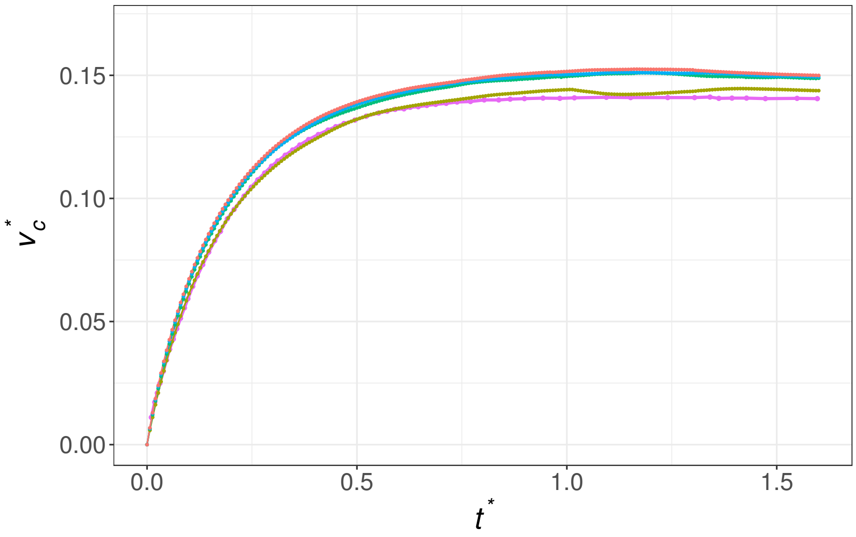


Figure 3.17 The drop migration velocity for Re = Ma = 0.72 and Ca = 0.0576 for $\Delta = 1/32$ (–), 1/64 (–), 1/128 (–), and 1/256 (–) compared with the result given in [2] (–) - with the resolution corresponding to $\Delta = 1/32$ in our results - for 2D simulations.

extending the vertical size of the domain. We do not show these results here for brevity. We must note here that for a mesh resolution below $\Delta = 1/32$, i.e., 8 grid points per drop radius, a consistent height-function approximation is no longer possible and we cannot expect accurate solutions at such low resolutions.

Next, we carry out another comparison for a smaller value of Re and Ca numbers and when Ma = 0; we choose Re = Ca = 0.066 in accordance with the results presented by Herrmann et al. [3] for the VoF method. The computational box is a rectangle of size $10a \times 15a$. The grid size is $\Delta = 5a/512$. The density of the ambient fluid is set to $\rho_1^* = 0.2$, and the viscosity is $\mu_1^* = 0.1$. The ratio of the physical properties of the drop to those of the ambient fluid is set to 1. The surface tension coefficient is $\sigma_0^* = 0.1$ at the reference temperature $T_R^* = 0$, with $\sigma_T^* = -0.1$, see Equation (2.20). The temperature gradient is set to $\Delta T^* = 0.1\bar{3}$, which is fixed by setting $T_1^* = 0$ and $T_2^* = 1$. The drop is initially centered horizontally at a distance $3a$ from the bottom wall. Figure 3.18 shows the comparison of our method with the results in [3], along with the temporal convergence of our method. Compared to the

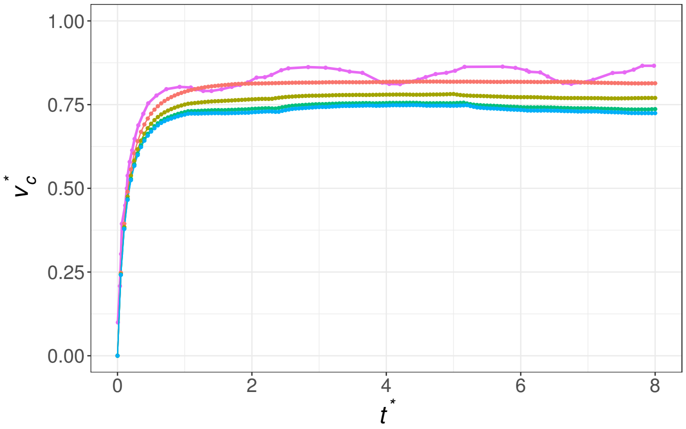


Figure 3.18 The convergence of the migration velocity as a function of the time step for $\Delta t = 10^{-4}$ (–), 5×10^{-6} (–), 10^{-5} (–), and 5×10^{-6} (–) compared with the results in [3] (–) for 2D simulation, for Re = Ca = 0.066 and Ma = 0. The velocity is rescaled by v^*_{ygb}. The grid size is $\Delta = 5a/512$.

results of [3], our results do not exhibit oscillations, which agrees with the asymptotic solution of constant rise velocity. Another difference is that our terminal velocity converges to a smaller value with decreasing time step. However, the timestep used in the results of [3] is not specified.

We also test the convergence to the analytical solution obtained in the limit of Ma and Re approaching zero and in the unbounded domain, where the terminal velocity approaches v^*_{ygb} value given in Equation (3.37). Figure 3.19 shows the terminal velocity of a droplet for a 2D simulation as a function of a distance from the wall for $\text{Re} = \text{Ma} = 2.5 \times 10^{-3}$ and $\text{Ca} = 1.25 \times 10^{-3}$. The grid size here is $\Delta = a/32$, except for the case $h_* = 1.666$, where $\Delta = a/48$. Note that the Ca number is chosen small enough to prevent droplet oscillations, and it does not affect the migration velocity. The terminal velocity converges to a value lower than v^*_{ygb} due to the difference in the geometry. We next show that our 3D result in fact converges to this analytical solution.

We perform similar tests for the 3D simulations. Figure 3.20 shows the migration velocity for $\text{Re} = \text{Ma} = 0.72$ and $\text{Ca} = 0.0576$. The parameters and the

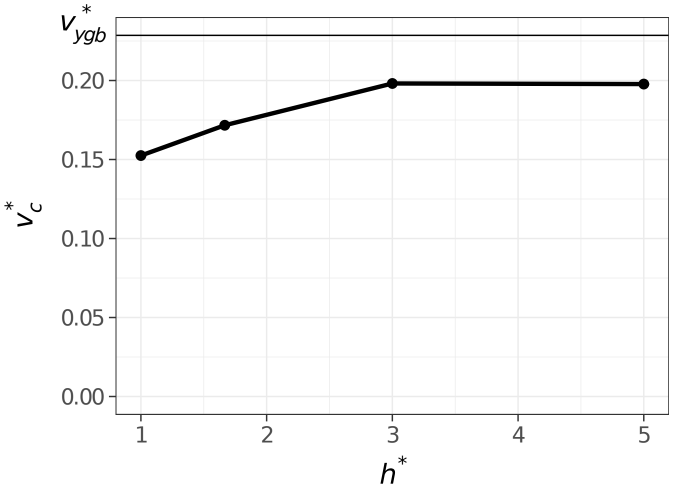


Figure 3.19 The convergence of the terminal velocity with an increased distance from the wall, h^*, for $\mathrm{Re} = \mathrm{Ma} = 2.5 \times 10^{-3}$ and $\mathrm{Ca} = 1.25 \times 10^{-3}$ for 2D simulations. The grid size is $\Delta = a/32$, except for $h_* = 1.666$ case where $\Delta = a/48$.

domain size are equivalent to the simulation results shown in Figure 3.17. We note again that for a mesh resolution below $\Delta = 1/64$, i.e., 16 grid points per drop radius, a consistent height-function approximation is no longer possible, and we cannot expect accurate solutions at such low resolutions; such a result for $\Delta = 1/64$ (–) is shown in Figure 3.20. (A consistent construction of height functions fails at a higher resolution in 3D than in 2D, see e.g., [1] and references therein.) Also as in 2D, we note a decay in the migration velocity at a later time. We have checked and this decay does not exists in the simulation when extending the vertical size of the domain. The results also show that the oscillations in the computed velocity decay with mesh refinement and the terminal velocity converges to a higher value compared to the 2D case. However, this value is still smaller than v^*_{ygb} due to the small domain size and relatively large Re and Ma numbers.

Figure 3.21 shows the terminal velocity of a droplet for a 3D simulation as a function of a distance from the wall for $\mathrm{Re} = \mathrm{Ma} = 2.5 \times 10^{-3}$ and $\mathrm{Ca} = 10^{-3}$. As shown, the terminal velocity approaches the analytical value v^*_{ygb}. The grid size here is $\Delta = a/16$, except for the case $h_* = 1.666$, where $\Delta = a/24$.

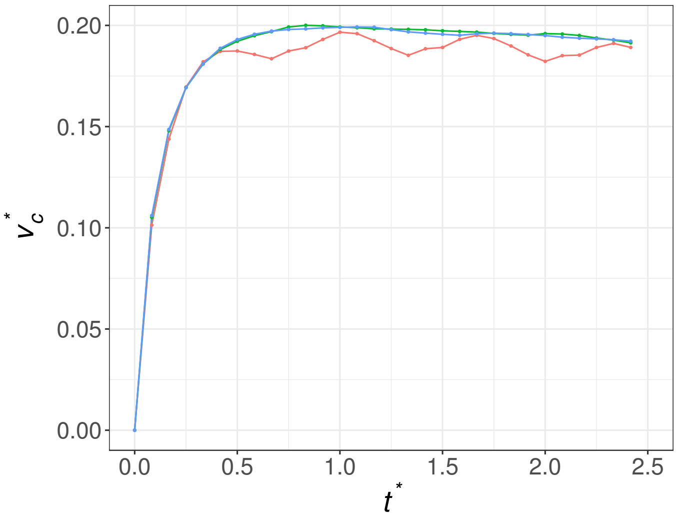


Figure 3.20 The convergence of the migration velocity in a 3D simulation with mesh refinement for $\Delta = 1/32$ (—), $1/64$ (—), and $1/128$ (—); $\mathrm{Re} = \mathrm{Ma} = 0.72$ and $\mathrm{Ca} = 0.0576$.

In this section, we have shown the comparison of our method with existing literature and with a limiting analytical solution. Our method shows the convergence to the analytical value of the terminal velocity. The trend of the obtained solutions under grid refinement, as well as the time needed to reach the terminal velocity are consistent with the previously reported results.

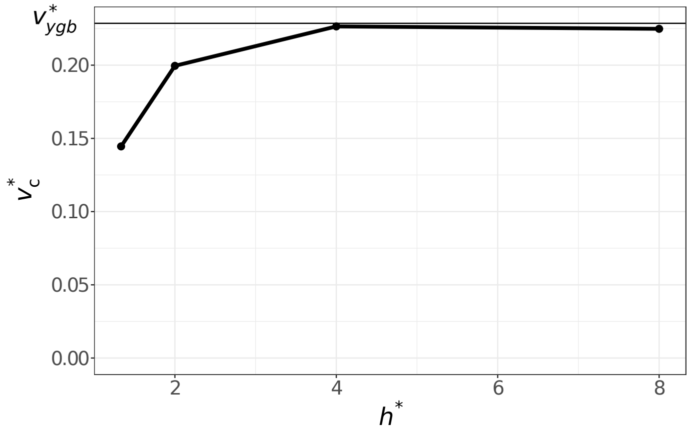


Figure 3.21 The convergence of the terminal velocity with increased distance from the wall, h^*, for $\mathrm{Re} = \mathrm{Ma} = 2.5 \times 10^{-3}$ and $\mathrm{Ca} = 10^{-3}$ for 3D simulations. The grid size is $\Delta = a/16$, except for the case $h_* = 1.666$, where $\Delta = a/24$.

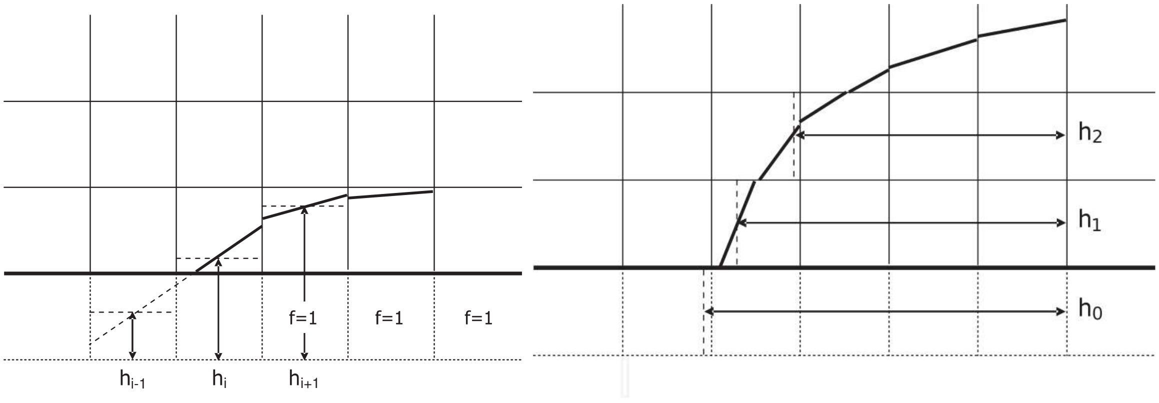
CHAPTER 4

CONTACT ANGLE

A contact angle is the angle which the fluid-fluid interface makes with a solid phase substrate. In VoF methods, the contact angle is imposed using geometrical methods, which ensure that the reconstructed interface reflects the imposed angle. One method to do this is based on the height function method developed in [48]. This method modifies the height functions at the contact line, which are used in the calculations of curvatures and normal vectors at the interface (see Section 2.2), and is currently implemented in Gerris flow solver only in 2D. However the extension of this algorithm to 3D is not trivial. Therefore, we propose and implement in 2D, an algorithm which can be more simply extended to 3D.

In order to implement the height function method for imposing contact angle boundary condition in 3D, we use an older version of the algorithm for computing the curvature at the interface using the height function approach. Gerris flow solver contains two algorithms for computing curvature and normals at the interface. The newer algorithm includes the implementation of the contact angle boundary condition. However, this algorithm is only implemented in 2D, due to its nontrivial extension to 3D for an adaptive mesh. The old algorithm for curvature computation is already implemented in 3D: therefore the remaining part is to impose the contact angle boundary condition.

The height function approach for imposing the contact angle at the solid boundary modifies the height function in the cells near the contact line [48]. The value of the height function is computed with two constraints: it preserves the volume fraction value in each cell, and it ensures that the normal vector at the contact line is consistent with the imposed contact angle. This height function value is then used for calculating the curvature at the contact line, which enters in the computation of the surface tension force.



(a)

(b)

Figure 4.1 Height function in a ghost cell in (a) vertical and (b) horizontal direction. *Source: [4].*

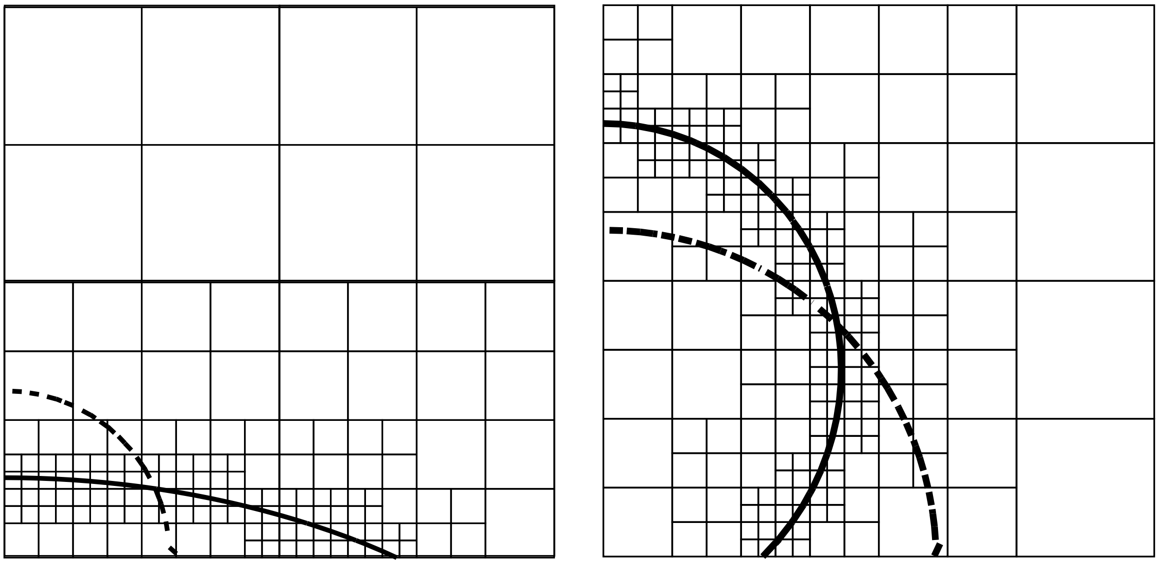
The height function at the contact line is computed differently depending on whether the height functions are collected in the direction normal or parallel to the solid boundary. For the height function normal to the solid boundary (see Figure 4.1 (a)) the height of both the contact line and the cell to the left, need to be modified by the boundary condition. Height h_i is computed so that the volume fraction of the ghost cell in the boundary reflects the direction of the contact angle θ, and height h_{i-1} is

$$h_{i-1} = h_i - \Delta \tan\theta \tag{4.1}$$

where Δ is the cell size. For height functions parallel to the solid boundary, Figure 4.1(b), the value of the height in the ghost cell is

$$h_0 = h_1 + \frac{\Delta}{\tan\theta}. \tag{4.2}$$

This implementation relies on the use of the height function algorithm, which defines heights in all cells first and then computes the curvature and the normals at the interface. Note that for an adaptive mesh, an interfacial cell may not have the same size as its neighbor, and heights are computed relative to each cell's size. In order to calculate the curvature at the cell center of each interfacial cell, an interpolation of the height function values in the neighboring cells is used, when calculating its derivatives in equations (2.33) [11]. The extension of this method to 3D is nontrivial for an adaptive mesh. Consequently, the curvature computation using this method has not been extended to 3D so far.



(a)

(b)

Figure 4.2 2D drops in equilibrium with imposed contact angle (a) $\theta = 25°$ and (b) $\theta = 135°$. The substrate is at the bottom boundary. The dashed line shows the initial condition for both simulations is $\theta = 90°$.

In order to overcome the difficulty described above, we implement a variation of the height function method, which computes the height functions in the neighboring cells relative to the cell size of the central cell, i.e., the cell for which we want to compute the curvature. For example, if the neighboring cell is larger than the center cell, then the height function will be calculated using a virtual cell of the same size as the center cell. This algorithm is easier to extend to 3D.

Figure 4.2 shows drops in the equilibrium using our new implementation with imposed contact angle (a) $\theta = 25°$ and (b) $\theta = 135°$. In both simulations, the initial condition is a semicircular drop with 90° contact angle. The homogeneous Neumann boundary condition is imposed on all boundaries for the pressure, and on all except the bottom boundary for the velocity field, where a no-slip and no penetration boundary condition is used. Note that the height function method implicitly defines a slip condition at the contact line with slip length of $\Delta/2$ [48].

The presented implementation in 2D provides the basis for extension to 3D, where conceptually the same approach could be used, although the geometrical complexity makes the algorithm more involved. This extension is left for future work.

CHAPTER 5

APPLICATIONS

5.1 Coalescence and Mixing of Sessile Drops

Here we demonstrate the performance of our numerical methods through an example of the coalescence behavior of sessile drops with different surface tension. We model the experiments of the coalescence of two droplets with different alcohol concentrations by Karpitschka et. al. [5, 83, 84]. In their experimental study, they show three coalescence regimes depending on the surface tension difference between the two droplets: immediate coalescence, delayed coalescence, and non-coalescence. They identify a key parameter that governs the transition between the delayed and non-coalescence regimes: specific Marangoni number $\mathrm{M} = 3\Delta\sigma/(2\bar{\sigma}\theta^2)$ [84], where $\Delta\sigma$ is the difference in the surface tension between the two drops and $\bar{\sigma}$ is the average of the surface tension of two drops. They determine a threshold Marangoni number $\mathrm{M}_t \approx 2 \pm 0.2$ experimentally for the transition between the delayed coalescence and non-coalescence regimes.

Here we show that our numerical simulations also reveal the three regimes in agreement with the experimental observations in [5, 83]. From the numerical simulation point of view, this problem involves a level of difficulty: unlike temperature, the concentration should remain strictly confined to the liquid phase and should not leak out to the ambient phase. To avoid this difficulty, we combine our variable surface tension methodology with the numerical technique already implemented in the original version of GERRIS[78] which prevents the concentration from leaking out of the liquid domain into the ambient phase.

We model the 2D problem since the dominant flow dynamics in the problem is in the region connecting the two droplets, where the surface tension gradient is the strongest, and in this region we can ignore the out of plane curvatures. Initially, the

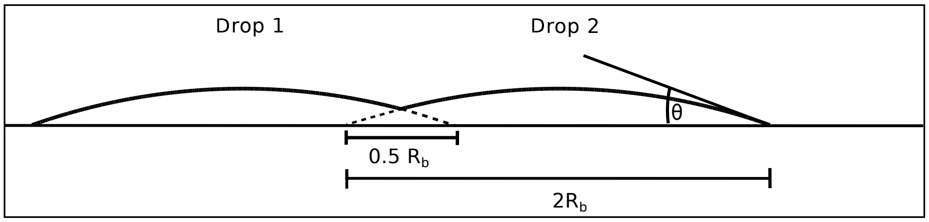
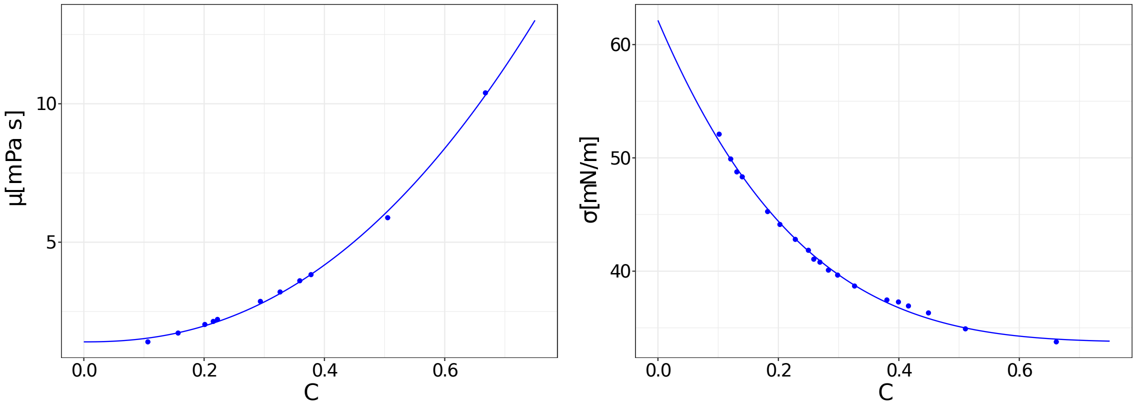


Figure 5.1 A schematic of the drop coalescence problem.



(a)

(b)

Figure 5.2 Viscosity (a) and surface tension (a) of the mixture of 1,2-Butanediol and water as a function of alcohol concentration. Points represent the data from [5], and the lines show the fit of the points.

drops have the shape of a circular segment with the base radius R_b and a contact angle θ, and are connected by an overlap of $0.25R_b$ (see Figure 5.1). The drops have equal base radius R_b and we assume that their densities are equal. The viscosity depends on the alcohol concentration C, where we use a nonlinear fit to the data given in [5] of the form

$$\mu\left(C\right) = \mu_1 + a_\mu\left(\mu_2 - \mu_1\right)\left(1 - C\right)^{n_\mu}, \tag{5.1}$$

shown in Figure 5.2(a). Drops are composed of the mixture of the 1,2-Butanediol and water, but they differ in the concentrations of alcohol. Figure 5.2(b) shows the surface tension dependence on the concentration of 1,2-Butanediol in water. Similarly

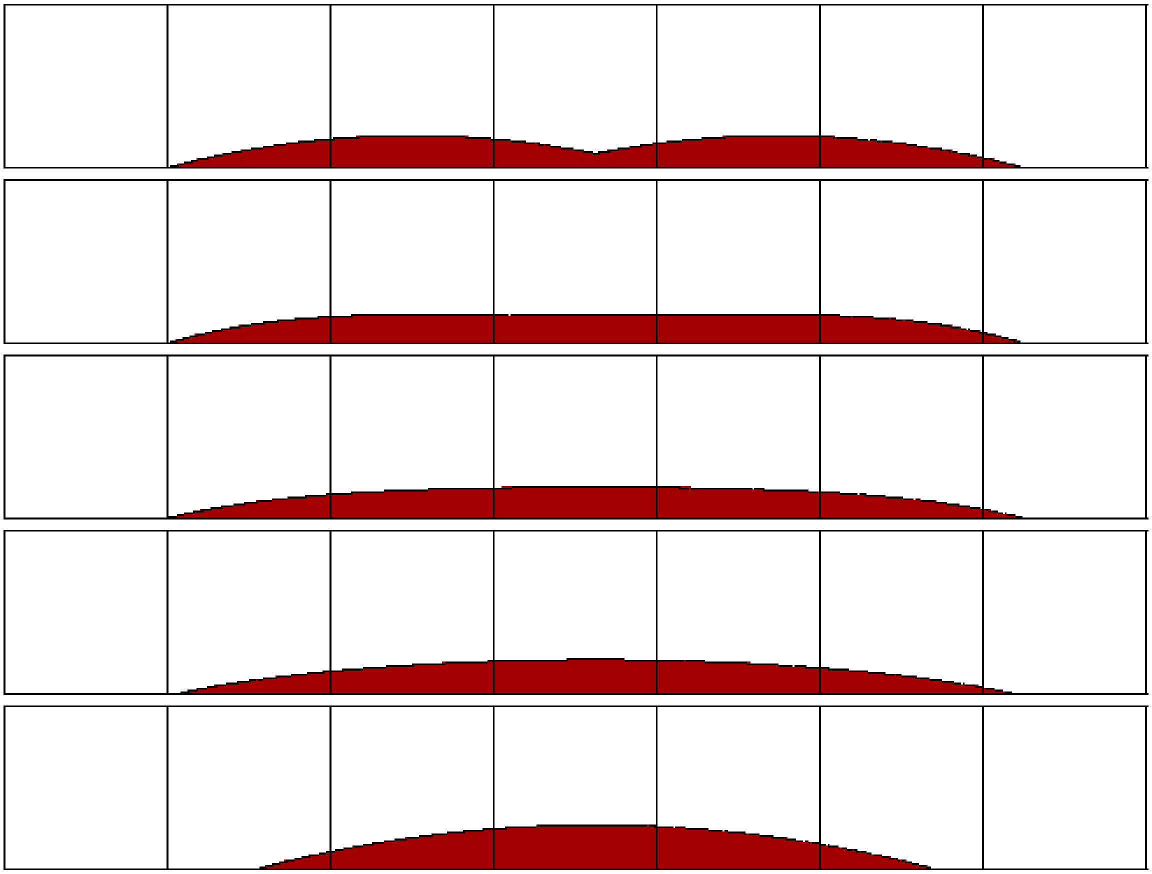


Figure 5.3 Evolution of two drops with equal alcohol concentration, i.e., no surface tension difference, at times $t = 0\,\mathrm{s}$, $t = 0.02\,\mathrm{s}$, $t = 0.04\,\mathrm{s}$, $t = 0.1\,\mathrm{s}$ and $t = 1\,\mathrm{s}$ from top to bottom. The color shows the concentration of alcohol. Each box is equivalent to 2 mm.

as for the viscosity, we fit this data to a function of the form

$$\sigma\left(C\right) = \sigma_1 + a_\sigma\left(\sigma_2 - \sigma_1\right)\left(1 - C\right)^{n_\sigma}, \tag{5.2}$$

where parameters a and n are determined from the fit.

