CAPILLARY FLOW OF LIQUID METALS IN BRAZING

By

MOHAMMAD DEHSARA

A dissertation submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
School of Mechanical and Materials Engineering

MAY 2017
To the Faculty of Washington State University:

The members of the Committee appointed to examine the dissertation of MOHAMMAD DEHSARA find it satisfactory and recommend that it be accepted.

______________________________
Sinisa Dj. Mesarovic, Ph.D., Chair

______________________________
Jin Liu, Ph.D.

______________________________
Alexander Panchenko, Ph.D.
ACKNOWLEDGMENT

First and foremost, I want to thank my advisor, Prof. Sinisa Dj. Mesarovic, for his continuously support and guidance during my Ph.D. study, for his patience, resourceful insight, and immense knowledge. I am greatly indebted to him for the brilliant interdisciplinary professional skills he has taught me. His problem solving, creative approach and critical thinking to tackle difficult challenges will always inspire me. It has been an honor to be his Ph.D. student, and I hope to continue to work with his noble thoughts.

I would like to thank to my thesis committee: Prof. Jin Liu, and Prof. Alexander Panchenko for kindly being on my thesis committee, for their insightful comments, and also for their constructive questions which incented me to widen my research from various perspectives. I would particularly like to acknowledge Prof. Dusan P. Sekulic, Prof. Mikhail Krivilev, and Dr. Hai Fu for providing the experimental results and making significant contributions to this cooperation research funded by NSF. I am especially grateful for their time and many insightful discussions with them that have benefited me throughout this research.

I wish to acknowledge the financial support of the US National Science Foundation (NSF) under grant number CBET # 1234581, and the Mechanical and Materials Engineering (MME) department during my doctoral studies. The numerical simulations presented in current study have been implemented using the high-performance computing facilities at the MME department.

Finally, I will forever be thankful to my family: my parents, my brother and sister for supporting me spiritually throughout the past few years and my life in general. I also thank my friends for being with me and creating wonderful times during my Ph.D. program.
Capillary flow is driven or controlled by capillary forces, exerted at the triple line where the fluid phases meet the solid boundary. Phase field (PF) models naturally accommodate diffusive triple line motion with variable contact angle, thus allowing for the no-slip boundary condition without the stress singularities. Moreover, they are uniquely suited for modeling of topological discontinuities which often arise during capillary flows. In this study, we consider diffusive triple line motion within two PF models: the compositionally compressible (CC) and the incompressible (IC) models. We derive the IC model as a systematic approximation to the CC model, based on a suitable choice of continuum velocity field. The CC model, applied to the fluids of dissimilar mass densities, exhibits a computational instability at the triple line. The IC model perfectly represents the analytic equilibria. We develop the parameter identification procedure and show that the triple line kinetics can be well represented by the IC model’s diffusive boundary condition.

The IC model is first tested by benchmarking the phase-field and experimental kinetics of water, and silicone oil spreading over the glass plates in which two systems do not interact with
the substrate. Then, two high-temperature physical settings involving spreading of the molten Al-Si alloy: one over a rough wetting substrate, the other over a non-wetting substrate are modeled in a T-joint structure which is a typical geometric configuration for many brazing and soldering applications. Surface roughness directly influences the spreading of the molten metal by causing break-ups of the liquid film and trapping the liquid away from the joint. In the early stages of capillary flow over non-wetting surface, the melting and flow are concurrent, so that the kinetics of wetting is strongly affected by the variations in effective viscosity of the partially molten metal. We define adequate time-dependent functions for the variations of Al-Si alloy viscosity and triple line mobility to describe the wetting kinetics.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGEMENT</td>
<td>iii</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>ix</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>x</td>
</tr>
<tr>
<td>CHAPTER</td>
<td></td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2. PROBLEM FORMULATION</td>
<td>8</td>
</tr>
<tr>
<td>2.1 Compositionally compressible (CC) flow</td>
<td>8</td>
</tr>
<tr>
<td>2.2 Incompressible flow as an approximation to the CC flow</td>
<td>17</td>
</tr>
<tr>
<td>2.3 Physical and computational parameters</td>
<td>21</td>
</tr>
<tr>
<td>3. COMPUTATIONAL RESULTS AND DISCUSSION</td>
<td>23</td>
</tr>
<tr>
<td>3.1 Numerical method, boundary and initial conditions</td>
<td>23</td>
</tr>
<tr>
<td>3.2 Equilibrium</td>
<td>25</td>
</tr>
<tr>
<td>3.3 The effects of the phase field mobility</td>
<td>26</td>
</tr>
<tr>
<td>3.4 Computational and experimental kinetics</td>
<td>27</td>
</tr>
<tr>
<td>3.4.1 Capillary flow of the low-temperature non-metallic liquid systems</td>
<td>28</td>
</tr>
<tr>
<td>3.4.2 Capillary flow of the high-temperature liquid metal systems</td>
<td>32</td>
</tr>
<tr>
<td>3.4.2.1 Spreading of the molten Trillium metal over the rough wetting surface</td>
<td>33</td>
</tr>
</tbody>
</table>
3.4.2.2 Spreading of the molten Trillium metal over the non-wetting surface

4. SUMMARY AND CONCLUSIONS

4.1 Future work

APPENDIX

A. PHASE FIELD DIFFUSION AS A CONSEQUENCE OF
BARYCENTRIC

B. ENERGY BALANCE AND THE PVP FOR THE CC MODEL

C. RELATIONSHIP BETWEEN THE TWO DEFINITIONS OF
DIFFUSIONAL FLUX

D. ENERGY BALANCE AND THE PVP FOR THE IC MODEL

E. USER’S MANUAL

1. Introduction

2. Implementation with COMSOL Multiphysics
   2.1 Parameters and variables definitions
   2.2 Geometry definition
   2.3 Meshing
   2.4 Problem specification
      2.4.1 General Form PDE Interface
      2.4.2 Weak Form PDE Interface
      2.4.3 Specifying and Interpreting Boundary Conditions
   2.5 Solver definition
2.6 Post processing and visualization…………………………58

3. Implementation with MATLAB……………………………59

3.1 The General Utility Functionality…………………………59

3.1.1 Managing The COMSOL model object………………59

3.1.2 Activating The Progress Bar…………………………60

3.2 The Geometry Sequence Syntax…………………………61

3.3 The Meshing Sequence Syntax……………………………61

3.4 The Physics Interface Syntax……………………………62

3.5 The Material Syntax……………………………………63

3.6 Adding Global Equations………………………………64

3.7 The Study Node…………………………………………65

3.7.1 The Solver Sequence Syntax…………………………66

3.7.2 Run the Solver Sequence……………………………67

3.8 Plot While Solving………………………………………68

3.8.1 The Plot Group Syntax……………………………68

REFERENCES………………………………………………………70
LIST OF TABLES

3.1 Physical and numerical parameters (Fu 2016 & Dehsara et al. 2017) ....................... 25
LIST OF FIGURES

1.1. The characteristic video frames of a clad melting sequence. The substrate is AA 3003 covered by the AA4343 clad and a thin layer of potassium fluoro-aluminate. (a) 573°C, the initial state of the molten flux at the onset of Al-Si melting. (b) 595.2°C, the liquid Al-Si layer formed and the onset of capillary flow. (c) 598.4°C, the formation of a rough surface due to separation of the liquid and residue in situ (alpha phase, Al-Si, lumps). (d) 600°C, the onset of solidification……………………………………………………………………6