We first show a simulation of two drops, with equal surface tension. We consider the case where the concentration of alcohol is 45%, and the base radii of the circular segments are both $R_b = 3\,\mathrm{mm}$. Along with a no-slip boundary condition at the substrate, we also impose a $\theta = 15^\circ$ contact angle. For the contact angle implementation in GERRIS and related numerical discussion, the reader is referred to [4, 85]. Figure 5.3 shows the evolution of the interface at different times. The droplets

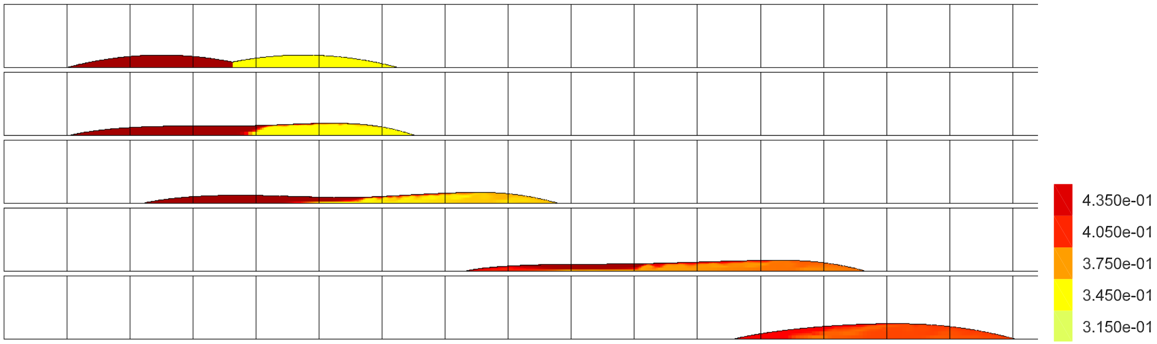


Figure 5.4 Evolution of two drops with small surface tension difference at times $t = 0\,\mathrm{s}$, $t = 0.1\,\mathrm{s}$, $t = 1\,\mathrm{s}$, $t = 5\,\mathrm{s}$, and $t = 10\,\mathrm{s}$ from top to bottom. The color shows the concentration of alcohol. Each box is equivalent to 2 mm.

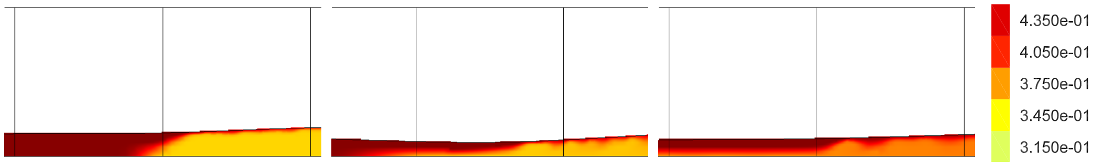


Figure 5.5 Closeup of the neck region between the two drops shown in Figure 5.4, at times $t = 0.1$, s, $t = 1\,\mathrm{s}$, and $t = 5\,\mathrm{s}$ from left to right. The color shows the concentration of alcohol.

coalesce immediately, fully merge after 0.1 s, and assume an equilibrium shape of one large circular segment at a later time. The color represents the concentration of alcohol, which is contained inside of the fluid and is zero in the surrounding region.

Next we examine the case where $\mathrm{M} \approx 1.2 < \mathrm{M}_t$. Figure 5.4 shows the simulations of this intermediate regime where droplet coalescence is delayed. Here, we set drop 1 to 45% and drop 2 to 35% of alcohol. The connected drops move in the direction toward the drop with higher surface tension due to the Marangoni induced flow until the concentrations are mixed, resulting in a smaller gradient in the surface tension. Figure 5.5 shows closeup images of the neck region between the two drops corresponding to the three middle panels of Figure 5.4. In this figure, we show the flow mixing dynamics which leads to the decrease of the surface tension difference in the neck region, resulting in a consequent full coalescence of the two drops.

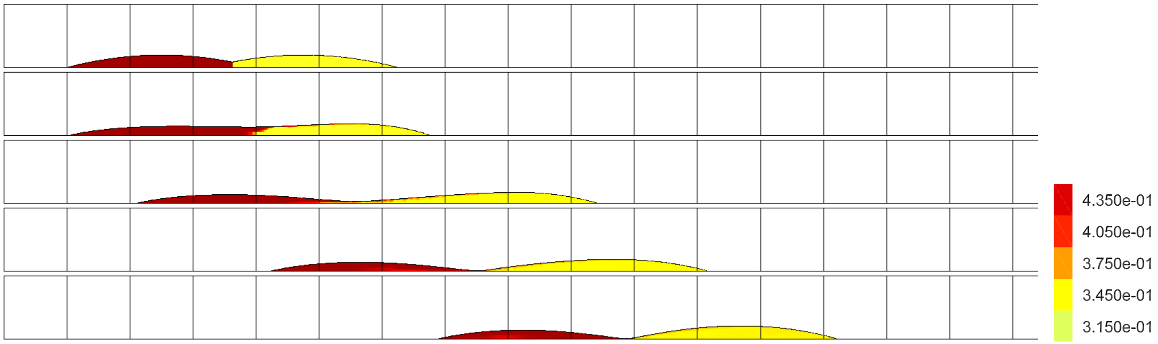


Figure 5.6 Non-coalescence of drops at times $t = 0\,\mathrm{s}$, $t = 0.1\,\mathrm{s}$, $t = 1\,\mathrm{s}$, $t = 2\,\mathrm{s}$, and $t = 6\,\mathrm{s}$ from top to bottom. The color shows the concentration of alcohol. Each box is equivalent to 2 mm.

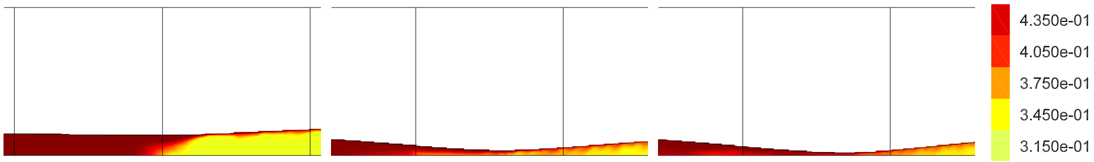
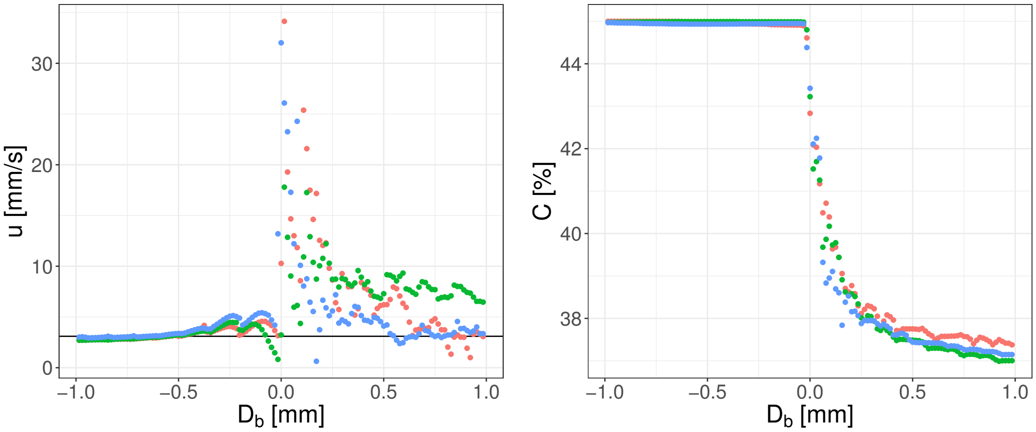


Figure 5.7 Closeup of the neck region between the two drops shown in Figure 5.6, at times $t = 0.1\,\mathrm{s}$, $t = 1\,\mathrm{s}$, and $t = 2\,\mathrm{s}$ from left to right. The color shows the concentration of alcohol.

Next we consider a case in the non-coalescence regime. We set drop 1 to 45% and drop 2 to 33% of alcohol. Figure 5.6 shows the simulation results for $\mathrm{M} \approx 1.8 \approx \mathrm{M}_t$. In this case, the Marangoni induced flow initially pushes the fluid from drop 1 towards drop 2. However, this results in the thinning of the connecting neck between the drops (at $t = 1\,\mathrm{s}$), and the fluid cannot pass from drop 1 to drop 2 anymore. Figure 5.7 shows closeup images of the neck region between the two drops corresponding to the middle three panels shown in Figure 5.6. Compared to the previous case where droplet coalescence is delayed ($\mathrm{M} \approx 1.2$), the behavior of the mixing of the fluids in the neck region is prevented by the thinning of the neck. Hence, these droplets do not coalesce, but instead they move together with a constant velocity u_d on the substrate in the direction of the higher surface gradient.

This quasi-steady behavior is also observed in the experiments [83]. Figure 5.8(a) shows the velocity of the points at the interface after the quasi-steady state is



(a) (b)

Figure 5.8 The x component of the velocity field at the interface interfacial points (a) and the concentration of alcohol at the interfacial points (b) for the non-coalescent drops for the example shown in Figure 5.6. The color shows different times, with red, green and blue being $t = 4\,\mathrm{s}$, $t = 5\,\mathrm{s}$, and $t = 6\,\mathrm{s}$, respectively.

reached as a function of the distance from the bridge region, D_b. The points to the left of the bridge region have a velocity $\approx 2u_d$ (solid line). At the bridge region the interface is close to the solid substrate and the velocity becomes close to zero due to the no-slip boundary condition. In the region close to the bridge in drop 2, the velocity has a jump and reaches the maximum value due to the Marangoni effect resulting from a high surface tension gradient at the neck region. Away from the bridge, the velocity is again comparable to u_d. This behavior is in qualitative agreement with the experimental observation [83, 84]. To provide more insight into the flow through the neck region, in Figure 5.8(b) we present the alcohol concentration at the interface as a function of the distance from the bridge region, D_b. As shown, a localized and steady state surface tension gradient is established through the neck region. This Marangoni effect can counteract the capillary effect that would otherwise result in the coalescence and can therefore sustain the non-coalescence and the movement of drops temporarily.

5.2 Breakup of Liquid Metal Filaments with Marangoni Effects

In this section, we consider a breakup of bi-metal filaments deposited on a solid substrate. These filaments are exposed to laser irradiation and, while in the liquid phase, evolve by a process resembling breakup of a liquid jet governed by the Rayleigh-Plateau (RP) instability. The Marangoni effect, resulting from a different surface tension coefficient of the two metals from which the filament is built, is crucial in understanding the instability development. In particular, the Marangoni effect may lead to the inversion of the breakup process, producing droplets at the locations where according to the RP theory dry spots would be expected. These results suggest the possibility of using the Marangoni effect for the purpose of self- and directed-assembly on the nanoscale.

In section 5.2.1, we present the experimental results carried out by our collaborators at the University of Tennessee: Chris A. Hartnett, Kyle Mahady, Jason D. Fowlkes and Philip D. Rack. In section 5.2.2, we show the results from the direct numerical simulations, using methods described in Chapters 2 and 3.

5.2.1 Experimental Methods

Initially, thin film strips (TFS) of Nickel (Ni) with thin Copper (Cu) layers on top are patterned on a supporting substrate of 100 nm SiO_2 on Si. The Ni-Cu system is chosen since it is an isomorphous system with complete solubility in the solid and liquid state; thus secondary phase formation and reactions do not complicate the physics or mass transport. The results presented here contain TFS of width $w = 185\,\text{nm}$, and height $h = 12\,\text{nm}$. The top Cu layers are of height 2 nm. Note that both the experiments and the simulations are also performed with $w = 370\,\text{nm}$, yielding the same qualitative behavior as the presented results.

For comparison, each experiment includes a non-perturbed Ni TFS and a Ni TFS with a co-linear Cu TFS with half of the Ni width patterned on it. Figure 5.9(a)

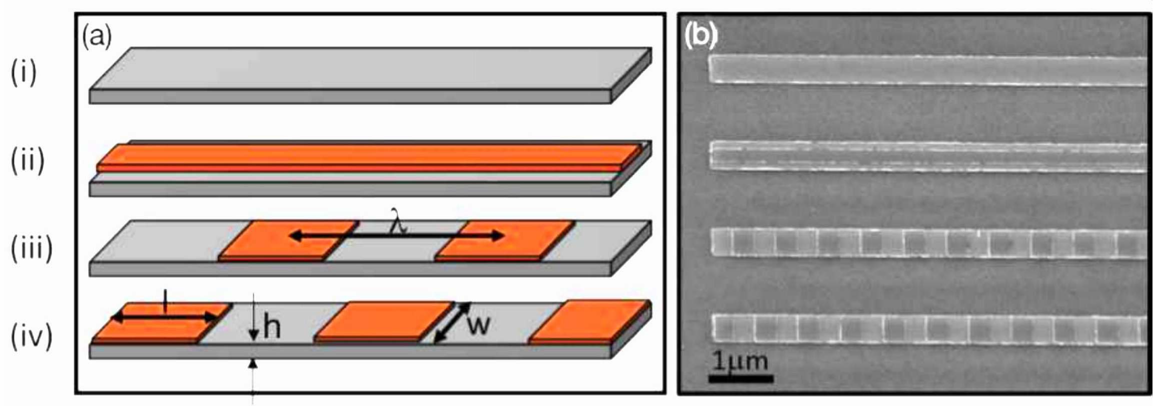


Figure 5.9 (a) Illustration and (b) SEM image of the experimental TFS geometry setup. Strip (i) is a pure Ni TFS, (ii) is Ni TFS with a Cu strip patterned on top along the centerline, at half the width of the Ni strip, iii) is a Ni strip with Cu patterned at a wavelength (λ_m) with the same width as the Ni but only half of the wavelength. Strip (iv) is similar to (iii) however the Cu pattern is phase shifted for $\lambda_m/2$ relative to strip (iii) (Cu in pattern (iv) is aligned with Ni-only regions in strip (iii)). The Ni and Cu regions in (b) are distinguished by the difference in the brightness, where Ni is brighter than the Cu.

shows an illustration of the experimental setup and geometry. Figure 5.9(b) shows a representative scanning electron microscope (SEM) image of a fabricated sample with (i) pure Ni TFS, (ii) co-linear Ni-Cu TFS, and (iii) and (iv) two Ni-Cu TFS with patterned Cu rectangle perturbations, one of which is simply phase shifted relative to the other. Note that the Cu volume for (ii) is the same as for (iii) and (iv), since the Cu in (ii) is the full length of the Ni strip, but half the width; whereas the Cu in (iii) and (iv) is the full width of the Ni but half of the total length. The differences in the results between (i)-(iv) will allow us to analyze the influence of the Marangoni effect on the instability development. Conveniently, there is adequate contrast in the secondary electron coefficients of Cu and Ni as the Cu appears darker and the Ni brighter in the as-synthesized patterned strips as evidenced in the secondary electron image.

The spacing of the top Cu rectangle perturbations corresponds to the fastest growing wavelength predicted by the (linear) RP stability analysis and the extent of the perturbation is half of the wavelength. According to the RP analysis [86, 87], growth rate, β, of the perturbations of a free jet is determined by

$$\beta^2 = \frac{\sigma}{\rho R^3}\left[kR\left(1 - k^2R^2\right)\frac{I_1(kR)}{I_0(kR)}\right] \tag{5.3}$$

where R is the radius of the jet, and I_0 and I_1 are the modified Bessel functions. Hence the stability of a jet depends on its radius, R, such that the modes k for which $kR < 1$ are unstable and the modes for which $kR > 1$ are stable. Here, k is the wavenumber related to the perturbation wavelength, λ, by $k = 2\pi/\lambda$. The fastest growing mode corresponds to $k_m R \approx 0.7$. In the context of a TFS on the substrate, R corresponds to the radius of the rivulet of the same cross-sectional area as a TFS of thickness h, width w, and the equilibrium wetting angle θ, i.e.,

$$R = \sqrt{\frac{hw}{\theta - \cos\theta\sin\theta}}. \tag{5.4}$$

Figure 5.10 shows the growth rate of the perturbations of a free standing jet given by Equation (5.3). In the experimental setup, with the equilibrium wetting angle of Ni in contact with SiO_2 of $\theta = 69^\circ$, the radius of the rivulet is $R = 51\,\mathrm{nm}$ and the fastest growing unstable wavelength is $\lambda_m = 455\,\mathrm{nm}$.

A krypton fluoride (KrF) excimer laser, with wavelength $\lambda_{KrF} = 248\,\mathrm{nm}$, fluence energy $E_0 = 250\,\mathrm{mJ\ cm}^{-2}$, and pulse-width of $t_p = 18 \pm 2\,\mathrm{ns}$, is used for pulsed laser induced dewetting (PLiD). The area of the laser spot is $\sim 1\,\mathrm{cm}^2$, which is much larger than the $\sim 1\,\mathrm{mm}^2$ area containing the patterned samples, thus ensuring a spatially homogenous fluence on the sample. The liquid lifetime of the TFS is dependent on the energy fluence delivered by a laser pulse and the thermal and optical properties of the Ni, the supporting SiO_2 layer, and the underlying Si substrate. Due to the finite liquid lifetime per laser pulse ($\sim 20\,\mathrm{ns}$) and rapid cooling, a series of individual laser pulses can be used to control the instability evolution and interrogate morphological changes of the liquid filament [57].

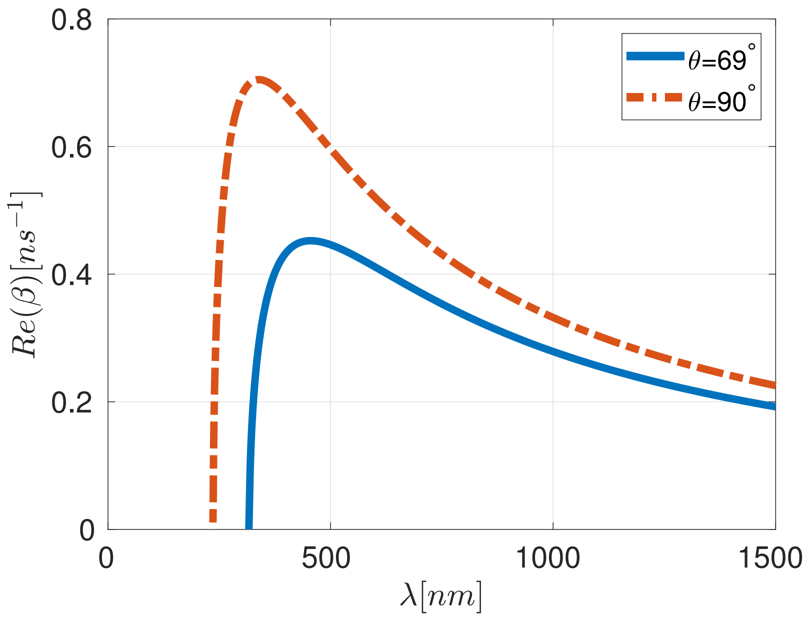


Figure 5.10 Growth rate of the perturbations of a free standing jet given in Equation (5.3), for $h = 12\,\text{nm}$, $w = 185\,\text{nm}$, and contact angles of $\theta = 69°$ and $\theta = 90°$ used in the experiments and simulations, respectively.

5.2.2 Computational Methods

The liquid metal filament is modeled as an isothermal, incompressible Newtonian fluid using the Navier-Stokes equations, given in Equation (2.1). The material concentrations are advected with the flow according to

$$\partial_t C_i + (\mathbf{u} \cdot \nabla) C_i = 0, \tag{5.5}$$

where C_i is the material concentration and i = Cu, Ni. Note that we have ignored the diffusion here (compared to Equation (2.13)). An extrapolation of the Cu self-diffusion (migration of the Cu atoms with the bulk) coefficient α at the Ni melting temperature results in a diffusion coefficient of $\alpha \sim 6 \times 10^{-9}\text{m}^2\text{s}^{-1}$ [88]. Thus an estimate of the diffusion distance ($\sqrt{4\alpha t}$, where t is the liquid lifetime, and α is assumed to be constant at the Ni melting temperature yields $\sqrt{4\alpha t} \sim 22\,\text{nm}$. The results of simulations show that advection leads to complete coverage by Cu, therefore Cu covers the distance of $\sim 100\,\text{nm}$ in approximately 1/3 of the liquid lifetime). We conclude that advection effects are dominant, justifying ignoring diffusive effects in the simulations.

The concentrations are initialized according to the experimental geometry shown in Figure 5.9(a) (iii) and (iv), where we simulate 4 perturbation wavelengths and one half of the filament along the long axis of symmetry. In order to model the fluctuations existing in fabrication of the experimental initial geometry we perturb the initial height of the filament by random noise of the form $h_r = \sum_i^N a_i \cos(i\pi x/4\lambda_m)$, with a constraint that $|h_r| < 1\,\mathrm{nm}$, where a_i and are random amplitudes taken from a uniform distribution, and $N = 125$ gives the smallest wavelength that can be resolved with at least 8 computational cells. We present the results for four realizations of random noise. In these simulations, due to the limitations of the present simulation setup, we use a contact angle of 90° resulting in $\lambda_m = 340\,\mathrm{nm}$ (see Figure 5.10). The boundary condition for the in-plane velocities on the substrate is the Navier-slip boundary condition $(u, w)|_{y=0} = \Lambda\partial_y (u, w)|_{y=0}$, where (u, w) are the in-plane components of the velocity, and $\Lambda = 20\,\mathrm{nm}$ denotes the slip length; note, y points out of the plane of the substrate, and x along the filament.

For computational efficiency and to minimize the effect of the surrounding vapor (i.e., air), we set $\mu_v = \mu_l/20$ and $\rho_v = \rho_l/20$; the factor of 20 is sufficiently large to ensure that its exact value is not important. To confirm that this is the case, we have carried out additional simulations with the factor of 100 and found that the only effect of this larger value is slightly faster $(1 - 2\,\mathrm{ns})$ breakup. Furthermore, we assume the values of μ_l and ρ_l are those of Ni at the melting temperature for both the Cu and Ni regions. Hence, the material parameters are: $\rho = 7810\,\mathrm{kg\ m^{-3}}$ and $\mu = 4.9\,\mathrm{mPa\ s}$. The spatial discretization is accomplished using an adaptive mesh, with the resolution at the fluid-vapor interface of being approximately 1.5 nm.

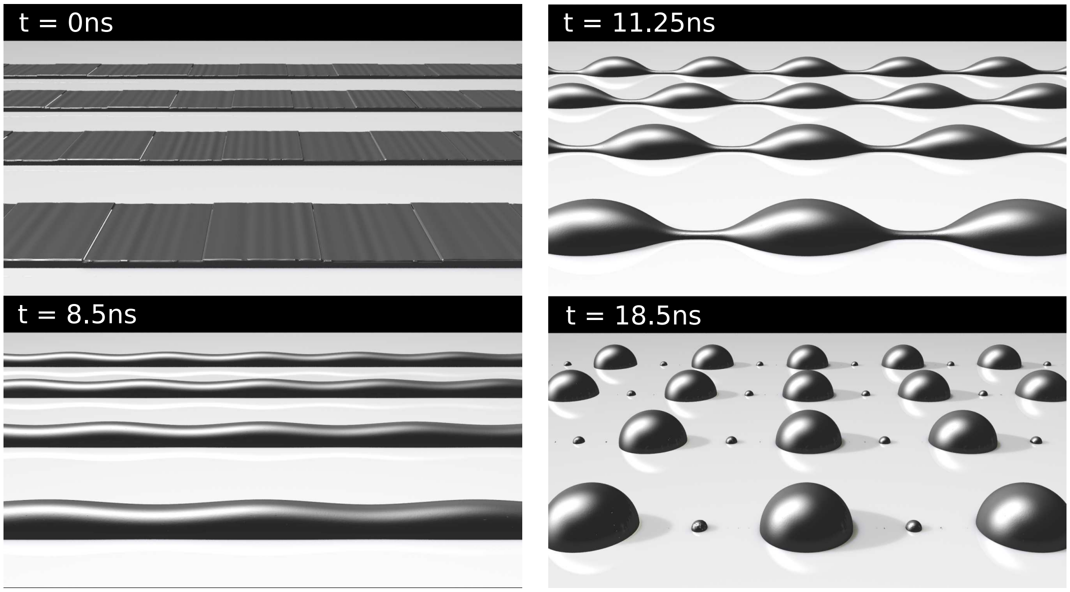


Figure 5.11 Simulation snapshots of the evolution of four realizations of the random noise on top of Ni-Ni TFS geometry where the spacing of the rectangular Ni perturbations corresponds to λ_m.

5.2.3 Results

We first illustrate the influence of the rectangular perturbations on top of Ni TFS on the instability evolution and final nanoparticle placement. The simulated geometry is the same as outlined above, however, we assume the perturbations are of Ni material, so that there is no difference in the surface tension coefficient between the TFS and the perturbations. Figure 5.11 shows the evolution of four filaments with different realizations of the random noise. The following purely geometric effects are observed: (1) following the melting by the laser, patterned 2 nm regions generate peaks in the height profile and troughs in the unpatterned regions, producing a filament height perturbation; (2) as the filament evolves, the troughs propagate toward the substrate since the wavelengths are unstable with respect to RP type of instability; (3) eventually the filament breaks up and the resultant nanoparticles are positioned at the locations where the original 2 nm of additional Ni material was patterned.

With this context, we demonstrate that patterned Ni-Cu setups evolve quite differently due to the Marangoni effect. The surface tension coefficient of Cu ($\sigma_{Cu} =$

$1.28\,\mathrm{N\ m^{-1}}$) at the Ni melting temperature ($T_M = 1728\,\mathrm{K}$) is lower than that of Ni ($\sigma_{Ni} = 1.78\mathrm{N\ m^{-1}}$). Thus, during PLiD, if the Marangoni effect is operative and dominant it should induce material flow from the thicker patterned Cu region to the adjacent Ni region and ultimately generate a surface perturbation where the resultant nanoparticles locate in the initially thinner Ni-only regions.

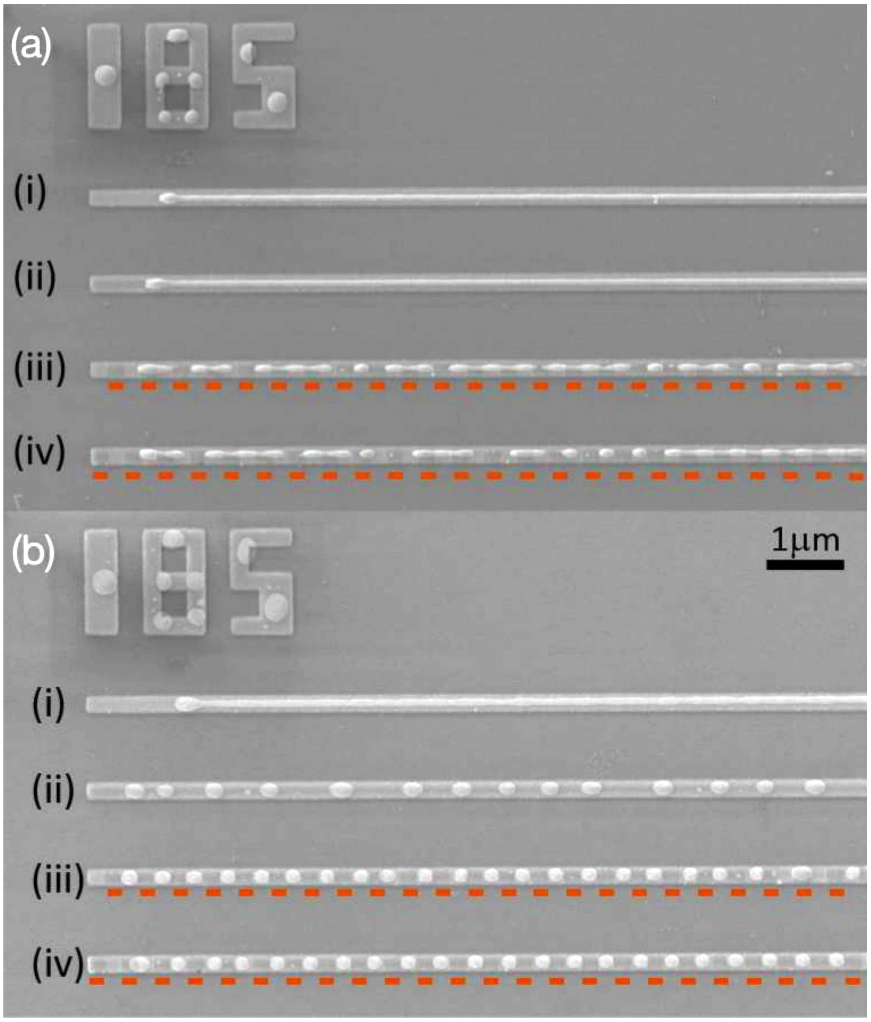


Figure 5.12 Experimental PLiD results for the Ni-Cu TFS geometry after (a) one and (b) five lasers pulses; (i) pure Ni TFS, (ii) co-linear Ni-Cu TFS, and (iii)-(iv) Ni-Cu TFS geometries with patterned Cu rectangle perturbations with spacing corresponding to λ_m, and one pattern is phase shifted relative to the other.

Figure 5.12 shows the experimental results for the Ni-Cu geometry patterned at the maximum instability wavelength; (a) shows the overlay of the original and after a single laser pulse and (b) is an overlay of the original and after 5 laser pulses. In the single laser pulse image, the instability development for the pure Ni thin film strip (i) and the co-linear Ni-Cu strip (ii) is much slower compared to the pattered rectangular Cu perturbations geometries in (iii) and (iv), where the breakup already occurred. The five laser pulse image shows little perturbation growth in (i), compared to the full development and breakup of the filaments in (ii), (iii) and (iv). The nanoparticle formation occurs with random positions in (ii), and with very good location fidelity corresponding to the original Ni-only regions in (iii) and (iv).

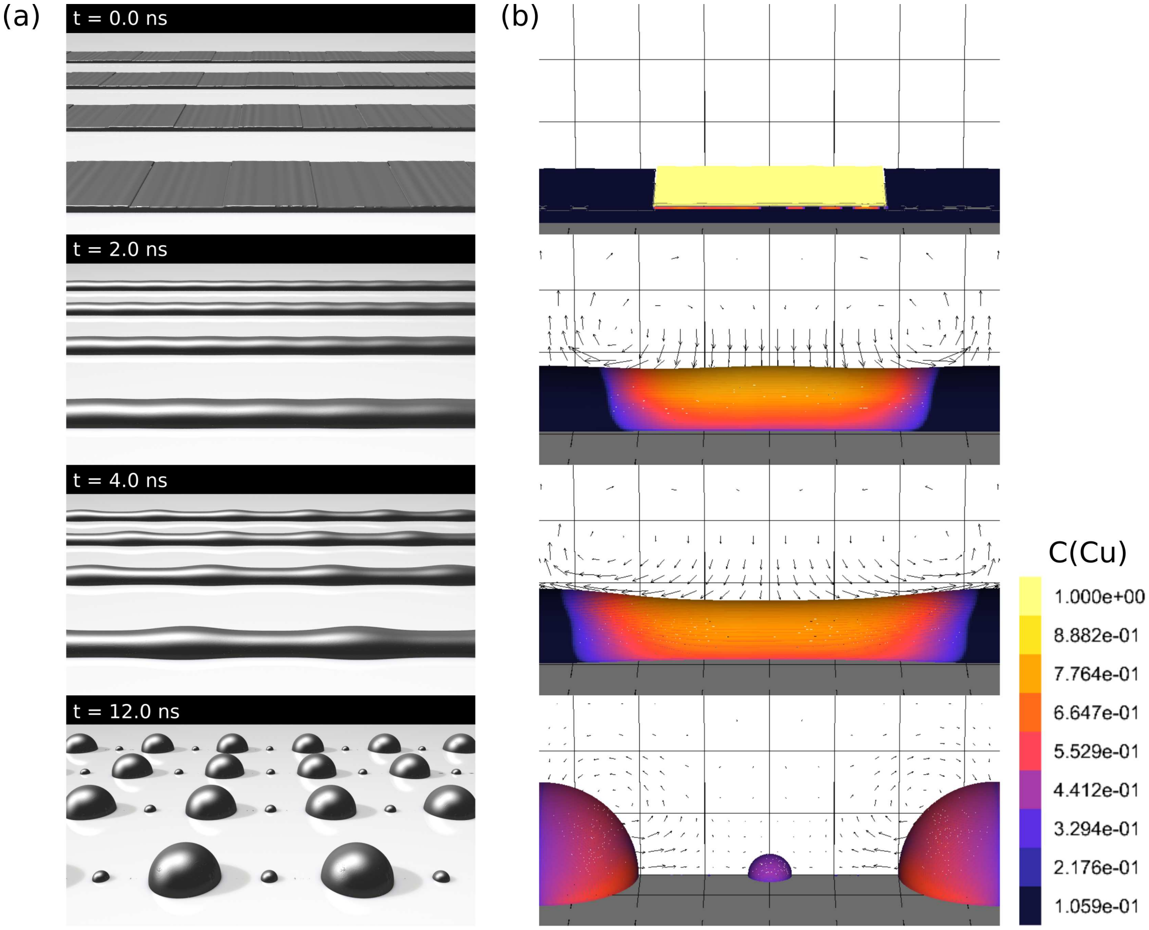


Figure 5.13 Simulation snapshots of the evolution of the Ni-Cu TFS geometry where the spacing of the rectangular Cu perturbations corresponds to λ_m; (a) four independent realizations of the random noise; (b) close up of (a) over one perturbation wavelength showing a half of the filament width, with the color representing the concentration of Cu at the interface, and the vector field representing the velocity field at the cross section along the axis of the symmetry of the filament.

Figure 5.13 shows the evolution of the Ni-Cu TFS. At $t = 2\,\mathrm{ns}$ we can see from the velocity field that the Marangoni effect drives the fluid away from the Cu regions towards the adjacent Ni areas. At $t = 4\,\mathrm{ns}$ the perturbation locations are inverting, and the peaks are shifting towards the Ni regions. As the filament continues to evolve with time, the RP-type instability eventually breaks the filament into nanoparticles located at the initially lower Ni regions. In agreement with the experimental results in Figure 5.12, the nanoparticle positions are at the locations where dry spots would be expected if there was no surface tension difference between the materials (as in Figure 5.11). We refer to this phenomenon as the "breakup inversion".

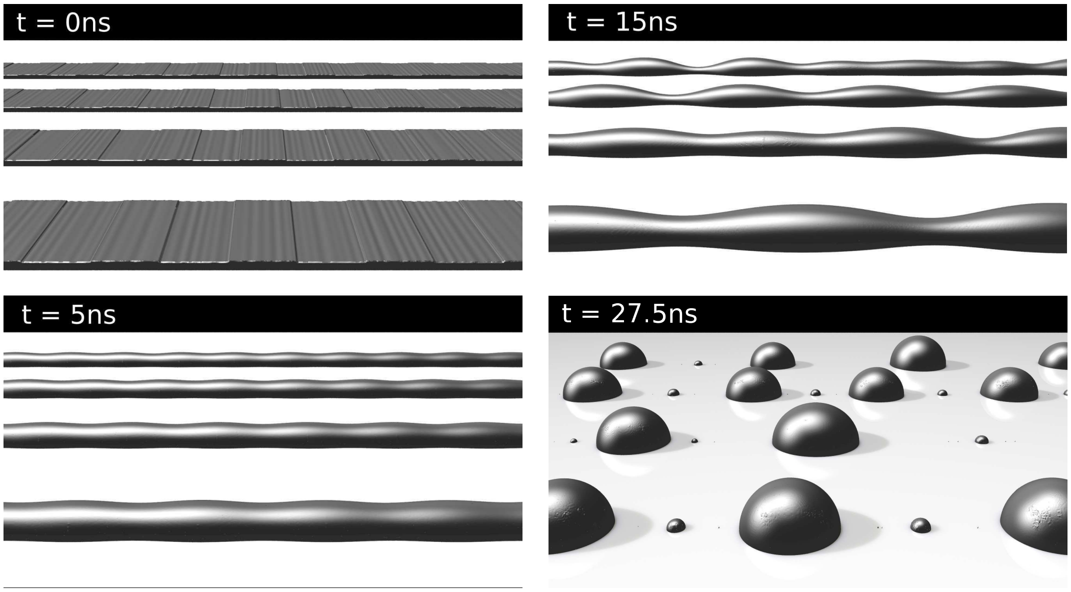


Figure 5.14 Simulation snapshots of the evolution of four realizations of the random noise on top of Ni-Cu TFS geometry where the spacing of the rectangular Cu perturbations corresponds to an unstable wavelength $\lambda = 182\,\mathrm{nm}$.

Next, we demonstrate the importance of the choice of the perturbation wavelength to achieve the breakup inversion. Figure 5.14 shows the evolution of a filament with the Cu perturbation spacing of $\lambda = 182\,\mathrm{nm}$, which corresponds to $kR = 1.1$, i.e., a stable mode based on Equation (5.3). At $t = 5\,\mathrm{ns}$ we can see that the perturbation locations are shifting, however, since the perturbation wavelength is stable, the filaments do not immediately break into nanoparticles at those locations.

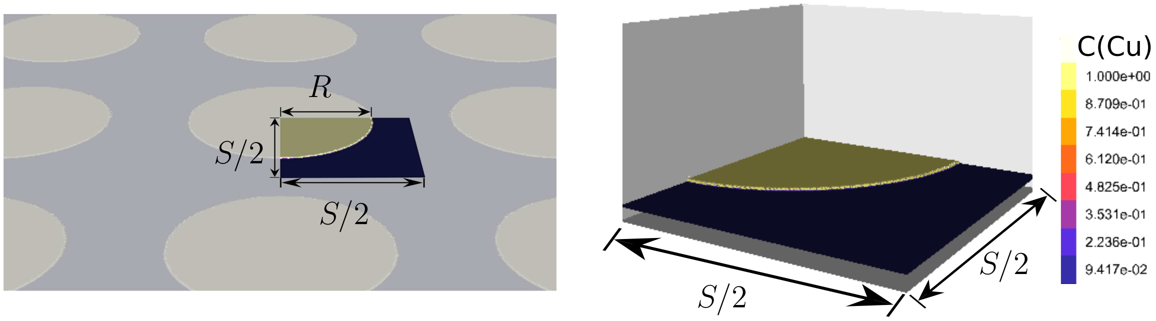
At $t = 15\,\text{ns}$, a perturbation with a larger wavelength than the initially patterned one is developed, eventually breaking the filament into nanoparticles with two primary nanoparticles per computational domain, corresponding to a spacing of $\sim 364\,\text{nm}$. Therefore, perturbing TFS by the wavelengths that are stable with respect to the RP stability analysis does not lead to the formation of particle assemblies conforming to the wavelength of imposed perturbations.

5.2.4 Conclusions

Until now, the directed assembly of metallic nanostructures using PLiD has focused on imparting instabilities by lithographically patterning geometric perturbations to direct the ordering of resultant nanoparticles. Here we investigate a new approach to imparting instabilities by templating the regions with different surface tension by selectively patterning the surface of a Ni thin films strips with Cu where the patterning length scales were selected as the wavelengths of maximum growth rate predicted by the Rayleigh-Plateau stability analysis. Experimental results are confirmed and rationalized via fully non-linear 3D simulations using a VoF method that includes the spatially varying surface energy. The patterned Cu regions have a lower surface energy than the Ni regions and induce Marangoni flow. Future work will explore applications of a similar approach to directed assembly for other relevant geometries.

5.3 Breakup of Flat Metal Films with Cylindrical Perturbations

In this section, we explore a concentration induced Marangoni flow for a different geometry compared to Section 5.2: we consider a flat Ni film of thickness $h_0 = 12\,\text{nm}$ with cylindrical Cu perturbations arranged in a square pattern with spacing S, height $\Delta h = 2\,\text{nm}$ and radius $R_{cyl} = S/4$. Figure 5.15(a) shows the initial setup of the cylinder arrangements. The symmetry in the cylinder arrangement allows us to reduce the computational domain to a square domain containing one quarter of the



(a)

(b)

Figure 5.15 (a) The initial setup of a flat film with cylindrical perturbations arranged in a square pattern; (b) the computational domain. The color represents the concentration of Cu at the interface.

cylinder and extending to $S/2$ towards centers of the neighboring cylinders, as shown in the highlighted shape in Figure 5.15(a). For clarity, Figure 5.15(b) shows the computational domain, where the boundary conditions at the horizontal boundaries are the symmetry boundary condition. The spacing between the cylinders, S, is chosen based on the LSA of a perturbed 3D film. In this section, we show the results for $S = \lambda_m$, the wavelength of the maximum growth, and $S \gtrsim \lambda_c$, an unstable wavelength close to the critical wavelength. We compare the simulations with Cu cylindrical perturbations – including the concentration dependent surface tension coefficient – with the simulations with Ni cylinders where the surface tension is constant.

The destabilizing mechanism leading to the breakup of the nanoscale films is modeled by the fluid-structure interaction in the form of a disjoining pressure [89]. The disjoining pressure can be included in the Navier-Stokes equations (Equation (2.1)) as a body force as follows

$$\rho(\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}) = -\nabla p + \nabla \cdot (2\mu D) + \sigma\left(\mathbf{x}\right) \kappa \delta_s \hat{\mathbf{n}} + \nabla_s \sigma\left(\mathbf{x}\right) \delta_s + \mathbf{F_{vdw}}\left(y\right) \delta_s, \quad (5.6)$$

where

$$\mathbf{F}_{\mathbf{vdw}}\left(y\right)=K_{\pi}\left[\left(\frac{h^{*}}{y}\right)^{m}-\left(\frac{h^{*}}{y}\right)^{n}\right]\hat{\mathbf{n}}, \tag{5.7}$$

$$K_{\pi}=\frac{\sigma_{0}(1-\cos\theta_{eq})}{Mh_{*}}, \tag{5.8}$$

$$M=\frac{n-m}{(m-1)(n-1)}, \tag{5.9}$$

where σ_0 is the surface tension coefficient of Ni, $\theta_{eq} = 70°$ is the prescribed equilibrium contact angle, $m = 3$ and $n = 2$ are the exponents in the Lennard-Jones type potential, $h_* \approx 2\,\text{nm}$ is the thickness of the equilibrium film. The details of the implementation of the disjoining pressure in the VoF method can be found in [90].

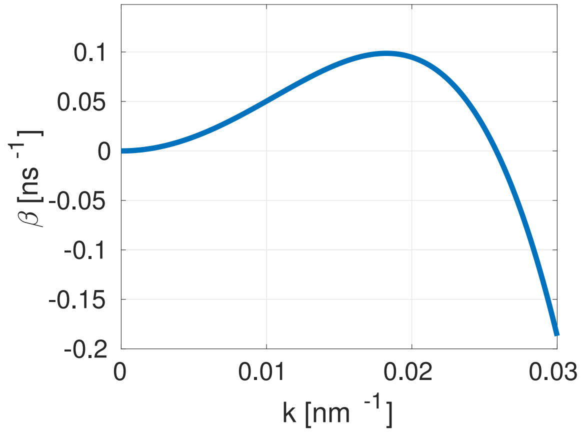


Figure 5.16 The growth rate of a perturbed film given in Equation (5.14).

As indicated above, the spacing between the cylinders, S, is chosen based on the linear stability of the 3D perturbed film. Starting from the long wave approximation

$$2\mu h_t+\sigma\nabla\cdot\left(h^{3}\nabla\nabla^{2}h\right)+\nabla\cdot\left[h^{3}\nabla\Pi\left(h\right)\right]=0, \tag{5.10}$$

$$\Pi\left(h\right)=\frac{\sigma\left(1-\cos\theta\right)}{Mh_{*}}\left[\left(\frac{h_{*}}{h_{0}}\right)^{n}-\left(\frac{h_{*}}{h_{0}}\right)^{m}\right] \tag{5.11}$$

$$M=\frac{n-m}{\left(n-1\right)\left(m-1\right)}, \tag{5.12}$$

we assume the film is perturbed around an equilibrium film thickness h_0 as $h\left(x,z,t\right) = h_0 + h_0\varepsilon e^{i(kx+lz)+\beta t}$, where ε is a small parameter, k and l are the wave numbers of the perturbations in the x and z directions respectively, and β is the growth rate

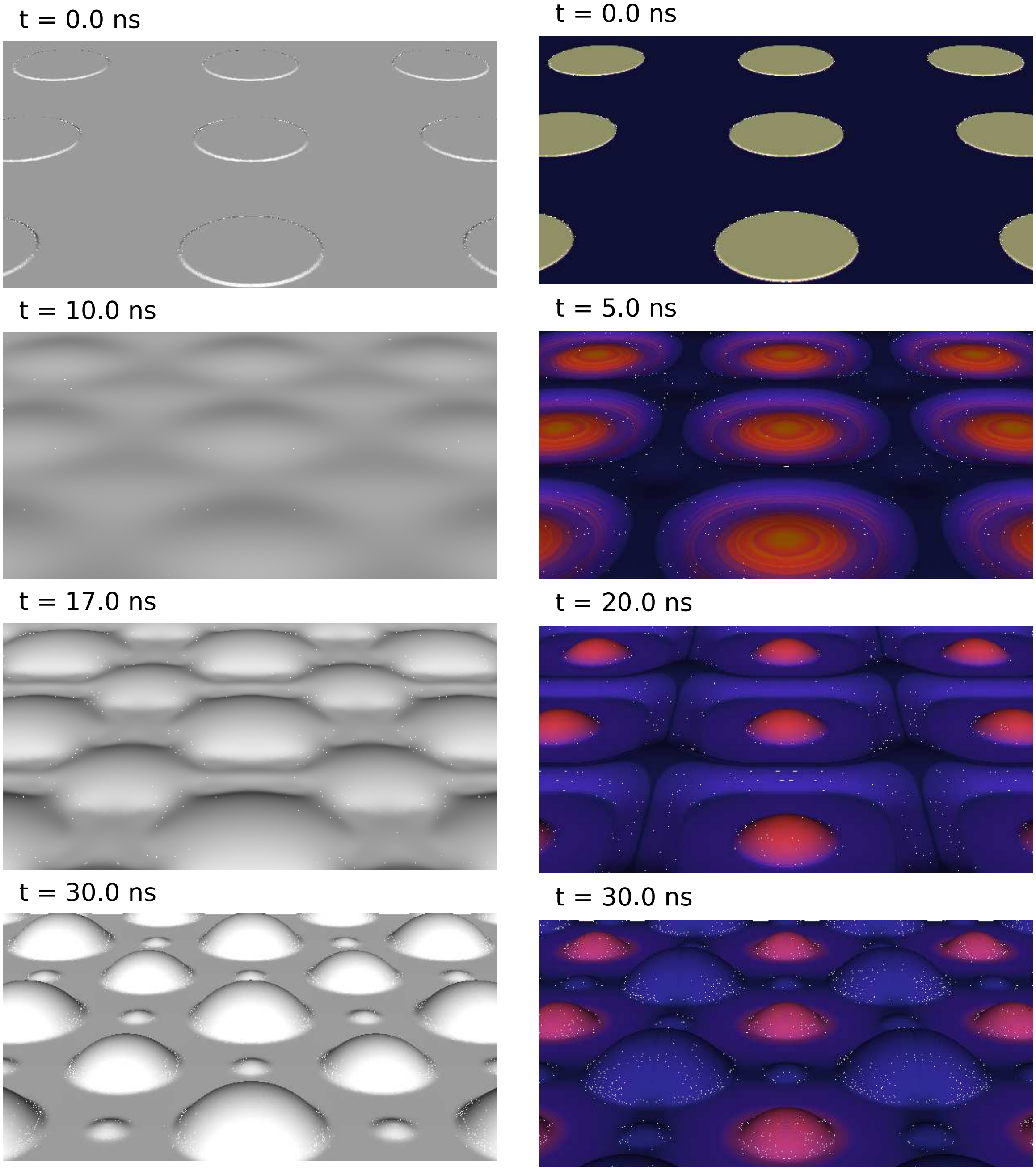
$$\beta = -\frac{1}{3\mu}\left(k^2+l^2\right)\left\{\frac{\sigma\left(1-\cos\theta\right)}{Mh_*h_0}\left[n\left(\frac{h_*}{h_0}\right)^n - m\left(\frac{h_*}{h_0}\right)^m\right] - \sigma\left(k^2+l^2\right)\right\}. \qquad (5.13)$$

To simplify the analysis, assume the perturbations in the x and z directions, have the same modes, i.e., $l = k$, then we have

$$\beta = -\frac{2k^2}{3\mu}\left\{\frac{\sigma\left(1-\cos\theta\right)}{Mh_*h_0}\left[n\left(\frac{h_*}{h_0}\right)^n - m\left(\frac{h_*}{h_0}\right)^m\right] - 2k^2\sigma\right\}. \qquad (5.14)$$

Figure 5.16 shows the growth rate β as a function of the wavenumber k. Note that to obtain the expression for the dispersion relation, Equation (5.14), we ignored the effects of the variable surface tension, which we included previously in Equation (3.3). The inclusion of the variable surface tension coefficient would significantly complicate the analysis here, since the surface tension gradients are initially only at the boundaries of the two materials, and as Cu spreads during the evolution the gradients change as well.