1.2. Schematic of the capillary evolution after melting over non-wetting horizontal substrate and wetting vertical substrate. The initial configuration is a solid block. The arrows show directions of motion of the two triple lines: The receding one and the advancing one…………………………………………………………………………………………………6

2.1. Free energy density of a homogeneous fluid as a function of the phase variable. The relevant parameter is $\Delta f$, see discussion in Section 2.3…………………………………11

2.2. Instability of the computational CC model with ratio of mass densities of liquid and gas of 811. Capillary rise between parallel walls. (a) Initial contours of the phase field. (b) Contours of the velocity divergence. (c) Magnified instability region circled in (a). (d) Line contours corresponding to (c)…………………………………………………………………………………..14

2.3. Ellipsoidal interface evolves into a spherical one and consequently changes its volume. If the rest of the fluid is incompressible, such volume change requires large scale motion of the surrounding fluid (indicated by arrows) and shrinking/expansion of the computational cell boundary…………………………………………………………………………………………15
3.1. Numerical and analytical solutions of the capillary rise between parallel plates in equilibrium condition. (a) Phase field contours. (b) Meniscus shape defined as the line $c = 1/2$. (c) Pressure variation across the interface in the centerline $x = 0$ ..................................26

3.2. The effect of the bulk mobility on the kinetics of the interface in capillary rise between two parallel plates. $t_\mu = 3.1 \times 10^{-5} s$; $t_b = 2.0 \times 10^{-2} s$ ..................................................27

3.3. Schematic of the T-joint configuration in experiment setup for low-temperature case. (a) Before the wetting process. (b) After the wetting process ..............................................28

3.4. The evolution of the interface shapes for the T-joint formation at times $t = 0$ s, 1 s, and 30 s. Contour plots of the phase field (left) and the corresponding adaptive meshes (right) ..................................................29

3.5. Triple line kinetics in log-log scale for (a) Water, and (b) Silicone oil. Comparison between numerical results and experiments. The triple line mobility $b$ is fitted to the initial slope .................................................................30

3.6. The formation of the wave in the water pool following the initial surge of the triple line (a) Experimental (19ms); (b) Computational (20ms) ..............................................30

3.7. Computational and analytical equilibrium profiles of menisci. The parameters are listed in Table 3.1 .................................................................31

3.8. Dynamic contact angle as function of time: Numerical prediction vs. experimental data .................................................................32

3.9. Schematic of the experiment setups: (a) 400-micron thickness Trillium filler on the wetting aluminum substrate. (b) 800-micron thickness Trillium filler on the non-wetting alumina substrate .................................................................33
3.10. Snapshots of the Trillium profiles on the wetting aluminum surface captured from experiment (a) Initial condition. (b) Equilibrium condition……………………………………34

3.11. The evolution of the interface for the molten Trillium spreading over the wetting aluminum surface at times t=1s, 20s and 120s. Contours of the phase field variable (left) and the corresponding adaptive meshes (right)………………………………………………………35

3.12. Numerical, and experimental data of the molten Trillium spreading over the aluminum surface. (a) Triple line kinetics. (b) Meniscus profile………………………………………………………36

3.13. Snapshots of the Trillium profiles on the non-wetting alumina surface captured from experiment. (a) Initial condition, (b) Equilibrium condition……………………………………37

3.14. Numerical, and experimental data of the Trillium spreading over the non-wetting surface. (a) Initial phase field contour. (b) Meniscus profiles in equilibrium condition. (c) Triple line kinetics……………………………………………………………………………………………………39
Dedication

To my mother Safoura (Simin) Mohitmafi, my father, Fereidoun Dehsara, my sister, Bahareh Dehsara, and my brother, Reza (Bahador) Dehsara for supporting and encouraging me over the years constantly and unconditionally. Their tremendous love and support has given me the courage to follow my dreams.
CHAPTER 1
INTRODUCTION

Flows controlled by capillary forces have long been a subject of interest (de Gennes 1985, de Gennes et al. 2004, Bonn et al. 2009), as a fundamental physical problem, as well as related to industrial and natural processes, including materials joining (Liu & Sekulic 2011, Liu et al. 2011, Eustathopoulos et al. 2013), painting (Patton 1979), biological phenomena such as superhydrophilic leaves in plant biology (Koch et al. 2008), geology such as molten lava flows (Lezzi & Ventura 2005). Among these applications, high-temperature wetting of solid metallic or ceramics substrates by liquid metal alloys is of importance in many technological processes including brazing (Sekulic 2013), soldering (Vianco 1999), and coating (Marrion 2004, Bose 2011). In applications like brazing, wetting kinetics has a strong influence on the final microstructure of the material within the joint domain and consequently, on the strength and durability of the bond between joined materials (Pan et al. 2002, Qian et al. 2003, Saiz et al. 2003). Thus, understanding of the kinetics of the wetting process through reliable predictive models is not only a non-trivial scientific problem, but also has a wide technological importance.

Wetting/spreading of liquid metals at high temperatures is typically accompanied by cross-diffusion, interfacial chemical reactions, ridging, dissolution of the solid substrate, etc., (Eustathopoulos et al. 1999, Saiz & Tomisa 2005, Kumar & Narayan Prabhu 2007). Under such conditions, a movement of the triple line, the locus of points where the fluid and gas phases intersect the solid boundary, is strongly affected by formation of new material compounds, composition variations, and diffusion induced phase formations (Saiz et al. 2000, Saiz & Tomsia
2004), so that uncovering the exact spreading mechanism that controls the wetting kinetics is difficult.

Sharp interface models applied to capillary flows face two key problems. The first is the singularity of the solutions to fluid dynamics equations with no-slip boundary condition in the vicinity of the triple line (Durbin 1988, Dussan et al. 1991). The remedies for this problem have been attempted by allowing a slip over a characteristic length within a sharp interface model (Blake & Shikhmurzaev 2002), and even within the phase field models (Qian et al. 2006), but suffered from physical ambiguities and parameter uncertainties. The second problem is computational treatment of topological discontinuities which typically arise in capillary flows. Moving boundaries and topological discontinuities (breakup and coalescence of domains) have been an outstanding computational problem for decades. Two solutions have emerged over the years: smoothed particle hydrodynamics (Monaghan 2005) and the phase field (or diffuse interface) models. The latter treat the multiple materials in multiple domains as one material with phases distinguished by the value of the phase field variable. The interface energies are represented as penalties imposed on the phase field gradient. The interfaces are moved with the flow but locally adjust by diffusive mechanism.

The earliest recorded suggestion that an interface between two fluids is diffuse is probably the one by Poisson (1831). Gibbs (1878) considered a similar idea, and van der Waals (1893) proposed the first mathematical model. Following the formulations of Cahn & Hilliard (1958) and Cahn (1961), and the advances in computing power, the diffuse interface or phase field formulation has been widely used for modeling moving interfaces in both solids and fluids. Both types of applications have been recently reviewed; see Chen (2002) for solids, and Anderson et al. (1998) and Kim (2012) for fluids. In most applications, the phase field does not
model the actual interface dimensions. Instead, the phase field model is viewed as a mathematical tool for computational treatment of changing topology. Its validity is based on its adherence to the corresponding sharp interface model and mapping of the phase field parameters to physical parameters (Wang & Wang 2007, Sibley et al. 2013).