Figure 5.17 shows the evolution of the Ni film with cylindrical perturbations where the spacing between cylinder centers is $S = \lambda_m$. There is a significant difference in the evolution dynamics and the final droplet configuration. In the simulation with Ni cylinders, at $t = 10\,\mathrm{ns}$ the cylinder perturbations are smoothed out, but secondary perturbations start forming at positions between the cylinders on diagonals of the squares of initial cylinder positions. At time $t = 17\,\mathrm{ns}$, the perturbations start to reach the precursor film thickness, and the primary and secondary droplets start forming at the cylinder and diagonal positions respectively. Additionally, filaments are formed connecting the secondary droplets. At time $t = 30\,\mathrm{ns}$ the film has reached the steady state, and tertiary droplets have formed from the filaments between the secondary droplets. In the simulation with Cu cylinders, at time $t = 5.0\,\mathrm{ns}$ rings of lower film height start forming around the cylinders due to the flow created by the surface tension gradients between Ni and Cu material. As the rings grow in depth



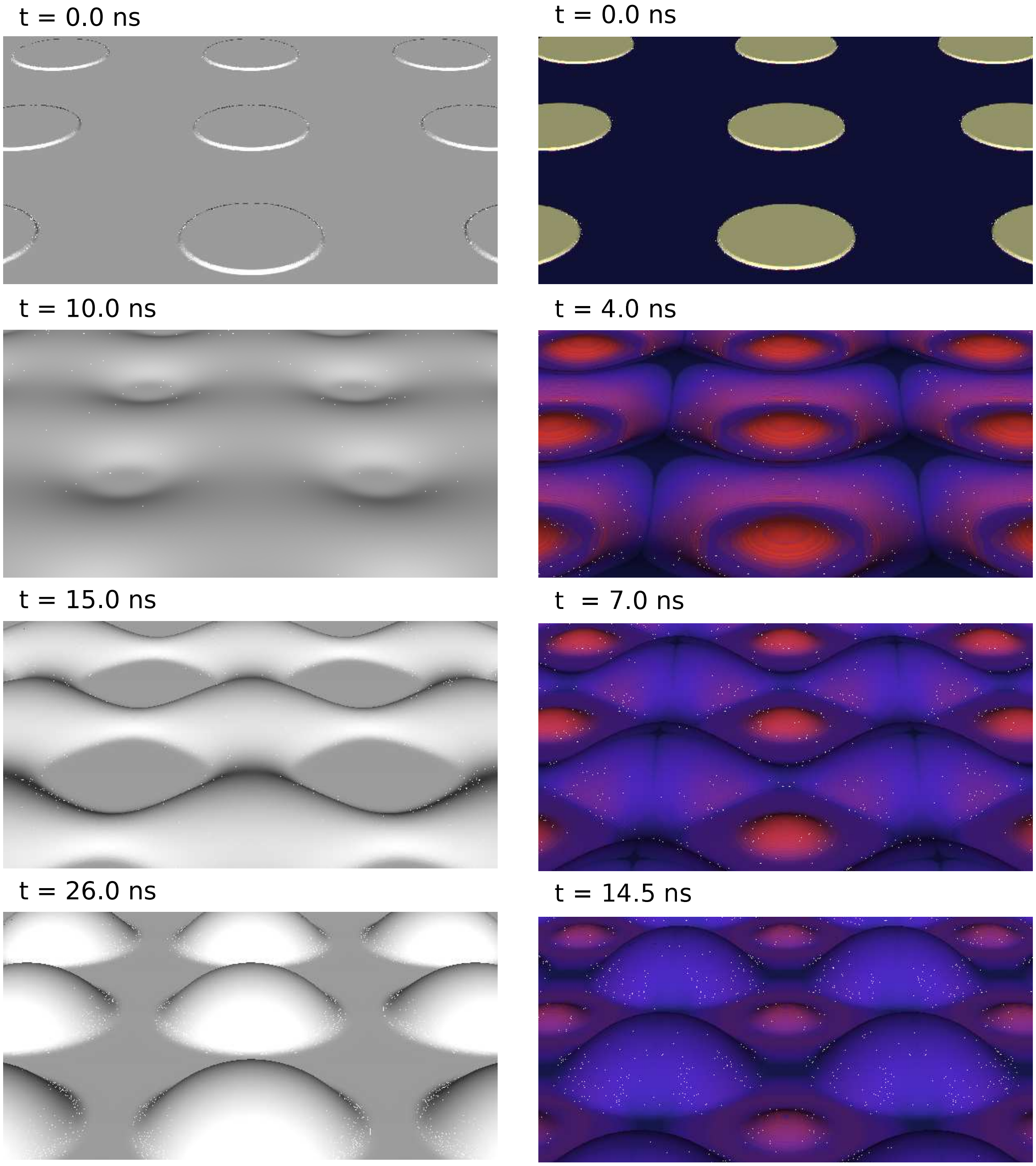
(a) (b)

Figure 5.17 The evolution of the Ni film with cylinder perturbations where the spacing between the cylinder centers is $S = \lambda_m$, and the perturbations are composed of (a) Ni and (b) Cu.

and reach the precursor thickness, they spread and, at time $t = 20.0\,\mathrm{ns}$, a network of rivulets forms intersecting in a square arrangement around the original cylinder positions from the material that was pushed by the growing rings. Each ring also leaves a droplet in the middle. At time $t = 30.0\,\mathrm{ns}$ the rivulets break into droplets with largest drops at the rivulet intersections, and smaller drops forming from the rivulets between the intersections.

Note that the final configuration of larger vs. smaller droplets in Figure 5.17 is reversed in the simulations with Ni and Cu cylinders. In the simulation with Ni cylinders, the perturbations are enhanced at the cylinder positions, hence the final droplets are largest at those positions. On the other hand, in the simulation with Cu cylinders, the Marangoni flow around the cylinders moves the material away from the cylinders in the early stage of the evolution, which eventually causes the droplets at the cylinder positions to be smaller than the droplets at the diagonals.

Next we attempt to make the secondary droplets at the cylinder locations smaller by allowing more time for the Marangoni flow to push the material away from the cylinder positions. We pick the spacing between the cylinders corresponding to an unstable wavelength close to the critical wavelength λ_c. Hence, the growth rate of the perturbations is slower compared to the growth rate at λ_m. Figure 5.18 shows the evolution of the Ni film with spacing between the centers of the cylinders $S \gtrsim \lambda_c$. The development of the instabilities and the breakup dynamics are similar to the case of $S = \lambda_m$. However, due to a smaller spacing between the initial cylinders, the final droplet configuration is different. In the case of Ni cylinders the perturbation peaks first start developing at the positions of the original cylinders. However, unlike in the case $S = \lambda_m$, there are no secondary peaks at the diagonals between the cylinders. In fact, at $t = 10\,\mathrm{ns}$ the valleys of the perturbations first reach the precursor film thickness at the diagonals between the cylinders. As a result, there are no secondary droplets. The tertiary droplets do not develop either. The primary droplets are larger



(a) (b)

Figure 5.18 The evolution of the Ni film with cylinder perturbations where the spacing between the cylinder centers is $S \gtrsim \lambda_c$, and the perturbations are composed of (a) Ni and (b) Cu.

compared to the case $S = \lambda_m$, and there is little space left between the primary drops. In the case of Cu cylinders, first at time $t = 4.0\,\mathrm{ns}$ rings of film with lower height start forming around the cylinders due to the Marangoni flow pushing the material away from the Cu cylinders. Similar to the case of the $S = \lambda_m$, a square network of rivulets forms as the rings reach precursor thickness, however, they break into larger droplets leaving no material or space for tertiary droplets to form.

5.3.1 Conclusions

In this section, we demonstrated the possibility of creating inverted droplet configurations, by exploiting the difference in the surface tension coefficient between the Cu and Ni material, for a second geometry in addition to the perturbed filaments presented in Section 5.2. The presented study is in no way extensive: further analysis can be done to study different arrangements of the cylinder perturbations, different radii of cylinders compared to the cylinder spacing, different film and cylinder heights.

5.4 Thermocapillary Breakup of Liquid Metals

The variations in the temperature of the liquid metal films melted by a pulsed laser, can be caused by the variations in the film height. More specifically, the variations in the temperature are caused by the dependence of the optical properties of the metal on the film height [69]. The variations in the temperature can lead to dynamics different from those occurring in an isothermal setting. In this section we explore the influence of the temperature dependent surface tension coefficient and viscosity on the breakup of liquid metal films and filaments.

The variations in the surface tension due to temperature in the liquid metal films can be spatial, due to the variations in the absorption of the laser energy, and temporal, due to the time dependence of the laser pulse fluence. In this section we study the influence of the spatial and temporal temperature variations on the breakup of 2D thin metal film and on the breakup of 3D filaments. We outline two temperature solutions used in the study of film breakup: first in Section 5.4.1, we present a reduced model that ignores in-plane heat conduction, as well as temporal evolution of the film or filament (referred to as the "reduced model" from now on) [58]; and second in Section 5.4.3, we present the numerical 2D temperature solution computed using Gerris (referred to as the "complete model"). In Section 5.4.2, we outline an analytical solution for a flat film, which we use for validating and comparing the two models above.

Next, we study the influence of the temperature on the stability of 2D thin films. First, in Section 5.4.4, using the linear stability analysis (LSA), we find that the spatial temperature variations in the film can have a stabilizing or destabilizing effect depending on the film height: for films of height less than a critical value h_c, the temperature variations have a stabilizing effect, and for the films of heights above h_c, the temperature variations have a destabilizing effect. Second, in Section 5.4.5, using the direct numerical simulations, we compare the influence of the temperature variations on the film breakup using reduced and complete temperature models. We find that in the reduced model the thermocapillary force is exaggerated due to the lack of in-plane heat conduction. Furthermore, we find that compared to the spatial variations, the temporal temperature variations have a stronger influence on the film stability.

Finally, in Section 5.4.7, we consider the influence of the thermal variations on the breakup on the liquid metal filaments. We find that the thermocapillary flow caused by these variations is weak compared to the evolution governed by the Rayleigh-Plateau (RP) instability.

5.4.1 Reduced Model for the Temperature of a Thin Film

In this section, we outline the reduced model for the metal temperature; a version of such model is discussed in Trice et al. [58]. The film–substrate bilayer is assumed to be infinitely wide in the in-plane directions, so that the film height is small relative to its width. The substrate layer is assumed to be thick compared to the film height and it is modeled as a semi-infinite medium $0 \leq y < -\infty$. The small ratio of the out-of-plane and in-plane scales in the metal film implies that the heat conduction in the in–plane direction in the film is negligible compared to the conduction in the out-of-plane direction. Hence, within this reduced model, the heat conduction in the bilayer can be described by the one-dimensional heat equation in each layer,

$$(\rho C_p)_m \frac{\partial T_m}{\partial t} = k_m \frac{\partial^2 T_m}{\partial y^2} + S(y,t) \quad \text{in the fluid,} \tag{5.15}$$

$$(\rho C_p)_s \frac{\partial T_s}{\partial t} = k_s \frac{\partial^2 T_s}{\partial y^2} \qquad \text{in the substrate,} \tag{5.16}$$

where C_p is the effective heat capacity and k is the thermal conductivity. The subscripts s and m correspond to substrate and metal properties, respectively. The source term can be written as

$$S(y,t) = \frac{E_0 f\left(t\right)}{\sqrt{2\pi}\sigma_{tp}} \left[1 - r_0\left(1 - e^{-a_r h}\right)\right] e^{-\alpha_m (y+h)}, \tag{5.17}$$

where the first factor represents the incident energy from the laser source, the second factor accounts for the reflectance of the metal film, and the last factor represents the energy absorbed by the film. More details regarding the derivation of the source term and the explanation of the parameters are given in Appendix A. In Equation (5.17), E_0 is the intensity of the incident radiation, σ_{tp} is the width of the Gaussian laser pulse, and $f(t)$ gives the temporal profile of the laser fluence, $f(t) = e^{-(t-t_p)^2/(2\sigma_{tp}^2)}$. The boundary conditions are as follows

$$
\begin{aligned}
\frac{\partial T_m}{\partial y} &= 0 && \text{at the fluid air interface } y = h(t,x), && (5.18)\\
k_m \frac{\partial T_m}{\partial y} &= k_s \frac{\partial T_s}{\partial y} && \text{at the fluid substrate boundary } y = 0, && (5.19)\\
T_m &= T_s && \text{at the fluid substrate boundary } y = 0, && (5.20)\\
T_s &\to T_0 && \text{as } y \to -\infty, && (5.21)
\end{aligned}
$$

where $y = h(x,t)$ corresponds to the air-film interface, and $y = 0$ is the film-substrate interface.

The spatial variations of the temperature in the metal film are expected to be small due to the small film height and high thermal conductivity of the metal. Hence, within this model, it can be assumed that the temperature is constant in the film, and the temperature of the liquid-air interface can be approximated by the average film temperature, $T_m^* = \frac{1}{h}\int_0^h T_m \, dy$. Integrating Equation (5.15) with respect to y, from $y = 0$ to $y = h$, and using the boundary conditions (5.18) and (5.19), gives the equation for averaged temperature of the film, T_m^*

$$
\begin{aligned}
\frac{\partial T_m^*}{\partial t} &= S^*(h,t) - \frac{1}{h}\frac{q_s(t)}{(\rho C_p)_m}, && (5.22)\\
S^*(h,t) &= \frac{E_0 f(t)}{\sqrt{2\pi}\sigma_{tp}(\rho C_p)_m}\left[1 - r_0\left(1 - e^{-\alpha_r h}\right)\right]\left[1 - e^{-\alpha_m h}\right]\frac{1}{h}, && (5.23)
\end{aligned}
$$

where $q_s(t) = \partial T_s/\partial y|_{y=0}$. The heat equation for the substrate (5.16) can be solved using Green's functions or Laplace transform. Using the boundary conditions (5.19), (5.20) and (5.21), the average temperature of the film is found to be

$$T_m^*(h,t) = T_0 + S^* \, e^{-\frac{t_p^2}{2\sigma^2}} \int_0^t \exp\left(-\frac{(t-u)^2}{2\sigma^2} + \frac{t_p}{\sigma^2}(t-u)\right) e^{K^2 u} \operatorname{erfc}(K\sqrt{u}) \, du, \quad (5.24)$$

where

$$K(h) = \frac{\sqrt{(\rho C_p k)_s}}{(\rho C_p)_m h}.$$

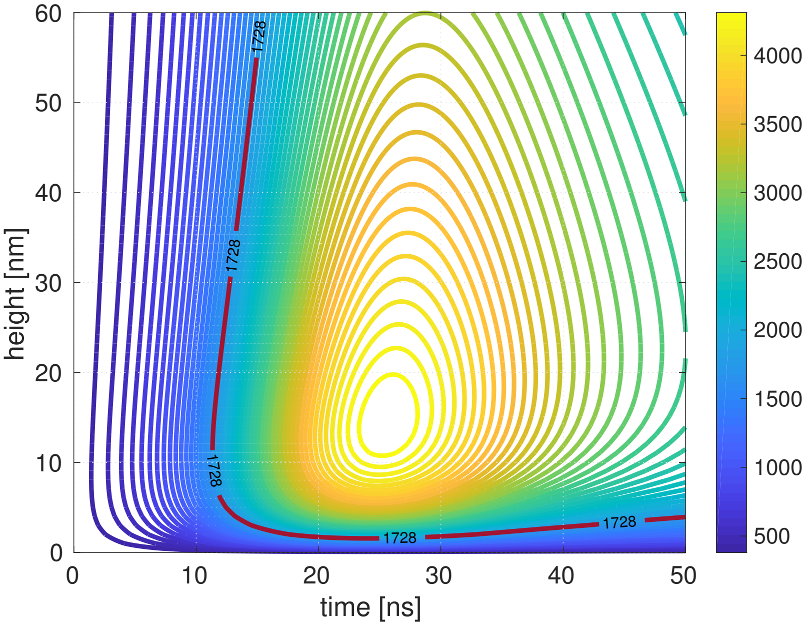


Figure 5.19 The average temperature of a metal film, T_m^* as a function of film height and time. The red highlighted curve represents the melting temperature of Nickel, $T_M = 1728\,K$.

Figure 5.19 shows the contour plot of the average temperature of the metal film, $T_m^*(h,t)$, as given by Equation (5.24). The red highlighted curve represents the melting temperature of Nickel ($T_M = 1728\,\mathrm{K}$). The dependence of the energy absorption on the film height leads to the non-monotonic dependence of the film temperature on the film height, h. For a small film height, $h < h_c \approx 14.3\,\mathrm{nm}$, only a part of the laser energy is absorbed, which leads to low film temperatures. For film heights $h > h_c$ the film absorbs most of the laser pulse energy. Hence, the film of height $h \approx h_c$ reaches the highest temperature, and for the film of heights $h > h_c$, the temperature decreases as h grows due to the larger amount of material that needs to be heated. Later in Sections 5.4.4 and 5.4.5, we study the dynamics of two different film heights such that one of them is $h_0 < h_c$ and one is $h_0 > h_c$.

A known temperature at the interface which is expressed as a function of the film height, h, and time, t only, is convenient for implementing the Marangoni force in the VoF solver. As described in Section 3.1, we can directly compute the surface gradients using Equation (3.1), where $\partial T/\partial h$ is computed as $\partial T_m^*/\partial h$ analytically from Equation (5.24) as follows

$$\begin{aligned}\frac{\partial T_m^*}{\partial h}(h,t) = \frac{\partial S^*}{\partial h}\, e^{-\frac{t_p^2}{2\sigma^2}} \int_0^t \exp\left(-\frac{(t-u)^2}{2\sigma^2} + \frac{t_p}{\sigma^2}(t-u)\right) e^{K^2 u}\, \mathrm{erfc}(K\sqrt{u})\, du + \\ 2S^*\, e^{-\frac{t_p^2}{2\sigma^2}} \int_0^t \frac{dK}{dh} \exp\left(-\frac{(t-u)^2}{2\sigma^2} + \frac{t_p}{\sigma^2}(t-u)\right) \left[K e^{K^2 u}\, \mathrm{erfc}(K\sqrt{u}) - \sqrt{\frac{u}{\pi}}\right] du\end{aligned} \tag{5.25}$$

where erfc(u) is the complementary error function and

$$\frac{dK}{dh} = -\frac{\sqrt{(\rho C_p k)_s}}{(\rho C_p)_m h^2},$$

and

$$\begin{aligned}\frac{\partial S^*}{\partial h} = \frac{E_0}{\sqrt{2\pi}\sigma_{tp}(\rho C_p)_m} \Bigg\{ \left[-r_0 a_r e^{-a_r h})\right] \left[1 - e^{-\alpha_m h}\right] \frac{1}{h} + \\ \left[1 - r_0\left(1 - e^{-a_r h}\right)\right] \left[\left[\alpha_m e^{-\alpha_m h}\right] \frac{1}{h} - \left[1 - e^{-\alpha_m h}\right] \frac{1}{h^2}\right] \Bigg\}.\end{aligned}$$

For small film heights, T_m^* and $\partial T_m^*/\partial h$ need to be carefully computed to ensure the integrals in Equations (5.24) and (5.25) converge, see Appendix B. When used in our simulations, both T_m^* and $\partial T_m^*/\partial h$ are evaluated for an array of t and h values before the start of the simulations. During the simulation, we use bilinear interpolation to find the temperature at each interfacial cell and each time step. This makes the computations significantly faster, since we do not need to use the numerical integration to compute the integrals in Equations (5.24) and (5.25) for each interfacial cell at each time step.

5.4.2 Analytical Solution of the Heat Equation in a Film–Substrate System

The system of Equations (5.15) and (5.16), including the spatial variations in the y-direction in both the metal and the substrate, can be solved analytically using separation or variables, following the technique given by Ozicsik [91]. The analytical solution presented here is used for the verification of the reduced model given in Equation (5.24) and the complete model presented in Section 5.4.3.

In contrast to the reduced model presented in Section 5.4.1, for computing the analytical solution, we assume that the substrate is of finite depth. However, when comparing the solutions from different models, we use substrate depth large enough that the temperature solution is converged with increasing substrate thickness, and the solution is equivalent to that for the semi-infinite substrate (see Figure C.1 in Appendix C). The modified boundary condition (5.21) is

$$T_s = T_0 \quad \text{at the bottom of the substrate } z = -b, \tag{5.26}$$

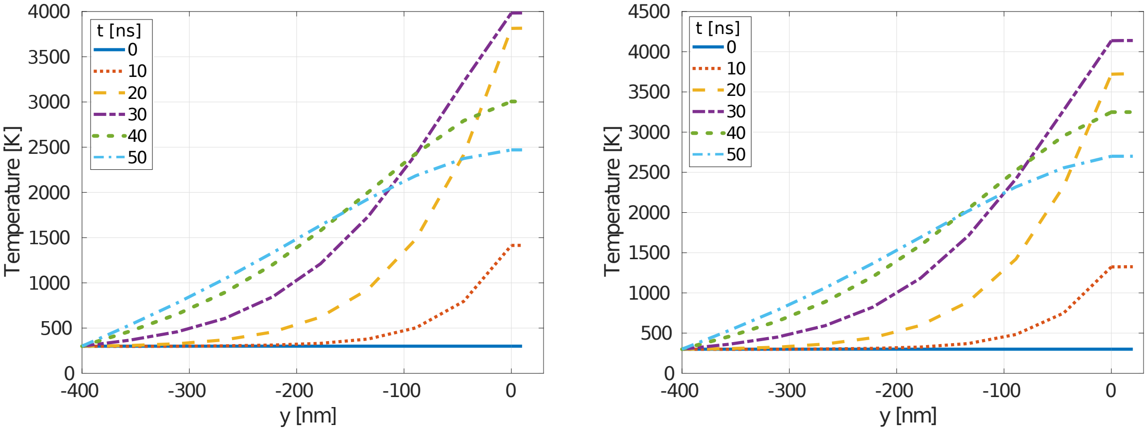
where b is the substrate thickness.

The solution can be found using separation of variables [91], and it can be written compactly in terms of Green's functions as

$$T_m(z,t) = \int_0^t \int_a^b G_{12}(z,t;\xi,\tau)\,\frac{\alpha_2}{k_2} S(\xi,\tau)\,\mathrm{d}\xi\mathrm{d}\tau \tag{5.27}$$

$$T_s(z,t) = \int_0^t \int_a^b G_{22}(z,t;\xi,\tau)\,\frac{\alpha_2}{k_2} S(\xi,\tau)\,\mathrm{d}\xi\mathrm{d}\tau \tag{5.28}$$

where G_{12} and G_{22} are given in Appendix C along with the details of the solution.



(a) (b)

Figure 5.20 The analytical solution for the temperature of the film ($y > 0$) and the substrate ($y < 0$). The film height is (a) $h_0 = 10\,\text{nm}$ (b) $h_0 = 20\,\text{nm}$.

Figure 5.20 shows the analytical temperature solution in the metal film ($y > 0$) and the substrate ($y < 0$). The temperature variation across the film height is small compared to the variation in the substrate. Therefore, ignoring temperature gradients across the film, as used in the reduced model, is justified. We note, however, that such a conclusion can be reached only for stationary flat films. As we will see later, using the reduced model for nonuniform films, or for the time dependent films, in general cannot be justified. On a different note, we point out that since there is no in-plane dependence in the source term, the 1D analytical solution presented here holds for a 2D or a 3D flat stationary film.

5.4.3 Numerical Model for the Temperature of a Film–Substrate System

Next we consider the outlined problem via direct numerical simulations, using the finite volume Navier–Stokes solver with VoF interface tracking (see Chapter 2). The presented numerical solution can be used for arbitrary metal-air interface shape. We solve the advection-diffusion equation in the liquid-air domain and the diffusion equation in the solid domain. The simulation setup has to address the following issue: using GERRIS flow solver we cannot solve for the temperature inside of the

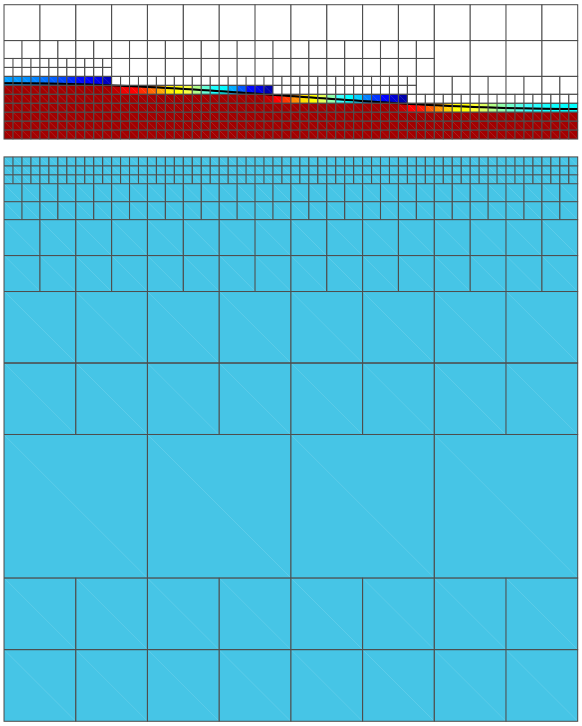


Figure 5.21 The fluid–substrate setup used in the direct numerical simulations.

solid substrate directly, since except on the boundaries, the implementation of the solid entities does not contain the computational cells. Hence, in order to solve for the temperature in the fluid and the substrate using Gerris we treat the solid domain – the substrate – as an immobile fluid. Furthermore, in order to impose the no-slip boundary condition on the metal-substrate boundary, we separate the two phases by a solid plate, which effectively creates a domain decomposition setup, i.e., the computational domain is separated into two disconnected subdomains, see Figure 5.21. Later in this section we show that this setup, referred to as the "complete model", is valid. This is done by direct comparison with the analytical solution.

We denote the top subdomain containing metal and air by Ω_f, and the bottom subdomain containing the substrate by Ω_s. In general, the temperature in Ω_f, denoted T_f, satisfies the advection-diffusion equation, and the temperature in Ω_s, T_s, satisfies the diffusion equation

$$\rho\left(\chi\right)C_p\left(\chi\right)\left[\partial_t T_f+(\,\mathbf{u}\cdot\nabla\,)T_f\right]=\nabla\cdot\left(k\left(\chi\right)\nabla T_f\right)+S_n(\mathbf{x},t)\quad\text{in }\Omega_f \tag{5.29}$$

$$\left(\rho C_p\right)_s\partial_t T_s=k_s\nabla^2 T_s\quad\text{in }\Omega_s \tag{5.30}$$

where $\rho\left(\chi\right)$, $C_p\left(\chi\right)$ and $k\left(\chi\right)$ are the phase dependent density, heat capacity and the conductivity of the metal and air, defined as the volume fraction weighted average of the metal and air properties (see Chapter 2). The boundary conditions for the temperature at the bottom boundary of Ω_f and at the top boundary of Ω_s, couple the temperatures of the two subdomains.

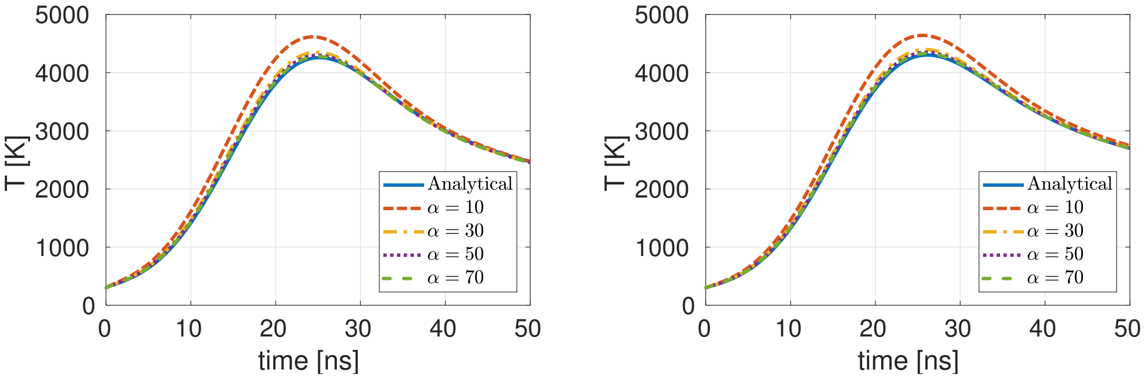
The coupling of the temperature solution between the liquid and solid domains is accomplished using Newton's law of cooling, which we impose on the top and bottom of the solid plate as follows

$$k_m\frac{\partial T_f}{\partial y}=\alpha\left(T_f-T_s\right)\quad\text{at the top of the solid,} \tag{5.31}$$

$$k_s\frac{\partial T_s}{\partial y}=\alpha\left(T_f-T_s\right)\quad\text{at the bottom of the solid,} \tag{5.32}$$

where α is the heat transfer coefficient. Note that the right hand sides of Equations (5.31) and (5.32) are equal, implying the continuity of the flux between the liquid and substrate. Furthermore, in the limit $\alpha\to\infty$, the boundary conditions (5.31) and (5.32) both imply $T_f=T_s$. Hence, the boundary conditions given in Equations (5.31) and (5.32) effectively encapsulate both continuity of flux, Equation (5.19), and continuity of temperature, Equation (5.20). Additionally, in Appendix C, in Figure C.2, we confirm that the analytical solution, given in Section 5.4.2, with the Newton's cooling law boundary condition converges to the solution with the continuity of temperature for large α.

In the remainder of this section, we verify that the numerical solutions to Equations (5.29) and (5.30), along with the boundary conditions, Equations (5.31) and (5.32), converge with the increasing heat transfer coefficient, α, and the substrate size, b. To do this, we consider the following test problem. Assume the metal-air interface is flat and the solution is independent with respect to the x-direction. Hence, the temperature for the 2D problem satisfies the 1D heat equation at any fixed position x, and the 1D analytical solution, presented in Section 5.4.2, holds. We compare the temperature solution obtained from the complete model, to the 1D analytical solution, where we average the temperature over the film height.

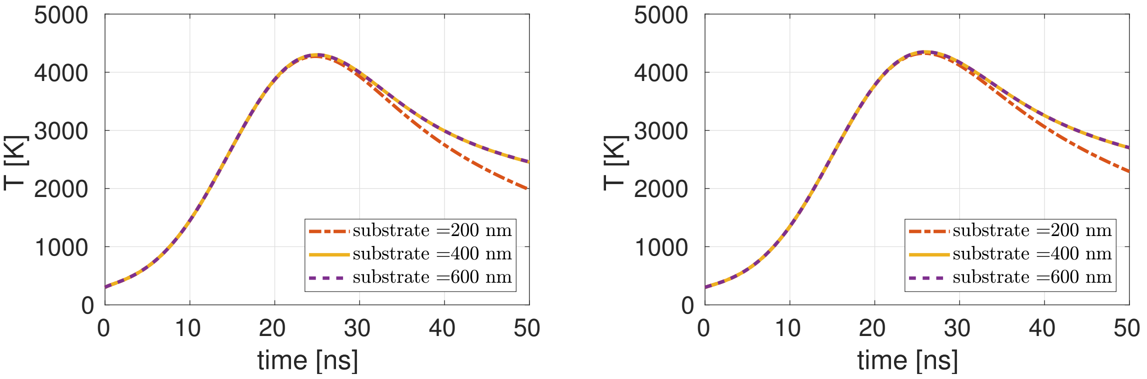


(a) (b)

Figure 5.22 The convergence of the average temperature in the metal film using the numerical solution of the complete model with increasing α for the film height (a) $h_0 = 10\,\text{nm}$ (b) $h_0 = 20\,\text{nm}$. The units of α in the legend are in W m^{-2}K^{-1}.

Figure 5.22 shows the temperature solution of the complete model for a flat film geometry compared to the analytical solution, with temperature averaged over the film height and for increasing values of α. The largest relative difference in the temperature between $\alpha = 50\,\text{W m}^{-2}\text{K}^{-1}$ and $\alpha = 70\,\text{W m}^{-2}\text{K}^{-1}$ is 0.7% for a $h_0 = 10\,\text{nm}$, and 0.6% for a $h_0 = 20\,\text{nm}$. Numerically, larger α decreases the required time-step, thus, in the simulations presented below, we use $\alpha = 50\,\text{W m}^{-2}\text{K}^{-1}$.

Figure 5.23 shows the convergence of the solution of the complete model for the averaged temperature for increasing substrate size, b. The largest relative difference in the temperature between $b = 400\,\text{nm}$ and $b = 600\,\text{nm}$ is 0.12% for a $h_0 = 10\,\text{nm}$, and 0.08% for a $h_0 = 20\,\text{nm}$. Hence, in the simulation presented below, to save computational time we use $b = 400\,\text{nm}$.



(a)

(b)

Figure 5.23 The convergence of the average temperature in the metal film using the solution of the complete model with increasing substrate size, b, for film height (a) $h_0 = 10\,\text{nm}$ (b) $h_0 = 20\,\text{nm}$.

We showed that the solution to the complete model described in this section converges with the increased heat transfer coefficient, α, and the substrate thickness, b, to the analytical solution with continuity of temperature boundary condition at the fluid-substrate interface. Finally, Figure 5.24 shows the comparison of the average temperature in the metal film from the reduced model given in Section 5.4.1, the analytical solution outlined in Section 5.4.2, and the complete model described in this section. The average temperature of a flat metal film in both the reduced and the complete model agrees with the analytical solution. Hence, we are confident in the numerical implementation of the temperature models in our Navier-Stokes solver.

5.4.4 Linear Stability Analysis (LSA) of a Thin Film in Two Dimensions

First, to gain basic insight, we present the results of the LSA carried out within the long wave model for thin films. While the LSA is carried in a setting which differs from the one considered computationally in Section 5.4.5, it is expected that it will still provide a useful insight. The LSA yields the dispersion relation that enables us to predict the stable and unstable perturbation modes. We will see that the reduced model overestimates the thermocapillary effect, and the particular reason for

Table 5.1 The Values of the Material Parameters Used in Simulations

Description	Notation	Value/Expression
Density of the metal	ρ_m	$7900\,kg/m^3$
Density of the substrate	ρ_s	$2200\,kg/m^3$
Room temperature	T_0	$300\,K$
Melting temperature of the metal	T_M	$1728\,K$
Viscosity of the metal at T_M	μ_m	$4.61 \times 10^{-3}\,Pa\,s$
Surface tension coefficient	$\sigma(T)$	$\sigma_0 + \sigma_T(T - T_M)$
Reference surface tension coefficient	σ_0	$1.778\,N/m$
Change of σ with respect to temperature	σ_T	$-3.3 \times 10^{-4}\,N/mK$
Conductivity of the metal	k_m	$90\,W/mK$
Conductivity of the substrate	k_s	$1.4\,W/mK$
Heat capacity of the metal	$(C_p)_m$	$0.44 \times 10^3\,J/kgK$
Heat capacity of the substrate	$(C_p)_s$	$0.712 \times 10^3\,J/kgK$
Laser fluence	E_0	$2500\,J/m^2$
Laser pulse width	t_p	$18 \times 10^{-9}\,s$
Absorption length	α_m	$0.11688 \times 10^{-9}\,m^{-1}$
Fit parameter for reflectance	r_0	0.459363
Fit parameter for reflectance	a_r	$(8.0 \times 10^{-9}\,m)^{-1}$
Equilibrium contact angle of metal with substrate	θ_{eq}	70°
Exponents in in the disjoining pressure model	(n, m)	$(2, 3)$
Precursor film height	h_*	$1.5 \times 10^{-9}\,m$

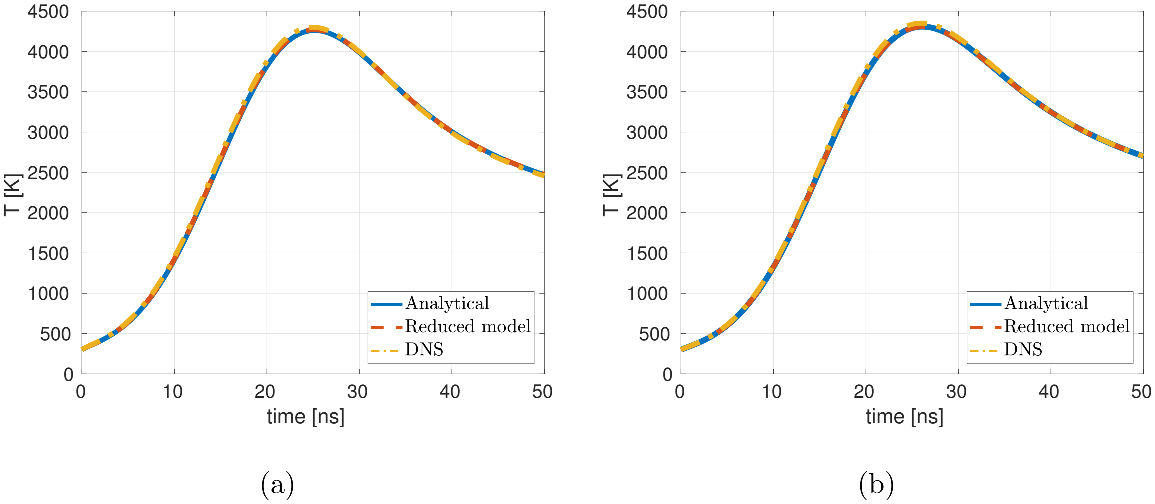


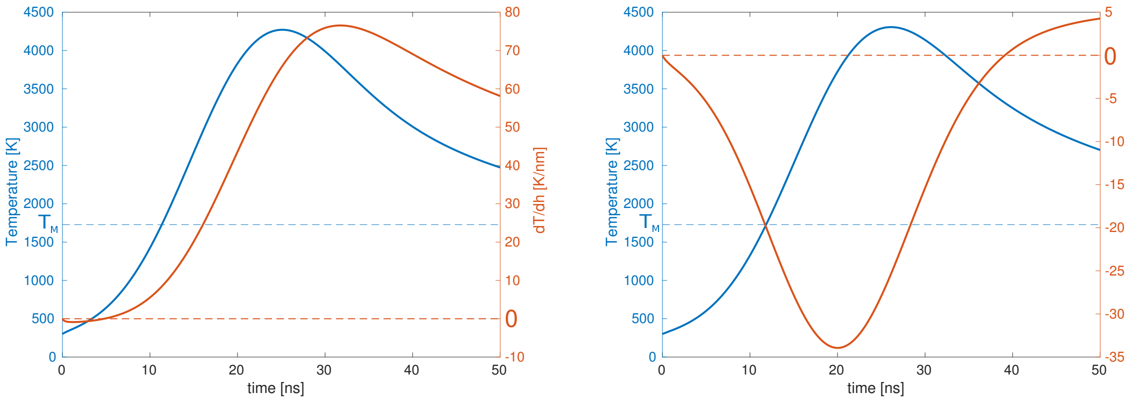
Figure 5.24 Comparison of the average temperature of a flat film for the reduced, the analytical, and the complete model – labeled as "DNS" in the legend. The film height is (a) $h_0 = 10\,\mathrm{nm}$ (b) $h_0 = 20\,\mathrm{nm}$, substrate thickness is $b = 400\,\mathrm{nm}$, and the heat transfer coefficient is $\alpha = 50\,\mathrm{W\ m^{-2}K^{-1}}$.

this is the omission of in-plane heat conduction. We will also see that the temporal temperature variations lead to a change in the surface tension coefficient, which can in turn affect the stability of the perturbed interface during the evolution.

The long wave approximation [77] for a Newtonian film with Marangoni effect and the fluid-substrate interaction in the form of the disjoining pressure [89] leads to the following 4th order nonlinear partial differential equation

$$3\mu\frac{\partial h}{\partial t} + \nabla\cdot\left[\sigma_0 h^3\nabla\nabla^2 h + \frac{3}{2}h^2\nabla\sigma(T) + K_\pi h^2\nabla\left(\frac{h_*^n}{h^n} - \frac{h_*^m}{h^m}\right)\right] = 0, \tag{5.33}$$

where K_π is given in Equation (5.8). We assume the film height is perturbed around the equilibrium height, h_0, as $h(x,t) = h_0 + \varepsilon h_0 e^{\beta t + ikx}$, where ε is a small parameter, β is the growth rate of the perturbation and k is the wavenumber. Note that temperature is not an independent variable here, it is a function of h. An alternative approach is to consider both h and T as independent variables, and perturb each of them separately, however, we are not doing this here for simplicity. Keeping only the



(a) (b)

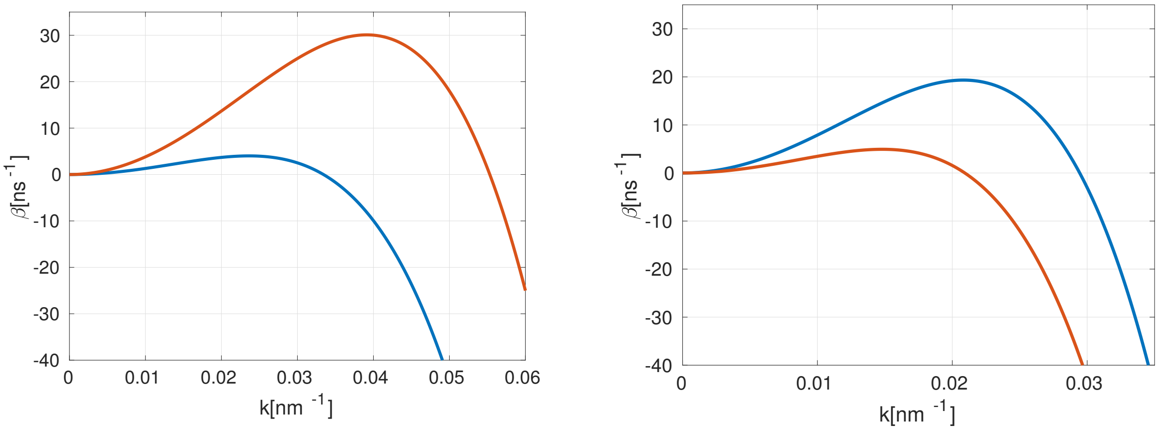
Figure 5.25 The average temperature, T_m^*, and $\partial T_m^*/\partial h$ of a metal film with height (a) 10 nm and (b) 20 nm as a function of time using the reduced model. The blue and orange dashed lines indicate the meting temperature, T_M, and the line $\partial T_m^*/\partial h = 0$, respectively.

leading order terms in ε we obtain the dispersion relation:

$$\beta = -\frac{h_0^2 k^2}{3\mu}\left[\sigma_0 h_0 k^2 - \frac{3}{2}\sigma_T \frac{\partial T}{\partial h} - K_\pi \left(n\frac{h_*^n}{h_0^n} - m\frac{h_*^m}{h_0^m}\right)\right]. \tag{5.34}$$

To illustrate the expected influence of Marangoni effect on stability, in Figure 5.25, we plot the temperature gradient, $\partial T_m^*/\partial h$, for a fixed film height computed using the reduced model (see Equation (5.25)). The temperature of the film, T_m^* (see Equation (5.24)), is plotted to show the melting time of the film. The value of the gradient changes as a function of time, and in order to compute the stability of a perturbed film using Equation (5.34), we approximate $\partial T/\partial h$ by the largest absolute value of $\partial T_m^*/\partial h$, during the time the film is melted. Hence, the dispersion curve provides the upper bound on the influence of the thermocapillary force.

Figure 5.26 shows the dispersion curve for films with $h_0 = 10$ nm and $h_0 = 20$ nm. For the 10 nm thick film, Figure 5.26(a), the thermocapillary force is stabilizing, since $\partial T/\partial h > 0$, for all times while the film is melted. Conversely, for the 20 nm film, Figure 5.26(b), the thermocapillary force is destabilizing for early times after the melting time, since $\partial T/\partial h < 0$.



(a) (b)

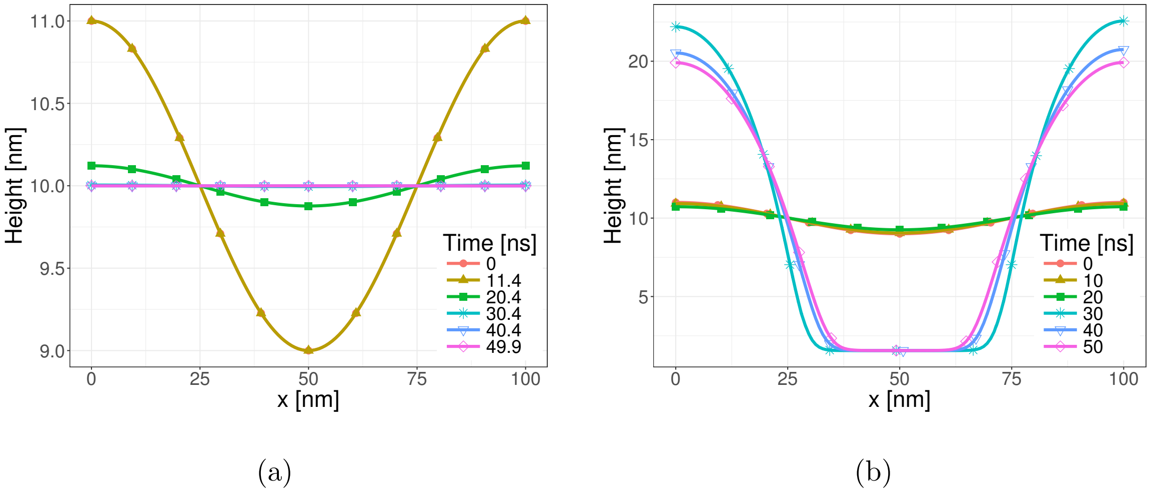
Figure 5.26 The growth rate of a perturbed film with equilibrium film height (a) $h_0 = 10\,\text{nm}$, $\partial T/\partial h = 76.7\,\text{K nm}^{-1}$ and (b) $h_0 = 20\,\text{nm}$, $\partial T/\partial h = -33.9\,\text{K nm}^{-1}$, with (blue) and without (orange) thermocapillary force.

5.4.5 Evolution of a Thin Film Interface in Two Dimensions

In this section, we examine the stability of the films by solving the Navier-Stokes equations including the thermal effects. We also include the fluid-structure interaction in the form of a disjoining pressure (see Equation (5.6)). We compare the influence of thermal effects on the film breakup using the temperature solution from the reduced model and from the complete model, described in Sections 5.4.1 and 5.4.3, respectively.

The initial geometry of the film in the simulations is a cosine perturbation around the height h_0 with the amplitude $\varepsilon = 0.1$. At the beginning of the laser pulse the metal is in the solid state. For the simulations using the reduced temperature model, we simulate only the times after the film is melted. For the simulations with the complete temperature model, we keep the velocity field stationary until the film is melted.

Figure 5.27 shows the evolution of the interface for a 10 nm film. The wavelength of the perturbation, $\lambda = 100\,\text{nm}$, is stable and slightly smaller than the critical wavelength, $\lambda_c = 114\,\text{nm}$ found from the LSA in Section 5.4.4. Hence, we expect



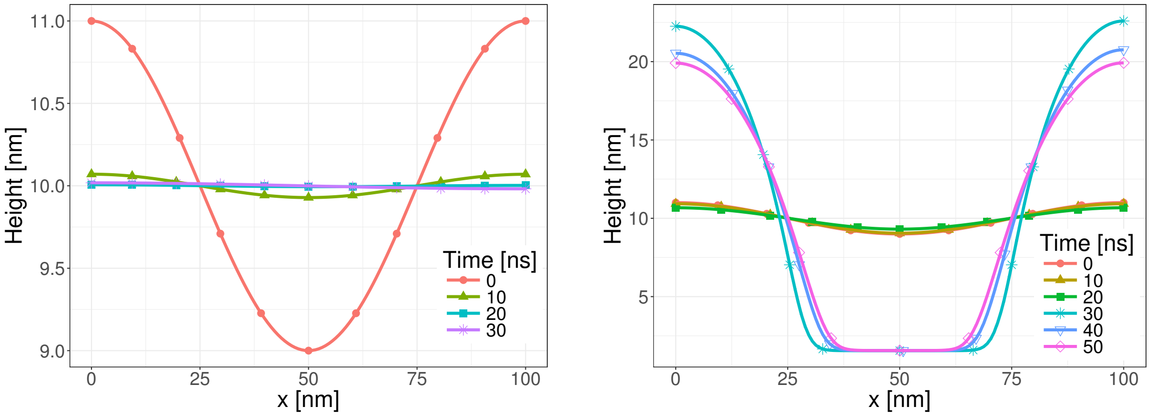
(a)

(b)

Figure 5.27 The comparison of the evolution of the interface using the film temperature solution from (a) the reduced model and (b) the complete model, for film height $h_0 = 10\,\text{nm}$, and the wavelength of the perturbation $\lambda = 100\,\text{nm}$. Note different height scales in (a) and (b).

the perturbation to be stable. Figure 5.27(a) shows the evolution of the interface with temperature solution from the reduced model. The perturbation of the interface decays. Figure 5.27(b) shows the evolution of the interface with the temperature solution from the complete model, where initially (see $t = 20\,\text{ns}$) the perturbation decays, then grows for all following times, and the film eventually breaks into drops. Hence, the two temperature models, which agree for a flat film, produce different evolution for a perturbed interface.

In order to examine the source of the difference in the evolution between the two models, we now consider the influence of the temperature on the normal component of the surface force, and ignore the tangential (thermocapillary) component. Figure 5.28 shows the evolution of the interface where the surface tension coefficient is (a) constant, $\sigma = \sigma_0$, and (b) temperature dependent, $\sigma = \sigma(T)$, but the Marangoni effect is not included; the complete model is used for temperature calculations. In the simulation with the constant surface tension coefficient the perturbation is stable, as expected from the dispersion relation in Equation 5.34. However, when the surface tension coefficient depends on the temperature, the perturbation initially decays

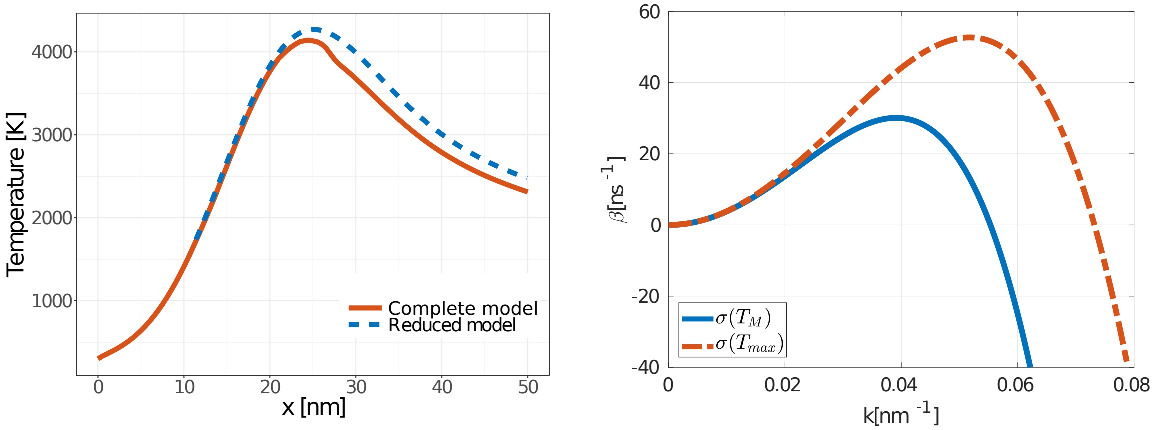


(a) (b)

Figure 5.28 The comparison of the evolution of the interface ignoring the thermocapillary force, with the surface tension coefficient (a) fixed to $\sigma = \sigma_0$ and (b) $\sigma = \sigma(T)$, with the temperature solution from the complete model for the film height $h_0 = 10\,\text{nm}$, and the wavelength of the perturbation $\lambda = 100\,\text{nm}$.

(see $t = 20\,\text{ns}$), but grows for all later times. Note in particular that the results shown in Figures 5.27(b) and 5.28(b) are almost identical, suggesting further that the Marangoni effect is essentially irrelevant in the present context. Therefore, the stability change is not due to the thermocapillary force (spatial variations of σ), but due to the change in the normal component of the surface tension force (temporal change of σ).

Figure 5.29(a) shows the average temperature of the film in the reduced and complete models from the results in Figure 5.27. According to both models, from the melting time at $t = 11.4\,\text{ns}$, the temperature of the film increases to $T \gtrsim 4000\,\text{K}$, which corresponds to the decrease in the surface tension coefficient from $\sigma(T_M) = 17.78\,\text{N m}^{-1}$ to $\sigma(T_{max}) = 10.15\,\text{N m}^{-1}$. Figure 5.29(b) shows the dispersion curve computed using Equation (5.34), with $\sigma_0 = \sigma(T_M)$ and $\sigma_0 = \sigma(T_{max})$, and $\partial T/\partial h = 0$. The change in σ_0 shifts the critical wavenumber, k_c, and the stable perturbation ($\lambda = 100\,\text{nm}$ which corresponds to $k \approx 0.0628\,\text{nm}^{-1}$) becomes unstable. As a result, the stable mode in simulations in Figures 5.27(b) and 5.28(b) becomes unstable as the film temperature increases, eventually leading to the film breakup.