The phase field model allows for distinction between convective and diffusive motion of the fluid. While the convective motion is still subject to the no-slip boundary condition, the contact line moves by surface diffusion mechanism, driven by fluid dynamics forces as well as the gradient in the diffusion potential. The mechanism is supported by molecular dynamics studies (Thompson & Robins 1990, Koplik & Banavar 1998) and molecular kinetic theory (Blake & Haynes 1969, Seveno et al. 2009). Building on the Seppecher’s (1996) analysis, the mathematical formulation of the problem has been developed by Jacqmin (2000) and applied, using finite difference method, to analyze contact line dynamics in capillary flows. The Jacqmin’s boundary condition was implemented into the finite element framework for incompressible (IC) flow by Yue & Feng (2011a, b) and Fu et al. (2016). The current study is to our knowledge the first finite element implementation of the Jacqmin’s diffusive boundary condition for the compositionally compressible\(^1\) (CC) model.

While the thermodynamic consistency of the CC model (Lowengrub & Truskinovsky 1998) weights in its favor, the computational instabilities which arise when fluids have very different mass densities present an obstacle for its application. On the other hand, the IC model is much simpler and computationally stable and efficient, albeit based on ad hoc derivations (e.g., Ding et al. 2007). Nevertheless, intuition dictates that the effect of compressibility in the thin interface layer (which is the main difference between the two models) is probably negligible. In

\(^{1}\) We believe that this name is more descriptive than quasi-incompressible used by Lowengrub & Truskinovsky (1998).
In this work, we establish the mathematical connection between the two models and show that the IC model is the approximation to the thermodynamically rigorous CC model. This enables the understanding of the nature of computational instabilities that arise in the CC flow.

The interface between immiscible fluids is represented by a conserved phase field leading to the 4th order partial differential equation, known as the Cahn-Hilliard (CH) equation (Cahn & Hilliard 1958). The problems in this class have been addressed in the past, with a variety of numerical methods, including the finite difference method (FDM) (Cahn and Kobayashi 1995, Jacqmin 2000), spectral, pseudo-spectral, and related methods (Liu & Shen 2002, Chen & Shen 2013). The drawbacks of such methods are seen in the difficulties in modeling irregular domains, nonlinear and history-dependent problems. The finite element (FE) formulations have been developed within the last decade. The mixed, discontinuous ($C^0$) Galerkin formulation has been first applied to a single, uncoupled Cahn-Hilliard equation (Kay & Welford 2006), while the Galerkin continuous FE formulation ($C^1$) for coupled Cahn-Hilliard/solid mechanics problem has been developed (Asle Zaeem & Mesarovic 2009, 2010a, b). The coupling with the Navier-Stokes equations ($C^0$) for the bulk multiphase flow with no solid boundaries has been implemented by Feng (2006), Hua et al. (2011) and Guo et al. (2014). The triple line motion with equilibrium contact angle (instantaneous relaxation) has often been considered by Villanueva & Amberg (2006), Ding & Spelt (2008), Gao & Feng (2009), Yue et al. (2010). The dynamically variable contact angle with Jacqmin’s diffusive boundary condition has been implemented by Yue & Feng (2011a, b) for the IC model.

In this study, the IC model is tested successfully by benchmarking the phase-field and experimental kinetics of water, silicone oil and Al-Si alloy spreading. First two systems do not interact with the substrate (glass plates). However, the Al-Si clad interacts with the Al substrate
through the Si diffusion process thus creating a nonuniform composition. The sequence of events during clad melting is shown in Figure 1.1. The hypoeutectic Al-Si alloy clad melts within a range of temperatures only partially, with the liquid portion flowing into the joint, while the rest being a solid solution with a high Al content (Gao et al. 2002, 2004, Zhao & Sekulic 2008). We find that during the brazing process up to 60% of the available Al-Si alloy ends up in the wedge-tee joint, and forms the capillary driven meniscus profile, while the rest of the melted alloy remains isolated from the meniscus between the solid lumps on the horizontal mating surface. This roughness on the bottom substrate is created simultaneously with the onset of melting. We investigate the effect of so formed surface roughness on the wetting process.

Further, we study a high-temperature capillarity driven flow in a system consisting of horizontal non-wetting and vertical wetting substrates in wedge-tee configuration. Since the capillary flow starts as soon as the first liquid appears, the initial flow is concurrent with the melting process. The spreading of the molten metal in both cases is considered as isothermal owing to a near eutectic Al-Si composition. In the non-wetting case, the vertical wetting surface used is AA3003, while the horizontal non-wetting surface is made of alumina (Al₂O₃). The contact angle between the horizontal surface and the triple line is greater than 90° during the onset and subsequent capillary rise. The molten clad metal spreads over the vertical sheet along the decreasing contact angle, hence recedes from the horizontal substrate, shown in Figure 1.2.

The experimental works presented in this communication were conducted by the collaborative team including Prof. Sekulic and Dr. Fu at University of Kentucky. The computational modelling and theoretical analysis are done here at Washington State University.
Figure 1.1. The characteristic video frames of a clad melting sequence. The substrate is AA 3003 covered by the AA4343 clad and a thin layer of potassium fluoro-aluminate. (a) 573°C, the initial state of the molten flux at the onset of Al-Si melting. (b) 595.2°C, the liquid Al-Si layer formed and the onset of capillary flow. (c) 598.4°C, the formation of a rough surface due to separation of the liquid and residue in situ (alpha phase, Al-Si, lumps). (d) 600°C, the onset of solidification. Photo credits to Prof. Sekulic and Dr. Fu.

Figure 1.2. Schematic of the capillary evolution after melting over non-wetting horizontal substrate and wetting vertical substrate. The initial configuration is a solid block. The arrows show directions of motion of the two triple lines: The receding one and the advancing one.
The current study is organized as follows. In Chapter 2, we review the main features of the CC model enhanced with the Jaqmin’s (2000) diffusive boundary condition (mathematical details are relegated to Appendices A and B). We also describe the computational instability of the CC model which arises when the two fluids have widely dissimilar mass densities. Then, we derive the IC model as a systematic approximation of the CC model and discuss parameter identification and dimensional analysis. In Chapter 3, numerical results are compared to theory and experiments, and the effect of the purely computational parameter – the phase field mobility, is discussed. Summary and conclusions are given in Chapter 4.
CHAPTER 2
PROBLEM FORMULATION

2.1 Compositionally compressible (CC) flow

We consider the isothermal flow of two immiscible fluids. The only mixing occurs in the thin interface layer, where a smooth transition in concentration of the components occurs. The key elements of the compositionally compressible (CC) model are:

(1) Adoption of the barycentric velocity as the continuum velocity field,
(2) Mass density as a function of the phase field variable, and,
(3) The thermodynamic requirement for the dependence of chemical potential on pressure, which implies that the specific\(^2\) free energy of a homogeneous fluid can only depend on \(\phi\), so that the homogeneous free energy density is proportional to the mass density.