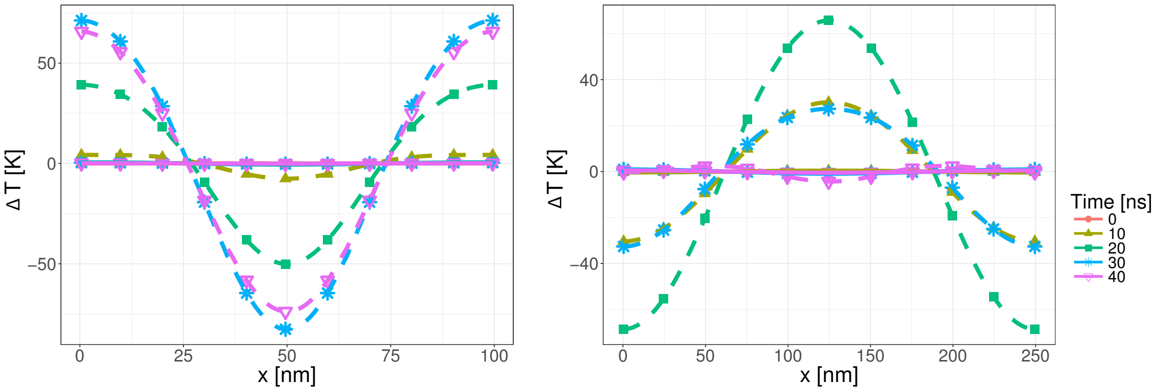


(a)

(b)

Figure 5.29 (a) The average temperature of the metal film from Figure 5.27. (b) Growth rate given in Equation (5.34), for $h_0 = 10\,\text{nm}$, $\partial T/\partial h = 0$, and σ_0 at the melting temperature, T_M, and the maximum temperature predicted by the reduced model, T_{max}.

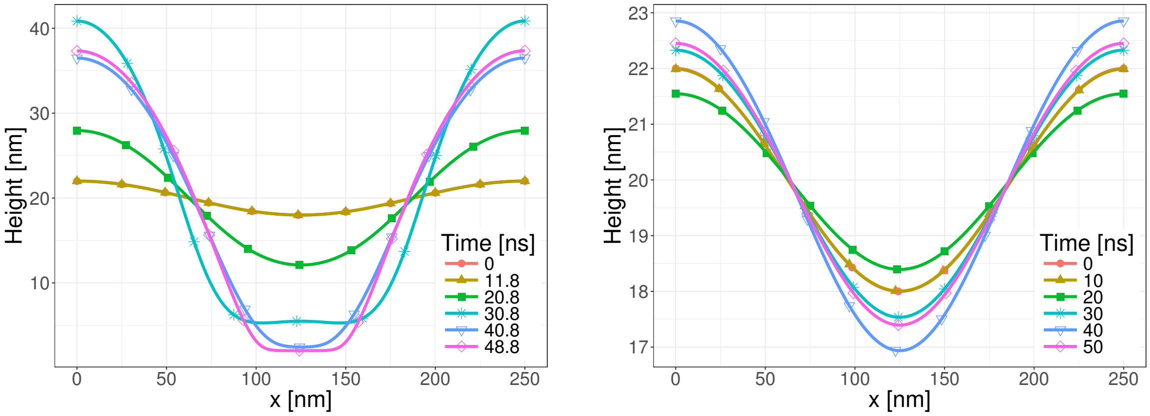
In contrast to the simulations with the complete temperature model, in Figures 5.27(b) and 5.28(b), the simulation with the temperature from the reduced model in Figure 5.27(a) remains stable, despite the fact that the average temperature of the film increases similarly as in the complete model, as shown in Figure 5.29(a). To gain better understanding of this result, we examine the difference in the temperature solutions along the liquid-air interface for a perturbed stationary film (the motivation for considering stationary film is due to the source term dependence on the film height, and the film evolution would affect the source term). We consider a film with the same geometry as the initial condition in the simulations in Figures 5.27 and 5.28. Figure 5.30(a) shows the difference of the temperature along the interface from the average temperature at different times. The temperature at the interface varies significantly more in the reduced model compared to the complete model. Thus, the temperature gradients at the liquid-air interface are larger in the reduced model compared to the complete model and the stabilizing thermocapillary force prevents the interface in Figure 5.27(a) from becoming unstable with the change in the temperature. This finding explains the different film evolution between the two models.



(a) (b)

Figure 5.30 The difference between temperature and average temperature, ΔT, at the interface of a static perturbed film for a subset of times. The dashed line shows the temperature solution of the reduced model, and the full line shows the temperature solution of the complete model, for (a) $h_0 = 10\,\text{nm}$ and $\lambda = 100\,\text{nm}$, (b) $h_0 = 20\,\text{nm}$ and $\lambda = 250\,\text{nm}$.

Next, we examine the influence of the thermocapillary force on the film of height $h_0 = 20\,\text{nm}$. Similarly as for $h_0 = 10\,\text{nm}$, we impose a perturbation of the film height with a wavelength, $\lambda = 250\,\text{nm}$, which is stable when thermocapillary effects are ignored, based on the dispersion relation, Equation (5.34). Figure 5.31 shows the comparison of the evolution of the interface using the reduced and complete temperature models. Using the reduced model, in Figure 5.31(a), the interface is unstable, and the perturbation grows until the film breaks into drops. This is not surprising since the reduced temperature model predicts $\partial T_m^*/\partial h < 0$ which destabilizes the film (see Figure 5.26(b)). In the simulation with the temperature from the complete model, in Figure 5.31(b), the stability of the film changes multiple times: at $t = 20\,\text{ns}$ the perturbation decays; at $t = 30\,\text{ns}$ and $t = 40\,\text{ns}$ the perturbation grows; and at $t = 50\,\text{ns}$ the perturbation decays again. Similarly as for $h_0 = 10\,\text{nm}$, to explain these dynamics, we examine the simulations without the thermocapillary force, and investigate the influence of the temperature on the normal component of the surface force.

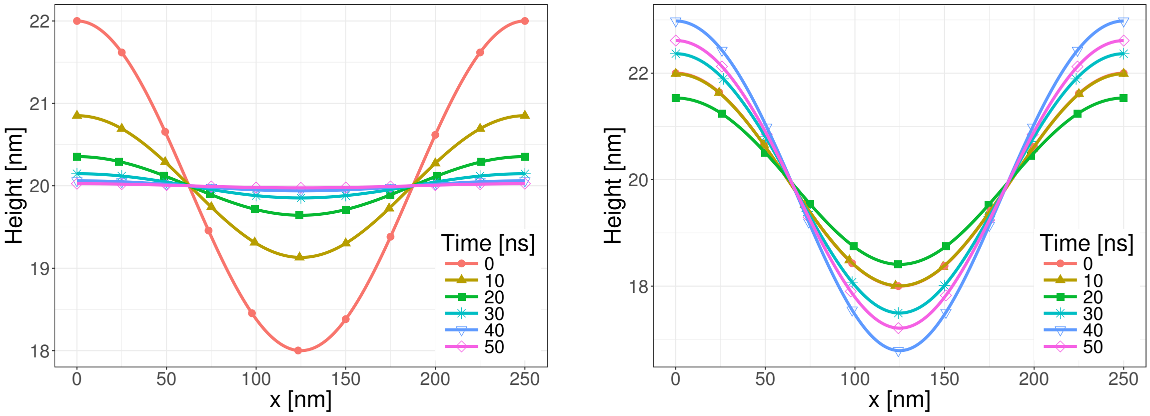


(a) (b)

Figure 5.31 The comparison of the evolution of the interface using the film temperature solution from (a) the reduced, (b) the complete model for the film height $h_0 = 20\,\text{nm}$, and the wavelength of perturbation $\lambda = 250\,\text{nm}$.

Figure 5.32 shows the evolution of the interface where the surface tension coefficient is (a) constant, $\sigma = \sigma_0$, and (b) temperature dependent, $\sigma = \sigma(T)$, where we ignore the thermocapillary effect. In the simulation with the constant surface tension, the perturbation is stable, as expected from the LSA. However, the simulation with temperature dependent surface tension shows the same dynamics as the simulation in Figure 5.31(b). Thus, we see again that the thermocapillary force is negligible and the changes in the stability in the complete model are due to the temporal changes of the surface tension, as we show next.

Figure 5.33(a) shows the average temperature of the film using the reduced and complete models from the results in Figure 5.31. From the melting time, at $t = 11.8\,\text{ns}$, the temperature rises to reach $T \gtrsim 4000\,K$, which corresponds to the decrease in the surface tension coefficient from $\sigma(T_M) = 17.78\,\text{N m}^{-1}$ to $\sigma(T_{max}) = 10.05\,\text{N m}^{-1}$. Figure 5.33(b) shows the dispersion curve computed using Equation (5.34), with $\sigma_0 = \sigma(T_M)$ and $\sigma_0 = \sigma(T_{max})$, and $\partial T/\partial h = 0$. The change in σ_0 shifts the critical wavenumber, k_c, and the stable perturbation ($\lambda = 250\,\text{nm}$ which corresponds to $k \approx 0.02512\,\text{nm}^{-1}$) becomes unstable, as we see in Figures 5.31(b) and 5.32(b) after



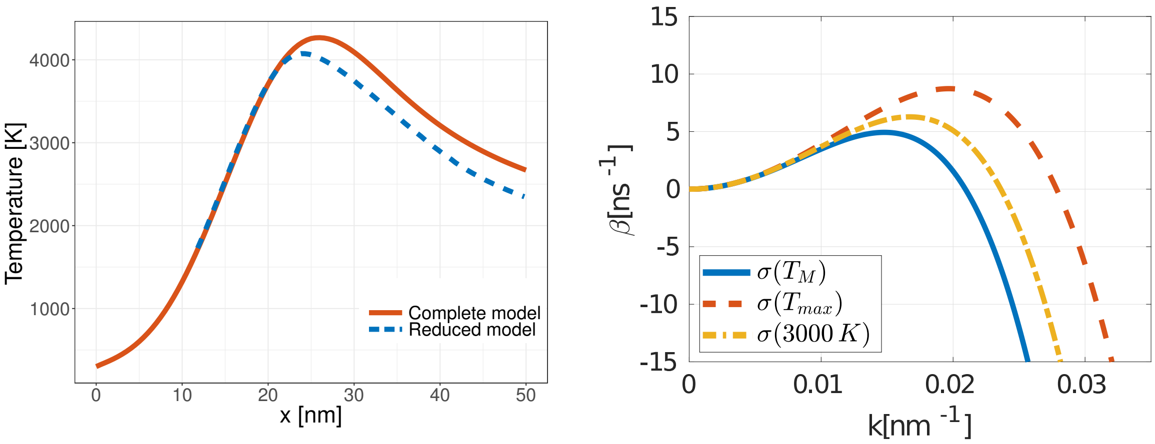
(a)

(b)

Figure 5.32 The comparison of the evolution of the interface ignoring the thermocapillary force, with the surface tension coefficient (a) fixed to $\sigma = \sigma_0$ and (b) $\sigma = \sigma(T)$, with the temperature solution from the complete model for film height $h_0 = 20\,\text{nm}$, the wavelength of the perturbation $\lambda = 250\,\text{nm}$.

$t = 20\,\text{ns}$. After $t = 40\,\text{ns}$, the temperature decreases again to $T \approx 3000\,\text{K}$, hence, the perturbation becomes stable again. In summary, similar to the $h_0 = 10\,\text{nm}$ film, the stability of the interface using the complete model is governed by the temporal variations of σ.

Similarly as for the $h_0 = 10\,\text{nm}$ film, the temporal changes of the surface tension do not explain the film instability for the temperature from the reduced model in Figure 5.31(a). Therefore, we compare again the temperature at the liquid-air interface of a stationary film using the reduced and complete models. Figure 5.30(b) shows the difference in the temperature from the average temperature at the interface of a stationary film with the same initial condition as in the simulations shown in Figure 5.31. We see once again that the effect of the thermocapillary force is augmented by the reduced model.



(a)

(b)

Figure 5.33 (a) The average temperature of the metal film from Figure 5.27. (b) Growth rate given by Equation (5.34), for $h_0 = 20\,\text{nm}$, $\partial T/\partial h = 0$, and σ at the melting temperature, T_M, at the maximum temperature predicted by the reduced model, T_{max}, and at the intermediate value of 3000 K.

5.4.6 The Influence of the Temperature Dependent Viscosity on the Breakup of 2D Films

Here we study the influence of the viscosity variations with temperature of the metal film on the stability and breakup dynamics. During the metal heating and melting process, the viscosity of the metal changes several orders of magnitude [92]. The viscosity of most metals can be modeled by an exponential as

$$\mu\left(T\right) = \mu_0 \exp\left(\frac{E}{RT}\right) \tag{5.35}$$

where $\mu_0 = 0.1663\,\text{mN s m}^{-2}$ and $E = 50.2\,\text{kJ mol}^{-1}$ are constants dependent on the material, and $R = 8.3144\,\text{J K}^{-1}\text{mol}^{-1}$ is the gas constant [92]. Figure 5.34 shows the viscosity of Nickel as a function of temperature.

The influence of the temperature dependent viscosity on the film breakup can be examined from the dispersion relation in Equation (5.34): the stability of the film and the critical wave number, k_c, do not depend on the viscosity. However, the growth rate, β, is inversely proportional to μ. Figure 5.35 shows the comparison of the

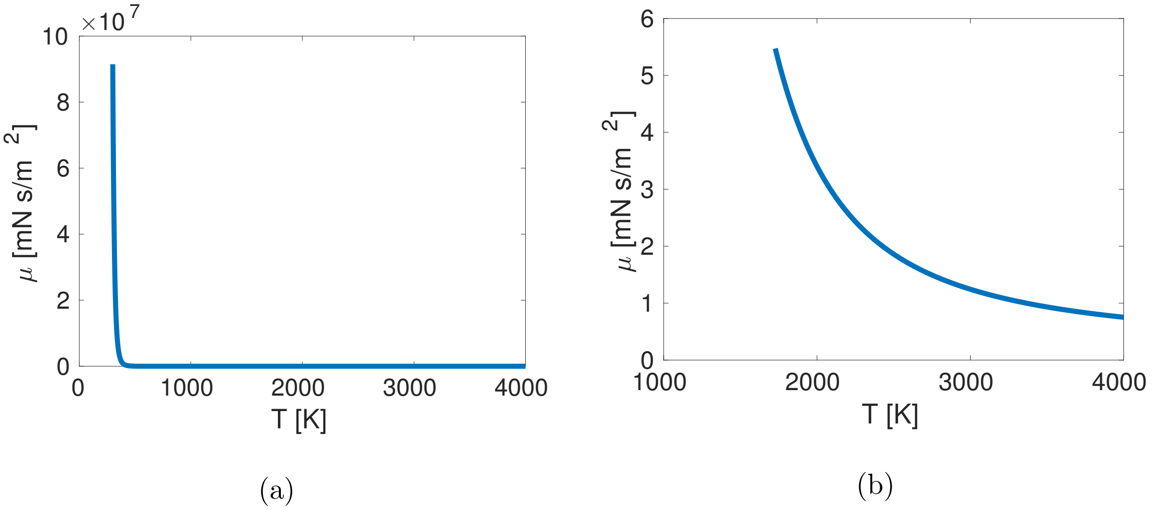
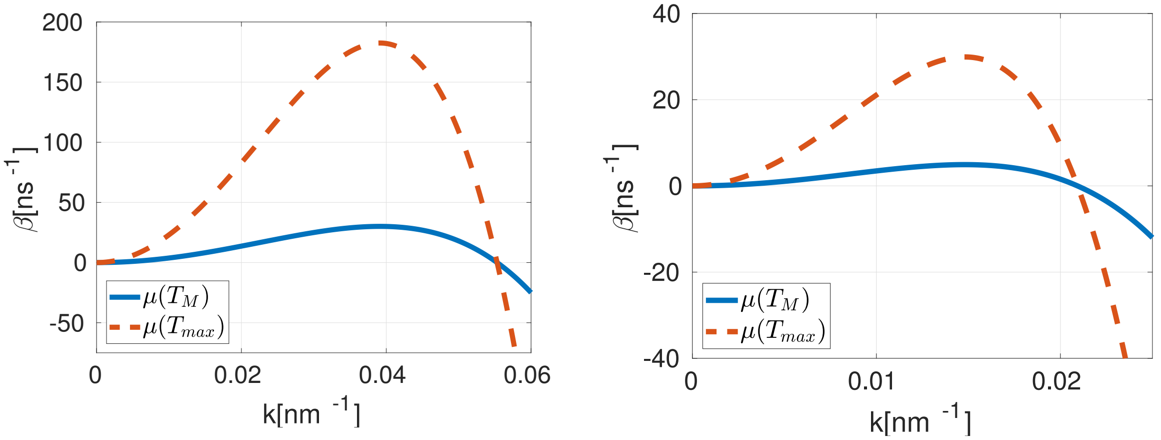


Figure 5.34 Viscosity of Nickel as a function of temperature given in Equation (5.35), for temperature range starting from (a) room temperature and (b) melting temperature of Nickel.

dispersion curves for μ at the melting temperature and at the maximum temperature. The maximum temperature is approximated by the temperature solution of a flat film obtained by the reduced model (see Section 5.4.1).

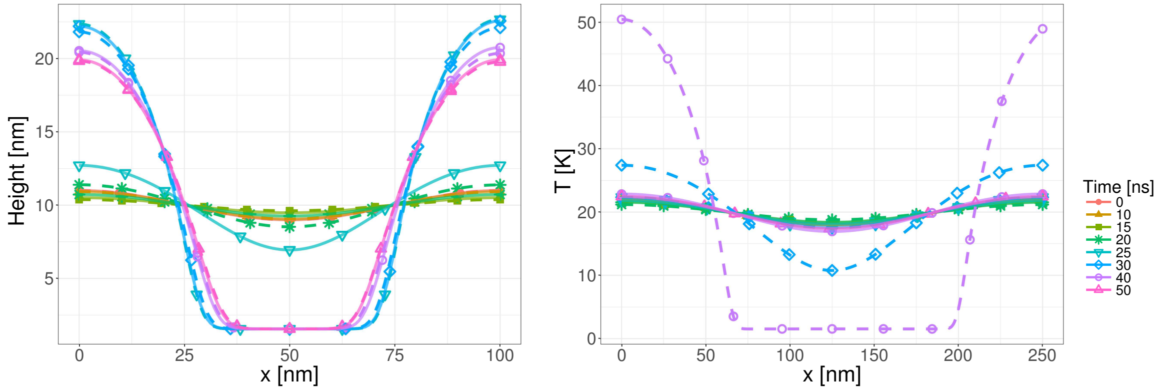
Next we study the influence of the variable viscosity on the breakup dynamics. We use the same initial geometry as in Section 5.4.5, and we implement the complete model described in Section 5.4.3. Figure 5.36 shows the evolution of the film interface with temperature dependent viscosity compared to the evolution for constant viscosity, $\mu = \mu(T_M)$. For the 10 nm film, the same evolution dynamics are present in both cases: perturbations initially decay, but start growing as the film temperature rises (see the discussion related to Figure 5.27(b) and Figure 5.28(b) above). As we expected from the LSA, the stability of the perturbations is not affected by the variable viscosity, but their growth rate is faster, and therefore the breakup time occurs ≈ 5 ns faster with variable viscosity compared to the constant viscosity case (see Section 5.4.5). For the 20 nm film, again, the decay and the growth of the perturbations for the variable viscosity follows the same direction as the constant viscosity in Figures 5.31(b) and 5.32(b). During the time of the perturbation growth, as in $t = 20$ ns to $t = 40$ ns in the



(a) (b)

Figure 5.35 Growth rate of a perturbed film from Equation (5.34), with $\partial T/\partial h = 0$, and film height (a) $h_0 = 10\,\text{nm}$ and (b) $h_0 = 20\,\text{nm}$.

$\mu = \mu(T_M)$ case, the perturbation in the $\mu = \mu(T)$ case grows fast enough that the film breaks. Recall that in the $\mu = \mu(T_M)$ case, the stability changes after $t = 40\,\text{ns}$, and the film stabilizes due to the decrease of the film temperature. Therefore, inclusion of temperature dependent viscosity can strongly influence the film evolution.



(a) (b)

Figure 5.36 The comparison of the evolution of the film interface with constant (full line) and temperature dependent viscosity (dashed line) using the complete temperature model, for (a) $h_0 = 10\,\text{nm}$ and (b) $h_0 = 20\,\text{nm}$.

5.4.7 Breakup of Liquid Metal Filaments

Previously, we showed that the breakup of liquid metal filaments can be directed via the surface tension variations along the filament (see Section 5.2). Here we examine the influence of the temperature gradients on the breakup of the filaments. We consider a similar initial geometry as in Section 5.2. However, here we focus on the perturbation wavelengths close to the critical ones, similarly as in Section 5.4.5.

The simulation setup is as follows: the initial geometry is a flat filament with rectangular perturbations (see Section 5.2). The filament height is $h_0 = 8\,\mathrm{nm}$, the perturbation height is $\Delta h = 8\,\mathrm{nm}$ and the width of the filament is $w = 185\,\mathrm{nm}$. Since we know from Section 5.4.5 that the temperature gradients are small in the metal film, we increase the height of the rectangular perturbations compared to Section 5.2 in an attempt to increase the temperature gradients. However, the average filament height is kept at 12 nm as the filament height in Section 5.2. We simulate one half of the perturbation wavelength and half of the filament width, and impose symmetry boundary conditions at all in-plane directions. The fluid is stationary until the melting time of the filament (see Section 5.4.5).

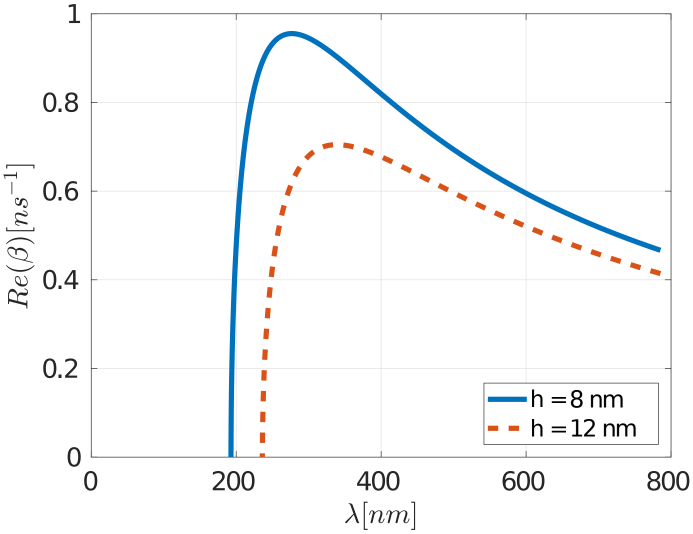
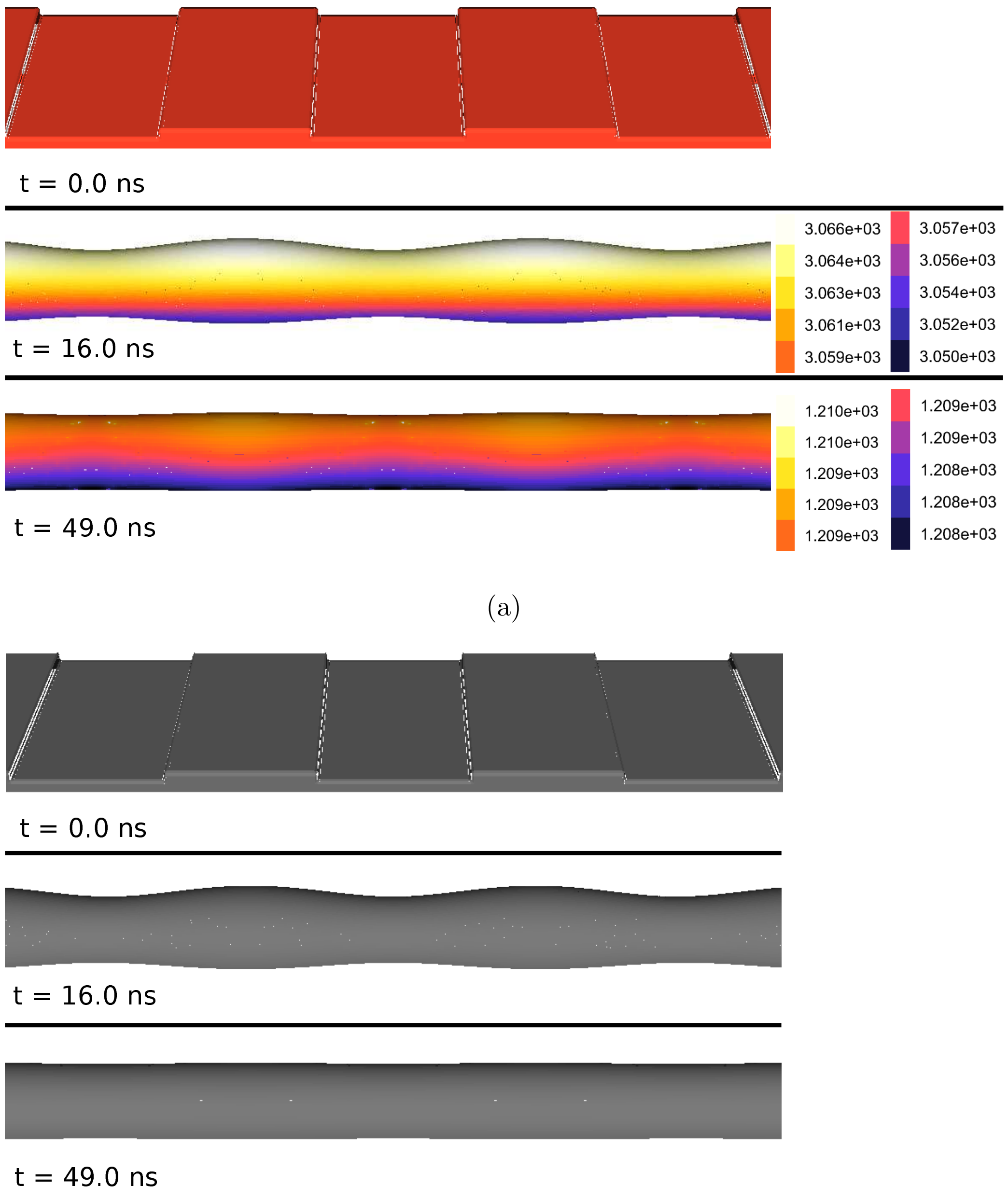


Figure 5.37 Growth rate as a function of the wavelength for filament width $w = 185\,\mathrm{nm}$. Since the perturbation is large, the growth rate is computed both with $h_0 = 8\,\mathrm{nm}$ and $h_0 = 12\,\mathrm{nm}$ (average height).