Consider an elementary volume \(V\), containing two species\(^3\), with partial mass densities \(\rho^A\) and \(\rho^B\), such that their sum is the total mass density \(\rho\):

\[
\rho^A = \frac{M^A}{V}; \quad \rho^B = \frac{M^B}{V}; \quad \rho = \frac{M^A + M^B}{V} = \rho^A + \rho^B. \tag{2.1}
\]

The components velocities are \(u_A\) and \(u_B\). Definition of the continuum velocity field, \(u(x)\), as the barycentric velocity:

\[
\rho u = \rho^A u_A + \rho^B u_B. \tag{2.2}
\]

\(^2\) Specific energy = energy per unit mass; energy density = energy per unit volume.
\(^3\) Since we consider immiscible fluids, the distinction between species and phase is immaterial.
preserves the total momentum of the mix and implies the standard mass balance (continuity equation):

\[
\frac{\overline{D}\rho}{Dt} = -\rho \nabla \cdot \mathbf{u}, \quad (2.3)
\]

where \(\overline{D}/Dt\) represents the material derivative. The phase field variable is introduced as a mass fraction:

\[
\phi = \frac{\rho^A}{\rho}, \quad (2.4)
\]

so that the mass density is a function only of the phase field variable (i.e., no pressure dependence of mass density):

\[
\rho = \rho(\phi). \quad (2.5)
\]

The definitions (2.2) and (2.4) imply\(^4\) the diffusion law for the phase field variable, with the diffusional flux proportional to the relative velocity of the component \(A\):

\[
\rho \frac{\overline{D}\phi}{Dt} = -\nabla \cdot \mathbf{q}; \quad \mathbf{q} = \rho^A(\mathbf{u}_A - \mathbf{u}). \quad (2.6)
\]

Further, the functional dependence (2.5) forbids full incompressibility; if any diffusion is to occur, the mass density of a material element must change:

\[
\frac{\overline{D}\rho}{Dt} = -\rho \nabla \cdot \mathbf{u} = \frac{d\rho}{d\phi} \frac{\overline{D}\phi}{Dt} = -\frac{1}{\rho} \frac{d\rho}{d\phi} \nabla \cdot \mathbf{q} \neq 0. \quad (2.7)
\]

While the choice of the barycentric velocity preserves the momentum of the mix, it fails to preserve the kinetic energy. Specifically, if the kinetic energy is written as:

\[
\overline{K} = \int_V \frac{1}{2} \rho \mathbf{u} \cdot \mathbf{u} dV, \quad (2.8)
\]

\(^4\) The fact the diffusion law (2.6) is the necessary consequence of the choice of the barycentric velocity as the material velocity, does not appear to be acknowledged in the literature. Even Lowengrub & Truskinovsky (1998) introduce (2.6) as an assumption, rather than a proven fact. We give the proof of (2.6) in Appendix A.
then the kinetic energy associated with the diffusional flux is neglected. From (2.2) and (2.6), this error is:

\[ \int_V \frac{q^2}{2\rho \phi(1-\phi)} dV. \]  

(2.9)

However, this kinetic energy plays a role only in the thin interface layer between the two fluids. Elsewhere, the kinetic energy (2.8) is exact. Similarly, compositional compressibility is only relevant in the thin interface region. Elsewhere:

\[ \mathbf{q} = \nabla \cdot \mathbf{q} = 0 \Rightarrow \frac{D\rho}{Dt} = -\rho \nabla \cdot \mathbf{v} = 0. \]  

(2.10)

We consider the following class of two-phase capillary flow problems. The volume of the computational cell, \( V \), has the boundary \( \partial V \), with the unit outward normal \( \mathbf{n} \). A part of the boundary, \( S \), is the interface with a rigid solid. The remainder, \( A \), is the computational cell boundary within the fluid, chosen for convenience, and often with the computational efficiency constraint. We take:

\[ \partial V = S \cup A. \]  

(2.11)

At the solid boundary, we expect no normal diffusional flux. If the boundary \( A \) passes through a single phase, the normal diffusional flux vanishes. Finally, if it is necessary to cut the computational domain through the phase field interface, this is typically done so that the boundary is orthogonal to the interface. Since the diffusional flux is orthogonal to the boundary, the normal flux is still zero. Thus, we will assume that the normal flux vanishes everywhere along the boundary:

\[ \mathbf{n} \cdot \mathbf{q} = 0 \quad \text{on} \ \partial V. \]  

(2.12)
In Appendix B, we give derivation of the governing equations based on the principle of virtual power. Moreover, we show how the diffusive boundary condition (Jacqmin, 2000) arises naturally from the linear dissipation assumption.

Let \( f(\phi) \) be the composition dependent specific free energy of a homogeneous fluid. It is a double-well potential with minima corresponding to the two phases (Figure 2.1). Then, the specific free energy density for a non-homogeneous fluid, thermodynamically consistent with compositional compressibility (Lowengrub & Truskinowsky 1998) is written in the Cahn & Hilliard (1958) form:

\[
\omega(\phi, \nabla \phi) = f(\phi) + \frac{1}{2} \kappa (\nabla \phi)^2.
\]

(2.13)

The parameter \( \kappa \) and the parameters of the function \( f(\phi) \) (Figure 2.1) are related to the interface energy – a physical parameter, and to the width of the diffuse interface – a computational parameter, chosen with some latitude. More detailed discussion of the model parameters will be given later in connection with the incompressible model.

![Figure 2.1](image-url)

**Figure 2.1.** Free energy density of a homogeneous fluid as a function of the phase variable. The relevant parameter is \( \Delta f \), see discussion in Section 2.3.
For shorter writing, we introduce the symmetric-deviatoric gradient operator:

\[ \Box \mathbf{u} = \mathbf{u} \nabla + \nabla \mathbf{u} - \frac{2}{3} (\nabla \cdot \mathbf{u}) \mathbf{I}, \]

(2.14)

so that the viscous deviatoric stresses are written as:

\[ \tau(\phi, \nabla \mathbf{u}) = \mu(\phi) \Box \mathbf{u}, \]

(2.15)

Like mass density in (2.5), the viscosity \( \mu(\phi) \) is interpolated between the values corresponding to the two fluids. Similarly, the solid-fluid interface energy \( \gamma(\phi) \) is interpolated between the values corresponding to the two fluids.

The governing equations, derived in Appendix B, consist of the Navier-Stokes equations (NS) modified to include the capillary stresses\(^5\), the compositional compressibility condition (CC), and the Cahn-Hilliard 4\(^{th}\) order diffusion equation, which we write here as two 2\(^{nd}\) order differential equations (CH1 and CH2) for the phase variable and the chemical potential \( M \):

\[
\begin{align*}
\text{NS:} & \quad \rho \frac{\mathbf{D} \mathbf{u}}{Dt} = \rho \mathbf{g} + \nabla \cdot (\mu \Box \mathbf{u} - \kappa \rho \nabla \phi \nabla \phi - \pi \mathbf{I}); \\
\text{CC:} & \quad \nabla \cdot \mathbf{u} = -\frac{1}{\rho^2} \frac{d \rho}{d \phi} \nabla \cdot (B \nabla M); \\
\text{CH1:} & \quad \rho \frac{\mathbf{D} \phi}{Dt} = \nabla \cdot (B \nabla M); \\
\text{CH2:} & \quad M = \frac{df}{d\phi} \left( \frac{1}{\rho} \nabla \cdot (\kappa \rho \nabla \phi) - \frac{\pi}{\rho^2} \frac{d \rho}{d \phi} \right). 
\end{align*}
\]

(2.16)