The stability of a cylindrical filament on a substrate is governed by the Rayleigh-Plateau (RP) type of instability, where the growth rate of the perturbations can be predicted based on Equation (5.3). Figure 5.37 shows the growth rate for a filament where we consider both the filament height without the perturbation, $h_0 = 8\,\text{nm}$ and the average filament height including the perturbations, $h_0 = 12\,\text{nm}$ (see Equation (5.4)). The RP stability curve gives us an approximation for the critical wavelength, $\lambda_c \sim 236\,\text{nm}$ using $h_0 = 12\,\text{nm}$. In our simulations we pick a range of wavelengths in the interval which includes the critical wavelength for the both heights h_0. Below we show results for two filaments: one with a stable and one with an unstable perturbation. Note that both results have perturbation wavelength larger than λ_c predicted by the RP theory. However, the RP theory is developed for a free standing jet, and the presence of the substrate slightly modifies λ_c [93]. We compare the results for simulations with and without the thermal effects. The temperature is governed by the complete model described in Section 5.4.3.

Figure 5.38 shows the evolution of a stable filament, with the wavelength of the perturbations close to the critical one. The results are similar, independently of whether the surface tension is treated as a constant or temperature dependent. First, the thermocapillary force is weak: the variations of the temperature at the filament surface are small. Second, unlike in the flat film case (see Section 5.4.4), the decrease of the surface tension does not change the stability of the filament; it only decreases the growth rate. Hence, the thermal variations of the surface tension do not change the qualitative behavior.

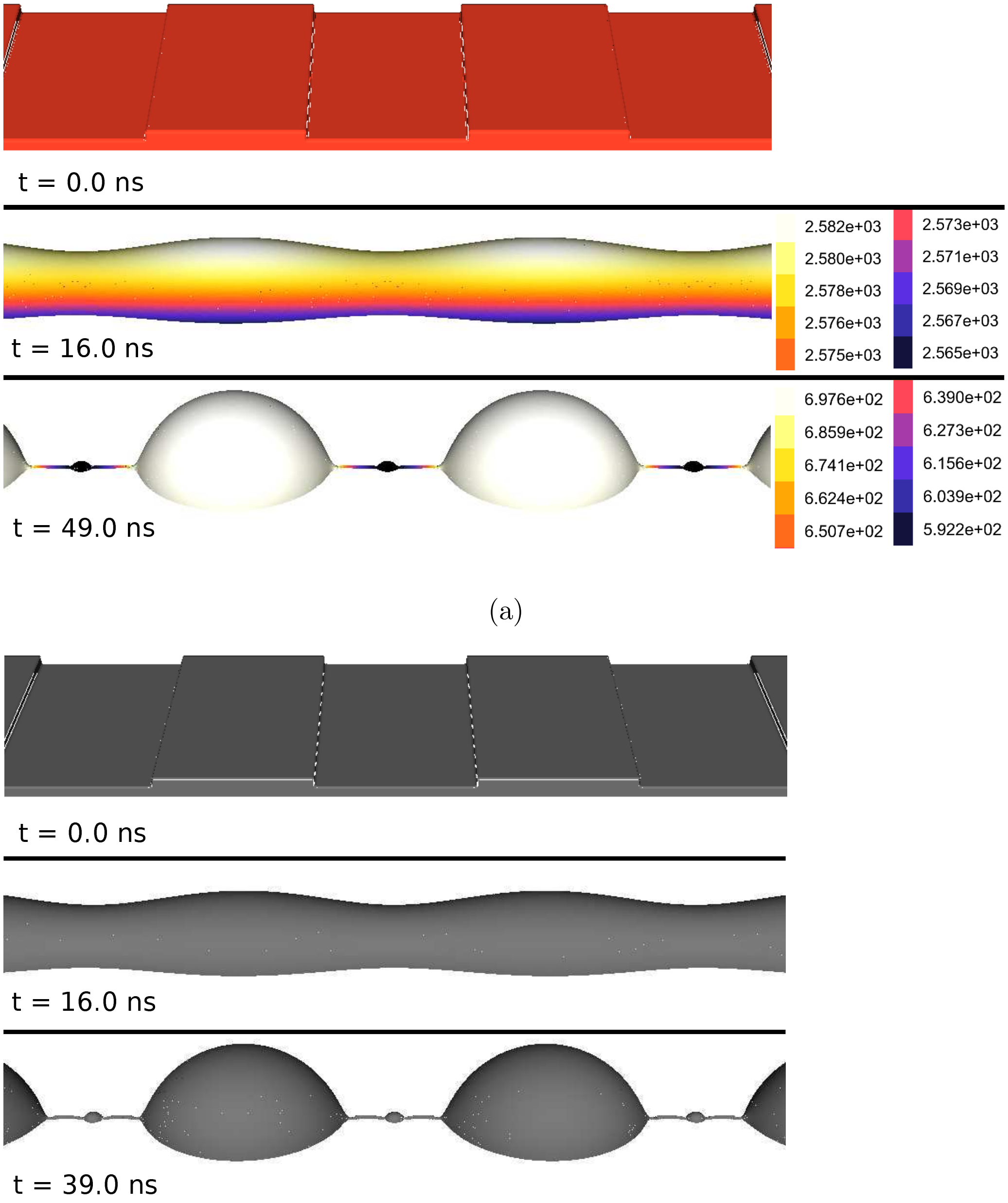
Figure 5.39 shows the evolution of an unstable filament. Again, the thermocapillary force does not change the qualitative breakup dynamics. However, the breakup with temperature dependent surface tension happens about 10 ns slower compared to the constant surface tension. This is expected, since the increase in the temperature



(a)

(b)

Figure 5.38 Evolution of a stable filament with wavelength $\lambda = 240\,\text{nm}$, (a) surface tension dependent on the temperature and (b) surface tension fixed at σ_0. The color in part (a) represents the temperature at the interface in degrees Kelvin.



(a)

(b)

Figure 5.39 Evolution of an unstable filament with wavelength $\lambda = 250\,\text{nm}$, (a) surface tension dependent on the temperature and (b) surface tension fixed at σ_0.The color in part (a) represents the temperature at the interface in degrees Kelvin.

leads to a decrease in the surface tension, which in turn leads to a decrease of the growth rate.

5.4.8 Conclusions

In this section we have studied the influence of the thermal effects on the evolution of thin metal films and filaments. For the films, we have shown that the dynamics of the evolution can change due to the surface tension dependence on temperature. Surprisingly, the influence of the temperature is not manifested through the thermocapillary force, but through the capillary force. More precisely, the thermal effects influence the interface evolution due to the time-dependent changes of the surface tension during a laser pulse.

We have also shown two different models for computing the film temperature. The reduced 1D temperature model (Section 5.4.1) overestimates the temperature gradients along the interface, due to the lack of the in-plane heat conduction. The complete model, based on the numerical computation of the temperature (Section 5.4.3) shows that the temperature gradients along the interface are in fact not strong enough to influence the breakup of the films. The changes in the viscosity during the metal film heating can accelerate the growth of the perturbations, leading to a breakup of films that would not break if a constant value of viscosity at the melting temperature was used.

In summary, thermal effects can lead to the changes in the metal film stability through the temporal change in the surface tension and viscosity.

In the case of filaments, the temperature dependence has little influence on the qualitative behavior of the breakup of the liquid metal filaments. At least for the parameters considered here, the stability of (single metal) filaments is not influenced by the variation of the surface tension. This is in contrast to the two-metal filaments, considered in Section 5.2, where concentration dependence of surface tension is much stronger and can qualitatively change the dynamics.

CHAPTER 6

CONCLUSIONS AND FUTURE DIRECTIONS

We have developed a new numerical methodology for including variable surface tension in a VoF based Navier-Stokes solver. The method handles both temperature or concentration dependent surface tension variations. We employ a height function inspired formulation to compute surface gradients and the resulting stresses at the interface (Marangoni forces) in a more general numerical framework. We show the accuracy of our developed method by studying the convergence of the computation of the surface gradient for multiple geometries and the convergence of the terminal velocity for the classical problem of the drop migration with an imposed constant temperature gradient. The drop migration simulation results are in agreement with the available theoretical and numerical results. We also show that our method produces results consistent with experimental data in the case of concentration dependent surface tension. Our numerical implementation extends to adaptively refined meshes, which improves the computational efficiency for Marangoni induced flows that require a high resolution around the interface.

The presented approach represents a first attempt for implementing a general variable surface tension in the VoF method. As presented here, our method can subsequently be used directly for surface tension dependence on the surfactant concentration. This includes implementing the solution to the surfactant transport equation for soluble and insoluble surfactants. Our methodology can provide tools for developing more robust and accurate numerical simulations for two-phase flows with surfactants. Surfactant flows have many applications, e.g., in the chemical industry, pharmaceuticals and technology [94], and their understanding will have far reaching effects in many areas.

The numerical verifications and validations with available literature demonstrate the efficiency and applicability of our methodology. Our numerical approach is implemented in an adaptive mesh refinement framework, which now makes detailed numerical simulations that incorporate the effects of tangential (Marangoni) stresses feasible. This is particularly relevant for a number of flow problems where the Marangoni effect may play a crucial role, such as the evolution of thin films on the nanoscale, where Marangoni effects may result either from concentration gradients (mixture of two fluids) or thermal gradients due to internal or external sources. Our future research will continue in this direction.

APPENDIX A

LASER SOURCE TERM

The absorption, reflectance and transmittance of a thin metal film can be computed from Maxwell's equations with appropriate boundary conditions. The equations are greatly simplified when considering a single film layer on a transparent (non-absorbing) substrate. The simplified expressions for computing reflectance and transmittance given in [69] are

$$R_1 = \frac{t_{12}^2 + u_{12}^2}{p_{12}^2 + q_{12}^2}, \tag{A.1}$$

$$T_1 = \frac{n_2}{n_0} \frac{((1+g_1)^2 + h_1^2)((1+g_2)^2 + h_2^2)}{e^{2\alpha_1} + (g_1^2 + h_1^2)(g_2^2 + h_2^2)e^{-2\alpha_1} + C\cos(2\gamma_1) + D\sin(2\gamma_1)} \tag{A.2}$$

where the terms in Equations (A.1) and (A.2) are defined as

$$\begin{aligned}
\alpha_1 &= \frac{2\pi k_1 h}{\lambda_l}, & \gamma_1 &= \frac{2\pi n_1 h}{\lambda_l} \\
g_1 &= \frac{n_0^2 - n_1^2 - k_1^2}{(n_0 + n_1)^2 + k_1^2}, & g_2 &= \frac{n_1^2 - n_2^2 + k_1^2}{(n_1 + n_2)^2 + k_1^2} \\
h_1 &= \frac{2n_0 k_1}{(n_0 + n_1)^2 + k_1^2}, & h_2 &= \frac{-2n_2 k_1}{(n_1 + n_2)^2 + k_1^2} \\
C &= 2(g_1 g_2 - h_1 h_2), & D &= 2(g_1 h_2 + g_2 h_1) \\
p_2 &= e^{\alpha_1}\cos(\gamma_1), & p_{12} &= p_2 + g_1 t_2 - h_1 u_2 \\
q_2 &= e^{\alpha_1}\sin(\gamma_1), & q_{12} &= q_2 + h_1 t_2 + g_1 u_2 \\
t_2 &= e^{-\alpha_1}(g_2\cos(\gamma_1) + h_2\sin(\gamma_1)), & t_{12} &= t_2 + g_1 p_2 - h_1 q_2 \\
u_2 &= e^{-\alpha_1}(h_2\cos(\gamma_1) - g_2\sin(\gamma_1)), & u_{12} &= u_2 + h_1 p_2 + g_1 q_2
\end{aligned}$$

and h is the metal film height, λ_l is the wavelength of the incident radiation, n_0 is the refractive index of air, n_1 and k_1 are the metal refractive index and the extinction coefficient respectively, and n_2 is the refractive index of the substrate.

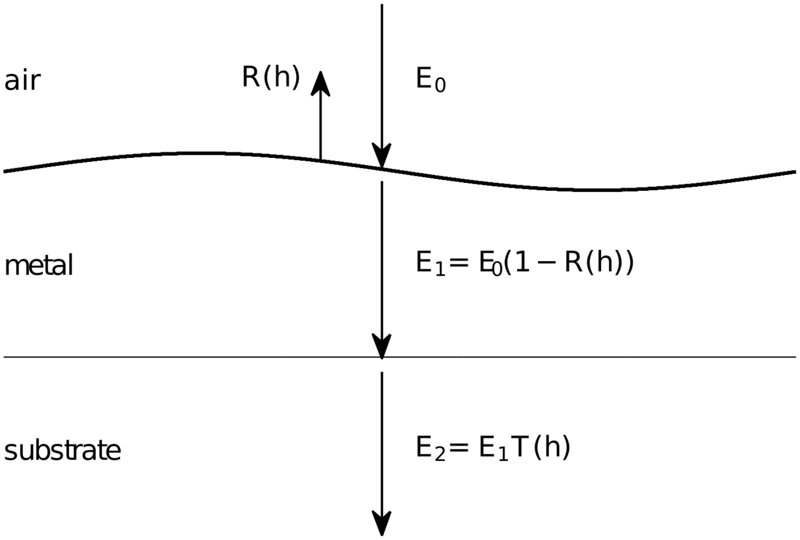


Figure A.1 E_0 is the intensity of the incident radiation. $R(h)$ and $T(h)$ are the height dependent reflectance and transmittance of the metal.

Figure A.1 shows a schematic of the laser energy absorption. The incident energy E_0 is perpendicular to the film surface. Part of the energy, $R(h)$, is reflected at the film surface, and the rest of the energy, denoted by E_1, penetrates the surface. Then part of the laser energy, denoted E_2 is transmitted through the metal. Hence the energy absorbed by the metal film is

$$A = E_0\left[1 - T(h)\right]\left[1 - R(h)\right]. \tag{A.3}$$

The expressions for R and T given in Equations (A.1) and (A.2), can be approximated by simpler functions, as it is done by Trice et al. [58]

$$T_2(h) = e^{-\alpha_m h}, \qquad R_2(h) = r_0\left(1 - e^{-a_r h}\right), \tag{A.4}$$

where $\alpha_m = \frac{4\pi k_1}{\lambda}$, and r_0 and a_r can be found by fitting the simplified expressions to the expressions in Equations (A.1) and (A.2). Figure A.2 shows the comparison of the reflectance and transmittance given by Equations (A.1) and (A.2) and Equation (A.4). The parameters used here are $\lambda_l = 248\,\mathrm{nm}$, $n_0 = 1$, $n_1 = 1.7167$, $k_1 = 2.3067$,

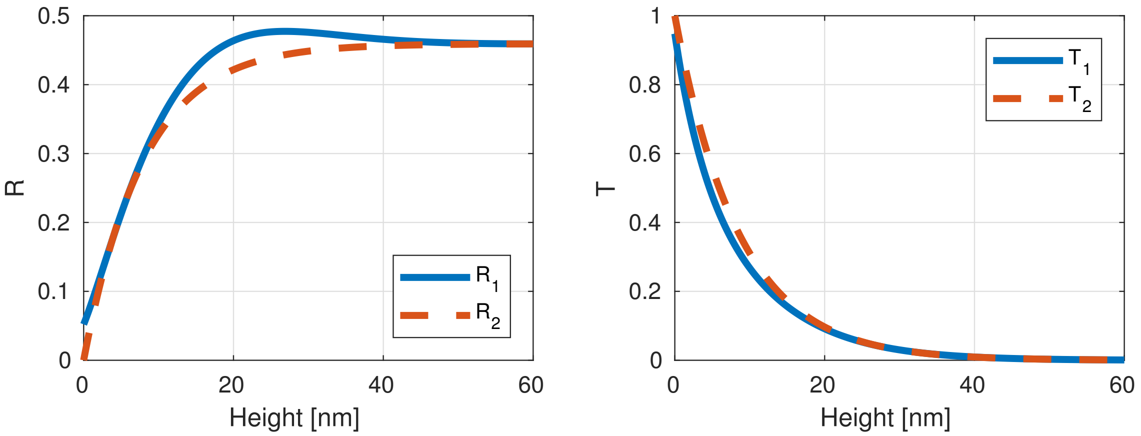


Figure A.2 The comparison of the reflectance and transmittance given by the Equations (A.1) and (A.2) and Equation (A.4).

$n_2 = 1.59157$, $R_0 = 0.4594$, and $a_r^{-1} = 8\,\text{nm}$. In Section 5.4 we use the simplified expression for computing the absorption of the laser energy by a metal film.

APPENDIX B

THE TEMPERATURE SOLUTION OF THE REDUCED MODEL IN THE LIMIT OF SMALL FILM HEIGHT

The solution to the reduced temperature model given in Section 5.4.1, contains integrals that pose numerical difficulties for small film heights. Here we give expressions that can be used for computing the temperature of the metal film, T_m^*, and the gradient of the temperature with respect to the film height, $\partial T_m^*/\partial h$ to alleviate those difficulties. In the limit of small film height, T_m^* and $\partial T_m^*/\partial h$, can be expanded using asymptotic series as

$$\begin{aligned} T_m^*\left(h \to 0, t\right) = T_0 + S^*\, e^{-\frac{t_p^2}{2\sigma_{tp}^2}} \int_0^t \exp\left(-\frac{\left(t-u\right)^2}{2\sigma_{tp}^2} + \right. \\ \left. \frac{t_p}{\sigma_{tp}^2}\left(t-u\right)\right)\left[\frac{h}{C_K\sqrt{\pi\, u}} - \frac{h^3}{2\sqrt{\pi}\left(C_K\sqrt{u}\right)^3} + ...\right] du, \end{aligned} \tag{B.1}$$

$$\begin{aligned} &\frac{\partial T}{\partial h}\left(h \to 0, t\right) = \\ &\frac{\partial S^*}{\partial h}\, e^{-\frac{t_p^2}{2\sigma_{tp}^2}} \int_0^t \exp\left(-\frac{\left(t-u\right)^2}{2\sigma_{tp}^2} + \frac{t_p}{\sigma_{tp}^2}\left(t-u\right)\right)\left[\frac{h}{C_K\sqrt{\pi\, u}} - \frac{h^3}{2\sqrt{\pi}\left(C_K\sqrt{u}\right)^3} + ...\right] du + \\ &S^*\, e^{-\frac{t_p^2}{2\sigma_{tp}^2}} \int_0^t \exp\left(-\frac{\left(t-u\right)^2}{2\sigma_{tp}^2} + \frac{t_p}{\sigma_{tp}^2}\left(t-u\right)\right)\left[\frac{1}{C_K\sqrt{\pi\, u}} - \frac{3\, h^2}{2\sqrt{\pi}\left(C_K\sqrt{u}\right)^3} + ...\right] du, \end{aligned} \tag{B.2}$$

where

$$\begin{aligned} S^*\left(h \to 0, t\right) = \alpha_m \left[1 - \left(\frac{\alpha_m}{2} + a_r r_0\right) h + \right. \\ \left. \frac{1}{6}\left(\alpha_m^2 + 3\alpha_m a_r r_0 + 3 a_r^2 r_0\right) h^2 + ...\right], \quad \text{as } h \to 0. \end{aligned} \tag{B.3}$$

Hence, the integrals in Equations (5.24) and (5.25) are convergent as $h \to 0$. In our simulations, we use the expression given here for small film heights, since the direct evaluation of the integrals in Equations (5.24) and (5.25) numerically is difficult.

APPENDIX C

ANALYTICAL TEMPERATURE SOLUTION

In this appendix, we provide the details of the analytical temperature solution in the fluid-substrate domain specified in Section 5.4.2. Note that we change the notation here for the domain boundaries compared to Section 5.4, and we denote the bottom of the substrate as $y = 0$, the fluid-substrate interface as $y = a$, and the fluid-air interface as $y = b$. The temperature in the fluid, T_m, and the temperature in the substrate, T_s, satisfy the diffusion equation

$$\frac{\partial T_s}{\partial t} = \alpha_s \frac{\partial^2 T_s}{\partial y^2} \qquad \text{in} \quad 0 < y < a \tag{C.1}$$

$$\frac{\partial T_m}{\partial t} = \alpha_m \frac{\partial^2 T_m}{\partial y^2} + S\left(y, t\right) \qquad \text{in} \quad a < y < b \tag{C.2}$$

where

$$\alpha_s = \frac{k_s}{\rho_s C_{eff_s}}, \qquad \alpha_m = \frac{k_m}{\rho_m C_{eff_m}},$$

along with the boundary conditions

$$T_s = T_0 \qquad \text{at the bottom of the substrate } y = 0, \tag{C.3}$$

$$T_s = T_m \qquad \text{at the fluid-substrate interface } y = a, \tag{C.4}$$

$$k_1 \frac{\partial T_s}{\partial y} = k_2 \frac{\partial T_m}{\partial y} \qquad \text{at the fluid-substrate interface } y = a, \tag{C.5}$$

$$k_2 \frac{\partial T_m}{\partial y} = 0 \qquad \text{at the fluid-air interface } y = b. \tag{C.6}$$

The source term, $S(y, t)$ is given by Equation (5.17). The solution to the above equations can be written compactly in terms of Green's functions as given in

Equations (5.27) and (5.28), where

$$G_{i,j}\left(y,t;\xi,\tau\right)=\sum_{n=1}^{\infty}e^{-\beta_n^2(t-\tau)}\frac{1}{N_n}\frac{k_j}{\alpha_j}\psi_{i,n}\left(y\right)\psi_{j,n}\left(\xi\right), \tag{C.7}$$

$$N_n=\frac{k_1}{\alpha_1}\int_0^a\psi_{1,n}^2\mathrm{d}\xi+\frac{k_2}{\alpha_2}\int_a^b\psi_{2,n}^2\mathrm{d}\xi, \tag{C.8}$$

where $\psi_{i,n}$ and β_n are eigenfunctions and eigenvalues computed using separation of variables, and

$$\psi_{i,n}=A_{i,n}\Phi_{i,n}(y)+B_{i,n}\Theta_{i,n}(y)\quad\text{in}\quad y_i<y<y_{i+1}, \tag{C.9}$$

$$\Phi_{i,n}(y)=\sin\left(\frac{\beta_n}{\sqrt{\alpha_i}}y\right), \tag{C.10}$$

$$\Theta_{i,n}=\cos\left(\frac{\beta_n}{\sqrt{\alpha_i}}y\right), \tag{C.11}$$

where $y_0=0$, $y_1=a$ and $y_2=b$. In order to simplify the notation, let

$$\gamma=\frac{a\beta_n}{\sqrt{\alpha_1}},\quad \eta=\frac{b\beta_n}{\sqrt{\alpha_2}},\quad \mathcal{K}=\frac{k_1}{k_2}\sqrt{\frac{\alpha_2}{\alpha_1}}. \tag{C.12}$$

The eigenfunctions $\psi_{i,n}$ satisfy the boundary conditions in Equations (C.3)-(C.6). Hence, it follows

$$\psi_{1,n}=0\text{ at }y=0\qquad\rightarrow B_{1,n}=0,\ A_{1,n}=1\text{ without loss of generality,} \tag{C.13}$$

$$\psi_{1,n}=\psi_{2,n}\text{ at }y=a\qquad\rightarrow\sin\gamma=A_{2,n}\sin\left(\frac{a}{b}\eta\right)+B_{2,n}\cos\left(\frac{a}{b}\eta\right), \tag{C.14}$$

$$\frac{k_1}{k_2}\frac{\partial\psi_{1,n}}{\partial y}=\frac{\partial\psi_{2,n}}{\partial y}\text{ at }y=a\qquad\rightarrow\mathcal{K}\cos\gamma=A_{2,n}\cos\left(\frac{a}{b}\eta\right)-B_{2,n}\sin\left(\frac{a}{b}\eta\right), \tag{C.15}$$

$$\frac{\partial\psi_2}{\partial y}=0\text{ at }y=b\qquad\rightarrow A_{2,n}\cos\eta-B_{2,n}\sin\eta=0. \tag{C.16}$$

We can solve for the coefficients $A_{2,n}$ and $B_{2,n}$ using Equations (C.14) and (C.15)

$$A_{2,n}=\frac{1}{\Delta}\left[-\sin\gamma\sin\left(\frac{a}{b}\eta\right)-\mathcal{K}\cos\gamma\cos\left(\frac{a}{b}\eta\right)\right], \tag{C.17}$$

$$B_{2,n}=\frac{1}{\Delta}\left[\mathcal{K}\cos\gamma\sin\left(\frac{a}{b}\eta\right)-\sin\gamma\cos\left(\frac{a}{b}\eta\right)\right], \tag{C.18}$$

where

$$\Delta = -\sin^2\left(\frac{a}{b}\eta\right) - \cos^2\left(\frac{a}{b}\eta\right) = -1. \tag{C.19}$$

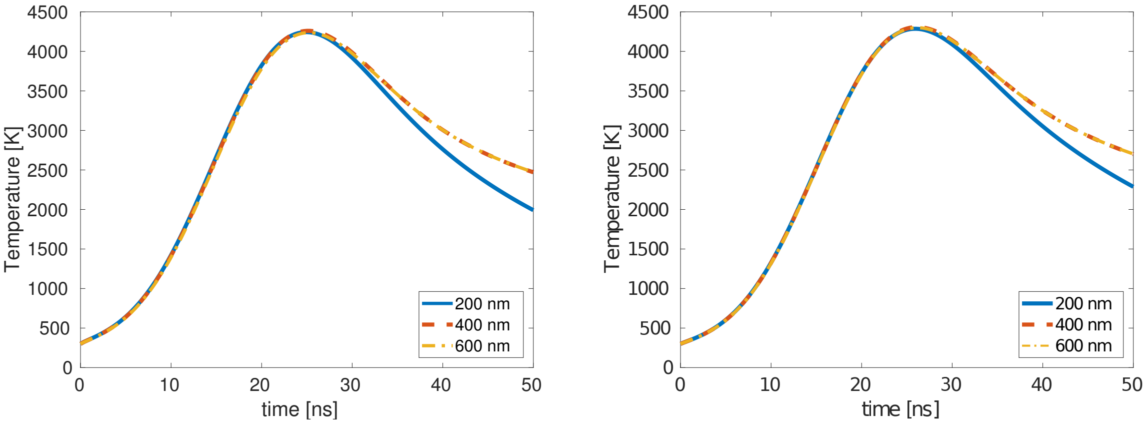
In order to have a solution, we require the determinant of the system of Equations (C.14)-(C.16) to be zero

$$\begin{vmatrix} \sin\gamma & -\sin\left(\frac{a}{b}\eta\right) & -\cos\left(\frac{a}{b}\eta\right) \\ \mathcal{K}\cos\gamma & -\cos\left(\frac{a}{b}\eta\right) & -\sin\left(\frac{a}{b}\eta\right) \\ 0 & -\cos\eta & -\sin\eta \end{vmatrix} = 0.$$

The equation above leads to the following equation for the eigenvalues, β_n

$$\tan\frac{a\beta_n}{\sqrt{\alpha_1}}\tan\left(\frac{\beta_n}{\sqrt{\alpha_2}}(b-a)\right) = \mathcal{K}, \tag{C.20}$$

which can be solved numerically.