\(^5\) The NS equation differs from the original Lowengrub \& Truskinowsky (1998) result in the treatment of volumetric viscosity. Consider again the CC condition in the form (2.7). It is clear that the diffusional dissipation (work done on \( \mathbf{q} \)) represents the same physical quantity as volumetric viscosity (work done on \( \nabla \cdot \mathbf{u} \)). Thus, in their original equations, Lowengrub \& Truskinowsky are counting the volumetric dissipation twice. The equations (2.16) have been derived using the principle of virtual power (Appendix B), which requires careful definition of work (power) conjugate fields, so that the viscous dissipation is strictly deviatoric, while the diffusional dissipation is strictly volumetric.
The field \( \pi(\mathbf{x}) \) is identified with the non-capillary pressure. It arises as the Lagrange multiplier associated with the enforcement of the CC condition. The diffusive boundary condition on the solid surface, derived in the Appendix B, is:

\[
\frac{\overline{D}\phi}{Dt} = -b\left(\kappa \rho \mathbf{n} \cdot \nabla \phi + \frac{d\gamma}{d\phi}\right); \quad \text{on } S. \tag{2.17}
\]

Such boundary condition has been proposed by Jacqmin (2000) motivated by the equilibrium condition (vanishing of the term in parentheses). In Appendix B, we show that this boundary condition arises as a direct consequence of the linear non-negative dissipation on the solid boundary (i.e., dissipation accompanying the motion of the triple line).

When applied to the combination of fluids with vastly different mass densities (i.e., a liquid and a gas), the above equations exhibit computational instabilities at the triple line. The diffuse interface spreads in an unstable manner, as illustrated in Figure 2.2. The initial acceleration of the triple line is quickly followed by an unstable spreading of the interface in the triple line region which eventually leads to computational failure. In investigating this instability, we found that the computational instability is not limited to the triple line region. The immediate cause of the instability is the appearance of the shadow band of velocity divergence tracing the interface band at the distance comparable to the interface thickness (Figure 2.2(b)). Similar shadow bands have been observed in multiphase flows without capillary walls (Guo et al. 2014).

In investigating this instability our first suspicion was that the presence of \( \rho \) in (2.17) which varies by few orders of magnitude across the interface is the cause of instability. An alternative to the above formulation where the diffusive boundary condition is independent of mass density can be readily obtained by assuming a different specific energy:
The alternative CC model has somewhat different governing equations. The CC and CH1 equations remain the same, while in the NS equation the capillary stresses are now independent of mass density. The CH2 equation is substantially different:

\[
\begin{align*}
\text{NS:} & \quad \rho(\phi) \frac{\partial \mathbf{u}}{\partial t} = \rho \mathbf{g} + \nabla \cdot (\mu(\phi) \nabla \mathbf{u} - \kappa \nabla \phi \nabla \phi - \pi \mathbf{I}) \\
\text{CH2:} & \quad M = \frac{df}{d\phi} - \frac{\kappa}{2\rho^2} \frac{d\rho}{d\phi} (\nabla \phi)^2 - \frac{\kappa}{\rho} \nabla^2 \phi - \frac{\pi}{\rho^2} \frac{d\rho}{d\phi}
\end{align*}
\]  

And the diffusive boundary condition is now independent of mass density:

\[
\frac{\partial \phi}{\partial t} = -b \left( \kappa \mathbf{n} \cdot \nabla \phi + \frac{d\gamma}{d\phi} \right) \quad \text{on } S.
\]  

Figure 2.2. Instability of the computational CC model with ratio of mass densities of liquid and gas of 811. Capillary rise between parallel walls. (a) Initial contours of the phase field. (b) Contours of the velocity divergence. (c) Magnified instability region circled in (a). (d) Line contours corresponding to (c).
The alternative formulation exhibits the same computational instability as the original one (Figure 2.2). The causes of this instability are not well understood at present. It bears emphasis that uniqueness results are only available for the IC model (Abels 2009). In below, we propose a physical explanation for this instability based on the analysis of the motion of the surrounding incompressible fluid caused by the compositional compressibility of the interface layer.

Consider a problem similar to the one studied by Guo et al. (2014) and illustrated in Figure 2.3, where an initially ellipsoidal drop of a dense incompressible fluid changes shape and becomes spherical. The computational domain includes the large volume of low density incompressible fluid and the volume of the CC interface with some average mass density. If the interface retains its original thickness, its total volume has decreased, so that a large volume of the surrounding incompressible fluid must be moved inwards.

**Figure 2.3.** Ellipsoidal interface evolves into a spherical one and consequently changes its volume. If the rest of the fluid is incompressible, such volume change requires large scale motion of the surrounding fluid (indicated by arrows) and shrinking/expansion of the computational cell boundary.
This is clearly not the physical reality. The local change of mass density will not be compensated by global motion of a large volume but rather by some local changes. On the level of the computational model, the solution will be prevented from evolving in this way, either by huge inertia of the surrounding fluid, or by the boundary conditions (consider rigid boundaries or periodic boundary conditions). How can this local accommodation be accomplished?

First, the thickness of the interface may increase, so that the total volume of the interface remains constant. Consider the case where \( t_B \ll t_\mu \). The large diffusional mobility \( B \) ensures that the shape of the interface is determined by the equilibrium version of the Cahn-Hilliard equation: \( \nabla^2 \mathcal{M} \approx 0 \). This does not allow for significant thickening of the interface, so that this option may not be available. Second, the interface fluid may exhibit elasticity so that the elastic volume change counteracts the compositional volume change. This option is not available simply because fluid elasticity is not included in the model. Finally, the numerical model may locally compensate for the local volume change of the interface by changing the mass density in the close neighborhood, hence the ghost (shadow) velocity divergence. It is not clear whether such solution is a valid solution to the mathematical problem, or does it emerge only as a numerical solution which minimizes the error norm.

While the CC formulation is locally thermodynamically consistent, there appears to be a global inconsistency between the compositional compressibility of the interface layer and incompressibility of the rest of the fluid. This can probably be resolved by introducing elasticity into the interface layer, but with a high mathematical and computational cost (cf. Steigmann & Li 1995). On the other hand, the IC model, previously considered as an ad hoc simple model, is derived here as a systematic approximation to the CC model. Such derivation allows (in
principle) error estimates, although the discussion in this work is limited to qualitative arguments about the size of the error.

2.2 Incompressible flow as an approximation to the CC flow

The functional dependence $\rho(\phi)$ is a necessary component of a phase field model for materials with different mass densities. This alone disallows the exact incompressibility $(\frac{\partial \rho}{\partial t} / \rho = 0 \Rightarrow \frac{\partial \rho}{\partial t} / \rho = 0)$. Thus, an incompressible model can only be devised as an approximation to the compositionally compressible model.

Consider $\rho(\phi)$ such that:

$$\frac{1}{\rho} = \frac{\phi}{\rho_{0A}} + \frac{1-\phi}{\rho_{0B}}; \quad \phi \in [0,1],$$

where $\rho_{0A/B}$ are the densities of pure components. Then:

$$\frac{1}{\rho^2} \frac{d\rho}{d\phi} = \frac{\rho_{0A} - \rho_{0B}}{\rho_{0A}\rho_{0B}} = \alpha = \text{constant.}$$

Now we consider alternative continuum velocity field:

$$\mathbf{v}(\mathbf{x}) = \mathbf{u}(\mathbf{x}) - \alpha \mathbf{q}(\mathbf{x}),$$

which, from (2.7) and (2.21), is divergence-free:

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

Note that, outside the thin interface region, the new velocity is identical to the old one:

$$\mathbf{v}(\mathbf{x}) = \mathbf{u}(\mathbf{x}); \quad \text{for } \mathbf{x} \in V \setminus I,$$
where \( V \) is the domain with size \( L^3 \), and \( I \) is the interface domain with size \( L^2h \). The interface thickness \( h \) must be small compared to the smallest radius of interface expected curvature \( R_{\text{min}} \), which is typically small compared to the domain size:

\[
h \ll R_{\text{min}} \ll L. \tag{2.26}
\]

If instead of the barycentric velocity \( \mathbf{u} \), we take \( \mathbf{v} \) to be the material velocity of the continuum, the error is only in the momentum and kinetic energy of the thin interface layer. In other words, only the inertia of the thin interface layer is affected. This appears to be a good approximation.