(a)

(b)

Figure C.1 Convergence of the analytical solution with increased substrate depth for film thickness of (a) $h_0 = 10\,nm$ and (b) $h_0 = 20\,nm$.

Figure C.1 shows the average temperature in the metal as a function of time for different substrate sizes. We see that the solution converges as the substrate size increases. Hence, for a large enough substrate size, the solution is equivalent to the solution for a setup with semi-infinite substrate. This result is used in Section 5.4 to justify comparing the temperature obtained using the reduced model and semi-infinite substrate, with the one obtained by using the complete model and fixed substrate size.

C.1 Newton's Law of Cooling

Next, we show that replacing the continuity of temperature boundary condition at the fluid-substrate interface with Newton's law of cooling yields an equivalent solution as long as the heat transfer coefficient is large enough.

We replace the boundary condition (C.4) with

$$-\,k_1 \frac{\partial T_s}{\partial y} = \alpha\left(T_s - T_m\right) \qquad \text{at fluid-substrate interface } y = a. \tag{C.21}$$

Then the eigenfunctions of the same form as given in Equation (C.9) satisfy the boundary conditions (C.3), (C.5), (C.6) and (C.21). Hence, it follows

$$\psi_{1,n} = 0 \text{ at } y = 0 \rightarrow B_{1,n} = 1,\ A_{1,n} = 1 \text{ without loss of generality,} \tag{C.22}$$

$$-\,k_1 \frac{\partial \psi_{1,n}}{\partial y} = \alpha\left(\psi_{i,n} - \psi_{2,n}\right) \text{ at } y = a$$

$$\rightarrow -k_1 \cos\gamma = \alpha \sin\gamma - \alpha A_{2,n} \sin\left(\frac{a}{b}\eta\right) - \alpha B_{2,n} \cos\left(\frac{a}{b}\eta\right), \tag{C.23}$$

$$k_1 \frac{\partial \psi_{1,n}}{\partial y} = k_2 \frac{\partial \psi_{2,n}}{\partial y} \text{ at } y = a \rightarrow \mathcal{K} \cos\gamma = A_{2,n} \cos\left(\frac{a}{b}\eta\right) - B_{2,n} \sin\left(\frac{a}{b}\eta\right), \tag{C.24}$$

$$\frac{\partial \psi_2}{\partial y} = 0 \text{ at } y = b \quad \rightarrow A_{2,n} \cos\eta - B_{2,n} \sin\eta = 0. \tag{C.25}$$

From the Equations (C.23) and (C.24), we can solve for the coefficients $A_{2,n}$ and $B_{2,n}$

$$A_{2,n} = \frac{1}{\Delta}\left[\left(-H\cos\gamma - \sin\gamma\right)\sin\left(\frac{a}{b}\eta\right) - \mathcal{K}\cos\gamma\cos\left(\frac{a}{b}\eta\right)\right], \tag{C.26}$$

$$B_{2,n} = \frac{1}{\Delta}\left[\mathcal{K}\cos\gamma\sin\left(\frac{a}{b}\eta\right) + \left(-H\cos\gamma - \sin\gamma\right)\cos\left(\frac{a}{b}\eta\right)\right], \tag{C.27}$$

where

$$H = \frac{k_1 \beta_n}{\alpha\sqrt{\alpha_1}}. \tag{C.28}$$

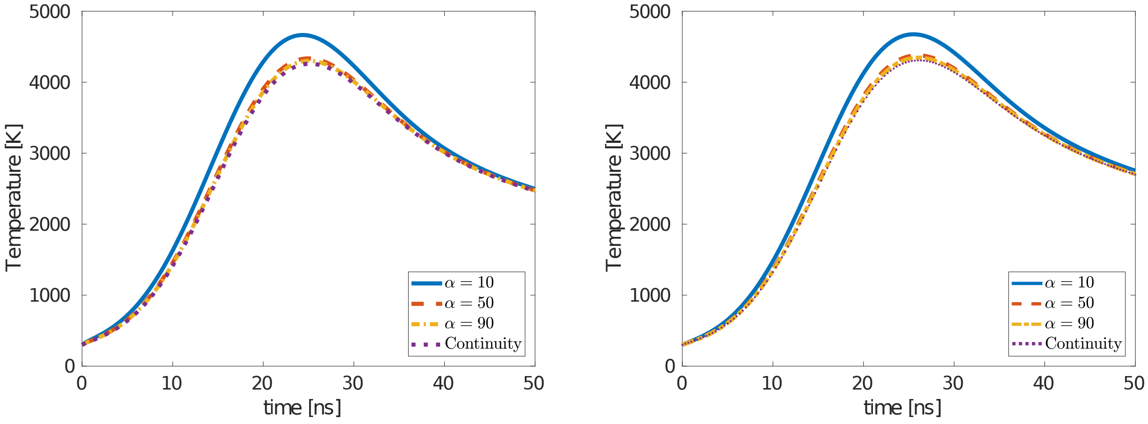
The condition for existence of a solution is vanishing determinant as follows

$$\begin{vmatrix} -H\cos\gamma-\sin\gamma & -\sin\left(\frac{a}{b}\eta\right) & -\cos\left(\frac{a}{b}\eta\right) \\ \mathcal{K}\cos\gamma & -\cos\left(\frac{a}{b}\eta\right) & -\sin\left(\frac{a}{b}\eta\right) \\ 0 & -\cos\eta & -\sin\eta \end{vmatrix} = 0.$$

The equation satisfied by the eigenvalues β_n is now

$$\left(H+\tan\frac{a\beta_n}{\sqrt{\alpha_1}}\right)\tan\left(\frac{\beta_n}{\sqrt{\alpha_2}}(b-a)\right)=\mathcal{K}, \tag{C.29}$$

which can be solved numerically.



(a) (b)

Figure C.2 Convergence of the analytical solution with increased α for film thickness of (a) $h_0 = 10\,nm$ and (b) $h_0 = 20\,nm$.

Figure C.2 shows the average temperature in the metal as a function of time for different α. We see that the solution with Newton's cooling law converges to the solution with continuity of temperature for large α. This result is used in Section 5.4 to justify comparing the reduced model that implements continuity of temperature with the complete model that uses Newton's law of cooling.

REFERENCES

[1] S. Popinet. An accurate adaptive solver for surface-tension-driven interfacial flows. *J. Comput. Phys.*, 228:5838–5866, 2009.

[2] C. Ma and D. Bothe. Direct numerical simulation of thermocapillary flow based on the Volume of Fluid method. *Int. J. Multiphase Flow*, 37:1045–1058, 2011.

[3] M. Herrmann, J. M. Lopez, P. Brady, and M. Raessi. Thermocapillary motion of deformable drops and bubbles. In *Proceedings of the Summer program*, pages 155–170, Stanford University, Center for Turbulence Research, 2008.

[4] S. Afkhami and M. Bussmann. Height functions for applying contact angles to 2D VOF simulations. *Int. J. Numer. Meth. Fluids*, 57:453–472, 2008.

[5] S. Karpitschka and H. Riegler. Quantitative experimental study on the transition between fast and delayed coalescence of sessile droplets with different but completely miscible liquids. *Langmuir*, 26:11823–11829, 2010.

[6] D. A. Drew. Mathematical Modeling of Two-Phase Flow. *Annu. Rev. Fluid Mech.*, 15:261, 1983.

[7] G. Tryggvason, A. Esmaeeli, J. Lu, and S. Biswas. Direct numerical simulations of gas/liquid multiphase flows. *Fluid Dyn. Res.*, 38:660, 2006.

[8] J. M. Hyman. Numerical methods for tracking interfaces. *Physica D*, 12:396, 1984.

[9] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114:146, 1994.

[10] M. Bonnet. Boundary integral equation methods for solids and fluids. *Meccanica*, 34:301, 1999.

[11] M. M. Francois and B. K. Swartz. Interface curvature via volume fractions, heights, and mean values on nonuniform rectangular grids. *J. Comput. Phys.*, 229:527–540, 2010.

[12] J. López and J. Hernández. On reducing interface curvature computation errors in the height function technique. *J. Comput. Phys.*, 229:4855–4868, 2010.

[13] R. B. DeBar. Fundamentals of the KRAKEN code. *Tech. rep., California Univ., Livermore (USA). Lawrence Livermore Lab.*, 1974.

[14] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.*, 39:201, 1981.

[15] W. F. Noh and P. Woodward. SLIC (simple line interface calculation). In *Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics*, page 330, Twente University, Enschede, 1976. Springer.

[16] D. L. Youngs. Time-dependent multi-material flow with large fluid distortion. In K. W. Morton and M. J. Baines, editors, *Numerical methods for fluid dynamics*, page 273. New York, NY: Academic Press, 1982.

[17] W. J. Rider and D. B. Kothe. Reconstructing Volume Tracking. *J. Comput. Phys.*, 141:112, 1998.

[18] J. López, J. Hernández, P. Gómez, and F. Faura. An improved PLIC-VOF method for tracking thin fluid structures in incompressible two-phase flows. *J. Comput. Phys.*, 208:51, 2005.

[19] D. Gerlach, G. Tomar, G. Biswas, and F. Durst. Comparison of volume-of-fluid methods for surface tension-dominant two-phase flows. *Int. J. Heat Mass Transfer*, 49:740, 2006.

[20] E. G. Puckett, A. S. Almgren, J. B. Bell, D. L. Marcus, and W. J. Rider. A high-order projection method for tracking fluid interfaces in variable density incompressible flows. *J. Comput. Phys.*, 130:269, 1997.

[21] Y. Renardy and M. Renardy. PROST: a parabolic reconstruction of surface tension for the volume-of-fluid method. *J. Comput. Phys.*, 183:400, 2002.

[22] V. R. Gopala and B. G. M. van Wachem. Volume of fluid methods for immiscible-fluid and free-surface flows. *Chem. Eng. J.*, 141:204, 2008.

[23] J. U. Brackbill, D. B. Kothe, and C. Zemach. A continuum method for modeling surface tension. *J. Comput. Phys.*, 100:335–354, 1992.

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

[25] S. J. Cummins, M. M. Francois, and D. B. Kothe. Estimating curvature from volume fractions. *Comput. Struct.*, 83:425–434, 2005.

[26] M. Malik, E. S.-C. Fan, and M. Bussmann. Adaptive VOF with curvature-based refinement. *Int. J. Numer. Methods Fluids*, 55:693, 2007.

[27] J. López, C. Zanzi, P. Gómez, R. Zamora, F. Faura, and J. Hernández. An improved height function technique for computing interface curvature from volume fractions. *Comput. Methods Appl. Mech. Eng.*, 198:2555, 2009.

[28] L. E. Scriven and C. V. Sternling. The Marangoni effects. *Nature*, 187:186–188, 1960.

[29] R. H. Farahi, A. Passian, T. L. Ferrell, and T. Thundat. Microfluidic manipulation via Marangoni forces. *Appl. Phys. Lett.*, 85:4237–4239, 2004.

[30] A. Kundan, J. L. Plawsky, P. C. Wayner, D. F. Chao, R. J. Sicker, B. J. Motil, T. Lorik, L. Chestney, J. Eustace, and J. Zoldak. Thermocapillary phenomena and performance limitations of a wickless heat pipe in microgravity. *Phys. Rev. Lett.*, 114:146105–5, 2015.

[31] R. S. Subramanian, R. Balasubramaniam, and G. Wozniak. Fluid mechanics of bubbles and drops. In R. Monti, editor, *Physics of Fluids in Microgravity*, pages 149–177. Taylor and Francis, 2002.

[32] J. Trice, D. Thomas, C. Favazza, R. Sureshkumar, and R. Kalyanaraman. Novel self-organization mechanism in ultrathin liquid films: Theory and experiment. *Phys. Rev. Lett.*, 101:017802–4, 2008.

[33] N. Dong and L. Kondic. Instability of nanometric fluid films on a thermally conductive substrate. *Phys. Rev. Fluids*, 1:063901–16, 2016.

[34] S. Davis. Thermocapillary instabilities. *Ann. Rev. Fluid Mech.*, 19:403–435, 1987.

[35] R.V. Craster and O.K. Matar. Dynamics and stability of thin liquid films. *Rev. Mod. Phys.*, 81:1131–1198, 2009.

[36] M. Muradoglu and G. Tryggvason. Simulations of soluble surfactants in 3D multiphase flow. *J. Comput. Phys.*, 274:737–757, 2014.

[37] J.-J. Xu, Z. Li, J. Lowengrub, and H. Zhao. A level-set method for interfacial flows with surfactant. *J. Comput. Phys.*, 212:590–616, 2006.

[38] K. E. Teigen, P. Song, J. Lowengrub, and A. Voigt. A diffuse-interface method for two-phase flows with soluble surfactants. *J. Comput. Phys.*, 230:375 – 393, 2011.

[39] F. Blanchette, L. Messio, and J. W. M. Bush. The influence of surface tension gradients on drop coalescence. *Phys. Fluids*, 21:072107–10, 2009.

[40] F. Blanchette and A. M. Shapiro. Drops settling in sharp stratification with and without Marangoni effects. *Phys. Fluids*, 24:042104–17, 2012.

[41] M.-C. Lai, Y.-H. Tseng, and H. Huang. An immersed boundary method for interfacial flows with insoluble surfactant. *J. Comput. Phys.*, 227:7279–7293, 2008.

[42] M. R. Booty and M. Siegel. A hybrid numerical method for interfacial fluid flow with soluble surfactant. *J. Comput. Phys.*, 229:3864–3883, 2010.

[43] F. S. Schranner and N. A. Adams. A conservative interface-interaction model with insoluble surfactant. *J. Comput. Phys.*, 327:653–677, 2016.

[44] M. A. Drumright-Clarke and Y. Renardy. The effect of insoluble surfactant at dilute concentration on drop breakup under shear with inertia. *Phys. Fluids*, 16:14–21, 2004.

[45] A. J. James and J. Lowengrub. A surfactant-conserving volume-of-fluid method for interfacial flows with insoluble surfactant. *J. Comput. Phys.*, 201:685–722, 2004.

[46] A. Alexeev, T. Gambaryan-Roisman, and P. Stephan. Marangoni convection and heat transfer in thin liquid films on heated walls with topography: Experiments and numerical study. *Phys. Fluids*, 17:062106–13, 2005.

[47] S. Popinet. Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *J. Comput. Phys.*, 190:572–600, 2003.

[48] S. Afkhami and M. Bussmann. Height functions for applying contact angles to 3D VOF simulations. *Int. J. Numer. Meth. Fluids*, 61:827, 2009.

[49] H. A. Atwater and A. Polman. Plasmonics for improved photovoltaic devices. *Nat. Mater.*, 9:9, 2010.

[50] Z. Nie, A. Petukhova, and E. Kumacheva. Properties and emerging applications of self-assembled structures made from inorganic nanoparticles. *Nat. Nanotechnol.*, 5:15, 2010.

[51] A. I. Kuznetsov, A. B. Evlyukhin, M. R. Gonçalves, C. Reinhardt, A. Koroleva, M. L. Arnedillo, R. Kiyan, O. Marti, and B. N. Chichkov. Laser fabrication of large-scale nanoparticle arrays for sensing applications. *ACS Nano*, 5:48439, 2011.

[52] P. D. Rack, Y. F. Guan, J. D. Fowlkes, A. V. Melechko, and M. L. Simpson. Pulsed laser dewetting of patterned thin metal films: A means of directed assembly. *Appl. Phys. Lett.*, 92:223108, 2008.

[53] Y. Wu, J. D. Fowlkes, P. D. Rack, J. A. Diez, and L. Kondic. On the breakup of patterned nanoscale copper rings into droplets via pulsed-laser-induced dewetting: Competing liquid-phase instability and transport mechanisms. *Langmuir*, 26:11972, 2010.

[54] J. D. Fowlkes, L. Kondic, J. Diez, and P. D. Rack. Self-assembly versus directed assembly of nanoparticles via pulsed laser induced dewetting of patterned metal films. *Nano Lett.*, 11:2478, 2011.

[55] J. D. Fowlkes, N. A. Roberts, Y. Wu, J. A. Diez, A. G. González, C. Hartnett, K. Mahady, S. Afkhami, L. Kondic, and P. D. Rack. Hierarchical nanoparticle ensembles synthesized by liquid phase directed self-assembly. *Nano Lett.*, 14:774, 2014.

[56] N. A. Roberts, J. D. Fowlkes, K. Mahady, S. Afkhami, L. Kondic, and P. D. Rack. Directed assembly of one- and two-dimensional nanoparticle arrays from pulsed laser induced dewetting of square waveforms. *ACS Appl. Mater. Interfaces*, 5:4450, 2013.

[57] C. Favazza, R. Kalyanaraman, and R. Sureshkumar. Robust nanopatterning by laser-induced dewetting of metal nanofilms. *Nanotechnology*, 17:4229, 2006.

[58] J. Trice, D. Thomas, C. Favazza, R. Sureshkumar, and R. Kalyanaraman. Pulsed-laser-induced dewetting in nanoscopic metal films: Theory and experiments. *Phys. Rev. B*, 75:235439, 2007.

[59] H. Krishna, R. Sachan, J. Strader, C. Favazza, M. Khenner, and R. Kalyanaraman. Thickness-dependent spontaneous dewetting morphology of ultrathin Ag films. *Nanotechnology*, 21:155601, 2010.

[60] J. T. McKeown, N. A. Roberts, J. D. Fowlkes, Y. Wu, T. LaGrange, B. W. Reed, G. H. Campbell, and P. D. Rack. Real-time observation of nanosecond liquid-phase assembly of nickel nanoparticles via pulsed-laser heating. *Langmuir*, 28:17168, 2012.

[61] A. G. Gonzalez, J. A. Diez, Y. Wu, J. D. Fowlkes, P. D. Rack, and L. Kondic. Instability of liquid Cu films on a SiO2 substrate. *Langmuir*, 29:9378, 2013.

[62] S. Yadavali, M. Khenner, and R. Kalyanaraman. Pulsed laser dewetting of Au films: Experiments and modeling of nanoscale behavior. *J. Mater. Res.*, 28:1715, 2013.

[63] Y. Wu, N. Dong, S. Fu, J. D. Fowlkes, L. Kondic, M. A. Vincenti, D. de Ceglia, and P. D. Rack. Directed liquid phase assembly of highly ordered metallic nanoparticle arrays. *ACS Appl. Mater. Interfaces*, 6:5835, 2014.

[64] C. Favazza, J. Trice, H. Krishna, and R. Kalyanaraman. Laser induced short- and long-range orderings of Co nanoparticles on SiO2. *Appl. Phys. Lett.*, 88:153118, 2006.

[65] H. Krishna, N. Shirato, C. Favazza, and R. Kalyanaraman. Energy driven self-organization in nanoscale metallic liquid films. *Phys. Chem. Chem. Phys.*, 11:8136, 2009.

[66] L. Kondic, J. Diez, P. Rack, Y. Guan, and J. Fowlkes. Nanoparticle assembly via the dewetting of patterned thin metal lines: Understanding the instability mechanism. *Phys. Rev. E*, 79:026302, 2009.

[67] A. Atena and M. Khenner. Thermocapillary effects in driven dewetting and self assembly of pulsed-laser-irradiated metallic films. *Phys. Rev. B*, 80:075402, 2009.

[68] M. Khenner, S. Yadavali, and R. Kalyanaraman. Controlling nanoparticles formation in molten metallic bilayers by pulsed-laser interference heating. *Math. Model. Nat. Phenom.*, 7:20, 2012.

[69] O. S. Heavens. *Optical properties of thin solid films.* New York, NY: Dover Publications, 1991.

[70] J. W. Strutt and L. Rayleigh. On the instability of jets. *Proc. London Math. Soc*, 10:4, 1878.

[71] K. Mahady, S. Afkhami, and L. Kondic. On the influence of initial geometry on the evolution of fluid filaments. *Phys. Fluids*, 27:092104, 2015.

[72] S. Afkhami and L. Kondic. On the dewetting of liquefied metal nanostructures. *J. Eng. Math.*, 94:5, 2014.

[73] L. D. Landau and E. M. Lifshitz. *Fluid mechanics, 2nd*, volume 6. Oxford, UK: Pergamon Press, 1987.

[74] V. G. Levich and V. S. Krylov. Surface-tension-driven phenomena. *Annu. Rev. Fluid Mech.*, 1:293–316, 1969.

[75] A. J. Chorin. On the convergence of discrete approximations to the Navier-Stokes equations. *Math. Comput.*, 23:341, 1969.

[76] J. B. Bell, P. Colella, and H. M. Glaz. A second-order projection method for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 85:257, 1989.

[77] A. Oron, S. H. Davis, and S. G. Bankoff. Long-scale evolution of thin liquid films. *Rev. Mod. Phys.*, 69:931, 1997.

[78] S. Popinet. Gerris flow solver. `http://gfs.sourceforge.net/wiki/index.php`, 1999. Accessed: 2013-12-06.

[79] E. Aulisa, S. Manservisi, R. Scardovelli, and S. Zaleski. Interface reconstruction with least-squares fit and split advection in three-dimensional Cartesian geometry. *J. Comput. Phys.*, 225:2301–2319, 2007.

[80] G. Wozniak, J. Siekmann, and J. Srulijes. Thermocapillary bubble and drop dynamics under reduced gravity-survey and prospects. *Zeitschrift für Flugwissenschaften und Weltraumforschung*, 12:137–144, 1988.

[81] S. Nas and G. Tryggvason. Thermocapillary interaction of two bubbles or drops. *Int. J. Multiphase Flow*, 29:1117–1135, 2003.

[82] N. O. Young, J. S. Goldstein, and M. J. Block. The motion of bubbles in a vertical temperature gradient. *J. Fluid Mech.*, 6:350–356, 1959.

[83] S. Karpitschka and H. Riegler. Noncoalescence of sessile drops from different but miscible liquids: Hydrodynamic analysis of the twin drop contour as a self-stabilizing traveling wave. *Phys. Rev. Lett.*, 109:066103–5, 2012.

[84] S. Karpitschka and H. Riegler. Sharp transition between coalescence and non-coalescence of sessile drops. *J. Fluid Mech.*, 743:R1, 2014.

[85] S. Afkhami, S. Zaleski, and M. Bussmann. A mesh-dependent model for applying dynamic contact angles to VOF simulations. *J. Comput. Phys.*, 228:5370–5389, 2009.

[86] L. Rayleigh. On the instability of jets. *Proc. London Math. Soc.*, 1:4–13, 1878.

[87] J. Eggers. Nonlinear dynamics and breakup of free-surface flows. *Rev. Mod. Phys.*, 69:865–930, 1997.

[88] A. Meyer. Self-diffusion in liquid copper as seen by quasielastic neutron scattering. *Phys. Rev. B*, 81:012102, 2010.

[89] J. A. Diez and L. Kondic. On the breakup of fluid films of finite and infinite extent. *Phys. Fluids*, 19:072107, 2007.

[90] K. Mahady, S. Afkhami, and L. Kondic. A volume of fluid method for simulating fluid/fluid interfaces in contact with solid boundaries. *J. Comput. Phys.*, 294:243, 2015.

[91] M. Necati Özışık. *Boundary value problems of heat conduction.* Mineola, NY: Dover Publications, 1989.

[92] W. F. Gale and T. C. Totemeier. *Smithells metals reference book.* Butterworth-Heinemann, 2003.

[93] J. A. Diez, A. G. Gonzalez, and L. Kondic. On the breakup of fluid rivulets. *Phys. Fluids*, 21, 2009.

[94] M. J. Rosen and J. T. Kunjappu. *Surfactants and interfacial phenomena.* John Wiley & Sons, 2012.