The new material velocity implies the new material derivative. For a scalar field \( Y(\mathbf{x}) \) advected with the velocity field \( \mathbf{v}(\mathbf{x}) \):

\[
\frac{D Y}{D t} = \frac{\partial Y}{\partial t} + \mathbf{v} \cdot \nabla Y = \frac{\bar{D} Y}{D t} + (\mathbf{u} - \mathbf{v}) \cdot \nabla Y. \tag{2.27}
\]

We introduce the new phase field variable, \( c \), based on the linear interpolation of mass density:

\[
\rho = c\rho_A + (1-c)\rho_B; \quad c \in [0,1]. \tag{2.28}
\]

Then, it is easily shown (Appendix C) that the diffusion equation (2.6) can be replaced by:

\[
\frac{D c}{D t} = -\nabla \cdot \mathbf{J}, \tag{2.29}
\]

where the new flux vector is related to the old one as:

\[
\mathbf{J} = \frac{\rho}{\rho_A \rho_B} \mathbf{q}. \tag{2.30}
\]

Now to complete the definition of the new continuum, the mass density \( \rho \) should be convected with the new velocity \( \mathbf{v} \), implying:

\[
\frac{D \rho}{D t} = -\rho \nabla \cdot \mathbf{v} = 0. \tag{2.31}
\]
Clearly, this new continuum has an intrinsic inconsistency. The dependence (2.28) is in direct contradiction to (2.31):

\[
\frac{D\rho}{Dt} = \frac{d\rho}{dc} \frac{Dc}{Dt} \neq 0.
\] (2.32)

One way to formulate a mathematically consistent model is to:

(a) Enforce the incompressibility condition (2.24) instead of the mass balance (2.31), and,

(b) Assume that the power expanded on mass density change \( D\rho/Dt \) vanishes.

Then, the function \( \rho(c) \) appears in the governing equations as a parameter function. Such procedure produces the IC model in the form of the original model H (Hohenberg & Halperin 1977, Gurtin et al. 1996). The derivation is given in Appendix D. It proceeds along the same steps as the one for the CC model (Appendix B) including: linear dissipation, energy balance and the principle of virtual power. An interesting technical point is the split of the rate of change of potential energy into the “chemical” and “densification” rates (D.4), where the latter is neglected following the discussion above.

For an IC fluid, the potential energy density need not depend on the mass density as in the CC case. The phase-field form of the energy density (Cahn & Hilliard 1958) is:

\[
\omega(c, \nabla c) = f(c) + \frac{1}{2} \kappa ( \nabla c )^2.
\] (2.33)

The remainder of the procedure is shown in Appendix D. The final governing equations take the following form:

\[
\text{NS:} \quad \rho \frac{Dv}{Dt} = \rho g + \nabla \cdot \mu (\nabla v + \nabla v) - \kappa \nabla c \nabla v - \nabla \pi;
\]

\[
\text{IC:} \quad \nabla \cdot v = 0;
\]

\[
\text{CH1:} \quad \frac{Dc}{Dt} = -\nabla \cdot (B \nabla \mathcal{M});
\]

\[
\text{CH2:} \quad \mathcal{M} = \frac{df}{dc} - \kappa \nabla^2 c
\] (2.34)
As before, \( \pi(\mathbf{x}) \) is the non-capillary pressure, which arises as the Lagrange multiplier associated with the enforcement of the IC condition, so that the total pressure is given as:

\[
p = \pi + \frac{1}{3} \kappa (\nabla \phi)^2 .
\]  

(2.35)

The diffusive boundary condition ensuring a non-negative dissipation on the solid boundary is:

\[
\frac{Dc}{Dt} = -b \left( \kappa \mathbf{n} \cdot \nabla c + \frac{d\gamma}{dc} \right); \quad b > 0; \quad \text{on } S .
\]  

(2.36)

This completes the derivation of the IC model as a systematic approximation to the CC model.

The two key steps are:

1. Definition of the divergence-free velocity field which is equal to the barycentric velocity outside the thin interface region

2. The assumption of incompressibility allowing: adoption of the divergence-free velocity as the continuum velocity, and, energy density independent of mass density.

The IC model has several versions which are \textit{mathematically equivalent}, but produce numerical differences. The main differences are in the order of derivatives of the main variables which are used in the formulation. This has an effect on the finite element computations, particularly if low order interpolation is used. One alternative is obtained by noting that:

\[
\nabla \cdot (-\kappa \nabla \epsilon \nabla c) = \mathcal{M} \nabla c - \nabla \omega .
\]  

(2.37)

Next, we note that division of pressure into capillary and non-capillary parts is arbitrary and replace the variable \( \pi \) with:

\[
\bar{\pi} = \pi + \omega .
\]  

(2.38)

This yields the alternative NS equation in the form:

\[
\text{NS-alt: } \quad \rho \frac{D\mathbf{v}}{Dt} = \rho g + \nabla \cdot (\mu \nabla \mathbf{v}) + \mathcal{M} \nabla c - \nabla \bar{\pi} .
\]  

(2.39)
where the total pressure is now given as:

\[ p = \bar{p} - f(c) - \frac{1}{6} \kappa (\nabla c)^2 . \]  

(2.40)

2.3 Physical and computational parameters

By physical parameters, we mean the parameters in the standard sharp interface fluid dynamics model. In addition to mass densities and viscosities of two fluids, A and B: \( \rho^A_0 \) and \( \mu^A_0 \), these include the interface energies, in the sense of Gibbs (1878), between the two fluids \( \Gamma \), and between the solid and the two fluids: \( \gamma_{Bsol} \) and \( \gamma_{Asol} \).

The interface width \( h \) is a computational parameter, chosen so that it is significantly smaller than the smallest expected radius of curvature. The equilibrium solution to the problem of flat infinite interface between fluids A and B gives the one-to-one relationship between the pair \( (\Gamma, h) \) and the phase field parameters \( (\kappa, \Delta f) \):

\[ \kappa = \frac{3}{2} \Gamma h; \quad \Delta f = \frac{3}{4} \Gamma / h . \]  

(2.41)

Experience indicates that the interface width varies very little during the flow and is at all times very close to the equilibrium value (Asle Zaeem & Mesarovic 2009, Asle Zaeem et al. 2011).

It is usually assumed that the exact shape of the free energy function (Figure 2.1) has no measurable effect on either equilibrium or kinetics of the multiphase flow. The only relevant parameters are the relative values of minima and maxima. In this case (Figure 2.1), the only relevant parameter is \( \Delta f \).
For the process driven by the capillary forces, the three characteristic times associated with the governing equations (2.34) and (2.36) are determined by dimensional analysis: the triple line mobility time $t_b$, the bulk mobility time $t_B$, and the viscosity time $t_\mu$:

$$
t_b = \frac{2}{3b\Gamma}; \quad t_B = \frac{2h^3}{3B\Gamma}; \quad t_\mu = \frac{2\mu_A^0 h}{3\Gamma}.
$$

(2.42)

For immiscible fluids, the bulk mobility $B$ has no physical significance; it is a computational parameter that enables diffusional rearrangements within the phase field interface layer. It should be large enough not to obstruct the motion, that is, the characteristic time for bulk mobility should be shorter than other two characteristic times:

$$
t_B \leq t_b; \quad t_B \leq t_\mu.
$$

(2.43)

The triple line moves by surface diffusion (uncorrelated motion of atoms) as opposed to slipping (correlated motion of atoms). The triple line mobility $b$ determines the dissipation in the surface diffusion process. It could be, in principle, determined from molecular dynamics (MD) simulations, or from the combination of molecular statics (energy landscape) simulations and activation rate theory. The most practical solution appears to be matching of experimental kinetics in the regime where the capillary forces dominate.
3.1 Numerical method, boundary and initial conditions

The computations are carried out with quadratic Lagrange interpolation for the velocity \( \mathbf{v} \), the phase field variable \( c \) and the chemical potential \( M \). Linear Lagrange interpolation is used for the non-capillary pressure \( \tilde{\pi} \). The alternative model (2.39) is used for the modified Navier-Stokes. The remaining governing equations are given in (2.34). The commercial FE software COMSOL Multiphysics (2015) with the parallel sparse direct solver is used. The time-stepping is based on the backward differentiation formula. The initial conditions are common to all runs.

The initial velocity and the diffusion potential are identically zero:

\[
\mathbf{v}(\mathbf{x}, 0) = 0; \quad M(\mathbf{x}, 0) = 0.
\]  

The initial configuration is always such that the denser fluid is on the bottom, so that the equilibrium solution (obtained from \( M = 0 \) and (2.32)) is:

\[
c(x, y, 0) = \frac{1}{2} \left( 1 - \tanh \frac{2y}{h} \right).
\]  

The initial pressure must be consistent with (2.38) and equilibrium under gravity:

\[
\tilde{\pi}(x, y, 0) = \tilde{\pi}_0(x, y) = -g \int_{y_0}^{y} \rho(c(y))dy; \quad \tilde{\pi}_0(x, 0) = 0.
\]

The types of boundary conditions used for different problems are as follows:

- Wetting wall boundary condition: On the fluid-solid interface, no slip/penetration boundary condition for velocity (\( \mathbf{v} = 0 \)) and the diffusive boundary condition (2.36).
• Inlet and outlet boundary conditions apply at the ends of the computational cell in fluids. Both are characterized by the equilibrium pressure distribution (3.3) and the condition that the phase field interface is orthogonal to the boundary:
\[ \mathbf{n} \cdot \nabla c = 0. \] (3.4)
The inlet boundary condition requires the vanishing viscous stresses (Panton 1996), while the outlet boundary condition (Ranacher 2007) disallows normal gradient of velocity. Moreover, we assume the equilibrium pressure distribution at both fluid boundaries:
Inlet: \( \mathbf{n} \cdot [\mu (\nabla \mathbf{v} + (\nabla \mathbf{v})^T)] = 0; \quad \vec{x} = \vec{x}_0(y); \)
Outlet: \( \mathbf{n} \cdot \nabla \mathbf{v} = 0; \quad \vec{x} = \vec{x}_0(y). \) (3.5)
The fluid (inlet/outlet) boundaries are assumed to be sufficiently removed from the main region of interest. We have verified that the liquid coming through the inlet boundary acquires the expected laminar flow characteristics at short distance from the boundary.

• On all boundaries, we require vanishing normal diffusional flux:
\[ \mathbf{n} \cdot \nabla \mathbf{M} = 0. \] (3.6)
The justification for (3.6) is obvious for the solid boundary. On the fluid boundary, (3.6) is consistent with normality of the phase field interface (3.4) (diffusive flux is away orthogonal to the phase field level lines). In other words, we require that all matter is transported into the computational cell by flow (\( \mathbf{v} \)), and none by diffusion (\( \mathbf{q} \)).

For various runs, the macroscopic length scale \( L \) is estimated from equilibrium solutions. This also represents the order of magnitude of the interface radius of curvature. Then, the interface width \( h \) is selected to be \( h \ll L \). In the runs presented below, the interface width varies between 0.09\( L \) for the meniscus between parallel plates, 0.07\( L \) for the T-joint case, 0.01\( L \) for the non-wetting case, and 0.065\( L \) for the rough wetting case. The adaptive mesh consisting of 2\textsuperscript{nd}
order triangular elements was finest at the interface. The element size is about $0.1h$ in all runs. The adaptive mesh density is determined based on the norm of the gradient of the phase field variable (COMSOL 2015). Examples of mesh evolution are shown in the Figures 3.4 and 3.11.

### 3.2 Equilibrium

To benchmark our capillary flow computational formulation with IC flow, we consider the simple 2D geometry of the capillary rise between parallel plates. The physical parameters are shown in Table 3.1. The computational results are compared to the analytic solution for the sharp interface model. The computational interface shape matches perfectly the analytical shape (Figure 3.1(b)). Moreover, the pressure jump at the interface with curvature $1/R$:

$$\Delta p = \frac{\Gamma}{R}, \quad (3.7)$$

is accurately represented by the computational phase field model (Figure 3.1(c)). The negative pressure represents interface tension (Davis & Scriven 1982).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Silicone</th>
<th>Water</th>
<th>Trillium</th>
<th>Nitrogen</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho$ [kg/m$^3$]</td>
<td>Density</td>
<td>960</td>
<td>997</td>
<td>2444.5</td>
<td>1.184</td>
</tr>
<tr>
<td>$\mu$ [Pa.s]</td>
<td>Viscosity</td>
<td>0.0096</td>
<td>0.0008937</td>
<td>0.001</td>
<td>1.849×10$^{-5}$</td>
</tr>
<tr>
<td>$\Gamma$ [J/m$^2$]</td>
<td>Interface energy</td>
<td>0.02089</td>
<td>0.07197</td>
<td>0.849</td>
<td>-</td>
</tr>
<tr>
<td>$\theta$ [degree]</td>
<td>Equilibrium angle</td>
<td>30</td>
<td>25</td>
<td>23 (wetting)</td>
<td>6 (non-wetting)</td>
</tr>
<tr>
<td>$l$ [m]</td>
<td>Interface thickness</td>
<td>1×10$^{-4}$</td>
<td>1×10$^{-4}$</td>
<td>1×10$^{-4}$</td>
<td>-</td>
</tr>
<tr>
<td>$\Delta f$ [J/m$^3$]</td>
<td>Potential barrier</td>
<td>1.56675×10$^2$</td>
<td>5.4645×10$^2$</td>
<td>6.3675×10$^3$</td>
<td>-</td>
</tr>
<tr>
<td>$\kappa$ [J/m]</td>
<td>Constant</td>
<td>3.13350×10$^{-6}$</td>
<td>1.09290×10$^{-5}$</td>
<td>1.2735×10$^{-4}$</td>
<td>-</td>
</tr>
</tbody>
</table>
We investigate the effect of the bulk mobility parameter $B$ on the diffuse interface kinetics. The triple line mobility $b$ is obtained from experiments. Details of the extraction procedure are discussed in the next subsection and also in Fu et al. (2016). The physical parameters (Table 3.1) and the selected interface width determine the physical characteristic times in (2.42), such that:

$$\frac{t_b}{t_\mu} \approx \frac{2}{3} \times 10^3.$$  \hfill (3.8)

Thus, the triple line mobility is the dominant physical process. Therefore, we investigate the effect of the relative magnitude of the bulk mobility $B$, characterized by $t_B$ (2.42). The results are shown in Figure 3.2. In the early stages the kinetics are practically indistinguishable for a 4 order of magnitude change in $B$. The differences appear at later stages, closer to the equilibrium solution.
3.4 Computational and experimental kinetics

To benchmark the computational kinetics against the experimental one, we consider the wetting process at high-temperature liquid metal and low-temperature non-metallic liquid systems. The study is performed numerically and is supported by experimental results simultaneously carried out by a collaborating team. High-temperature spreading to be considered takes place during a transient formation of the free liquid surface in a so-called wedge-tee configuration, using aluminum (AA3003) and alumina (Al₂O₃) metals as substrates. In a high-temperature molten metal experiment series, Al-Si clad alloy, Trillium™, is used. In addition, low-temperature benchmark cases of spreading of water and silicone oil over quartz glass as a substrate are considered. Benchmark tests of low-temperature systems are being conducted under ambient environment’s conditions. The details of the experimental procedure are given in Fu (2016).
3.4.1 Capillary flow of the low-temperature non-metallic liquid systems

Two sets of low-temperature experiments include the spreading of silicone oil and water have been performed at the ambient temperature to investigate the kinetics of the capillary rise and to verify the equilibrium conditions such as contact angles, and meniscus shapes.

We consider the T-joint experimental configuration shown in Figure 3.3 (Fu et al. 2016). The fluid silicone oil/water rests on the horizontal glass plate. The vertical glass plate is brought gradually into contact and the capillary rise follows. Computational results for one half of the symmetric configuration are shown in Figure 3.4, for silicone oil together with the adaptive mesh.

![Figure 3.3](image-url)

**Figure 3.3.** Schematic of the T-joint configuration in experiment setup for low-temperature case. (a) Before the wetting process. (b) After the wetting process.
The two low-temperature wetting systems have viscosities differing by an order of magnitude: 1 [mPa.s] for water, and 10 [mPa.s] for silicone oil, both at room temperature. This difference results in a slower kinetics of silicone oil, Fig. 3.5. The characteristic times of wetting when the triple line achieves 80% of its equilibrium height are about $10^{-1}$ and $10^0$ s for water and silicone oil, respectively. This means that viscosity plays a decisive role in controlling the kinetics.

The visualizations of the benchmark systems of water and silicone oil confirm fully this hypothesis supported well by the empirical evidence. In case of water, a strong depletion of the liquid film is observed soon after the flow driven by surface tension takes place. This leads to a formation of a surface wave spreading along the interface outward of the joint. In case of silicone oil this wave is not observed. Apparently, in the first case the viscous boundary layer
(Schlichting 2000) formed along the horizontal surface is thinner than the film thickness. So water easily flows into the joint until the film is depleted. In the second case, the viscous boundary layer is as thick as the film. Therefore, the viscous forces control the flow in the film preventing wave formation at the flow onset. This effect is clearly registered experimentally and is also reproduced in simulations, Fig. 3.6.

**Figure 3.5.** Triple line kinetics in log-log scale for (a) Water, and (b) Silicone oil. Comparison between numerical results and experiments. The triple line mobility $b$ is fitted to the initial slope.

**Figure 3.6.** The formation of the wave in the water pool following the initial surge of the triple line (a) Experimental (19ms); (b) Computational (20ms).
In the early stages of this experiment, the triple line surges rapidly. Only a small quantity of liquid is moved initially, so that capillary forces dominate over viscous and inertial forces. We fit the triple line mobility to the initial tangent to the triple line position history, and benchmark the results against the remainder of the experiments. The results are illustrated in Figure 3.5. The computational results match the experimental measurements very closely.

The equilibrium shape of the meniscus can be determined analytically (neglecting the mass density of the gas). The comparison between computational and analytical equilibrium meniscus profiles is shown in Fig. 3.7. The profiles are practically identical.

![Graph showing computational and analytical equilibrium profiles of menisci](image)

**Figure 3.7.** Computational and analytical equilibrium profiles of menisci. The parameters are listed in Table 3.1.

Figure 3.8 demonstrates an excellent agreement between numerical prediction and experimental data for the dynamic contact angle. Following (i) very good agreement in predictions of the equilibrium state, (ii) confirmed kinetics of the triple line, and (iii) very good prediction of the dynamic contact angle for both benchmark systems, we may confirm that the developed phase-field method offers a reliable tool for modeling the process of spreading of low-temperature non-metallic liquid systems.
3.4.2 Capillary flow of the high-temperature liquid metal systems

Two sets of experiments include the spreading of the molten Al-Si alloy (Trillium) over the rough wetting and smooth non-wetting surfaces in the form of the T-joint configuration shown in Figure 3.9. The Trillium™ alloy (Fu et al. 2016) is essentially Al-Si alloy with embedded flux nano-particles. The experiments are carried out at the high temperature in ultra-high purity nitrogen atmosphere. The T-joint experimental configuration of the Trillium metal coupon with 400-micron thickness located on the wetting aluminum (AA3003) surface, and the Trillium metal coupon with the 800-micron thickness placed on the non-wetting alumina surface are presented in Figures 3.9(a) and (b), respectively. In both cases, the assemblies are heated at the high-temperature between 560°C and 590°C to melt the clads and the capillary rises take place.
over the vertical aluminum sheet. The details of the experimental procedure are given in Fu (2016).

![Diagram of experiment setups](image)

**Figure 3.9.** Schematic of the experiment setups: (a) 400-micron thickness Trillium filler on the wetting aluminum substrate. (b) 800-micron thickness Trillium filler on the non-wetting alumina substrate.

### 3.4.2.1 Spreading of the molten Trillium metal over the rough wetting surface

The initial and final snapshots of the molten Al-Si alloy profiles on the wetting aluminum substrate are shown in Figure 3.10. In this experiment, a solid Al-Si coupon is placed on the horizontal aluminum surface and melted. A part of the molten alloy flows to the meniscus with 23° contact angle on the vertical sheet. At the horizontal surface, the molten alloy flows over the rough surface formed by the process discussed in the Chapter 1 and illustrated in Figure 1.1. A portion of the melted alloy remains isolated from the meniscus between the solid lumps on the horizontal mating surface. The measurements of the meniscus profile volume indicate that up to 60% of the available Al-Si alloy ends up in the joint. This observation suggests that the roughness created by partially melted Al-Si alloy leaves the part of initial molten metal disconnected from the meniscus as shown in Figure 3.10.
The computational results for one-half of the symmetric configuration of molten alloy, spreading over the wetting aluminum surface are shown in Figure 3.11, together with the adaptive mesh. In the simulation, it is assumed that all available Al-Si alloy is completely melted. The roughness of the horizontal surface is modelled with the wavelength of 1 mm and the amplitude 325 μm. The smallest radius of curvature on the rough surface is significantly larger than the interface width. The breakups of the liquid film take place along the rough substrate. Some of the liquid alloy is caught in the troughs of the rough surface, while the remainder rises up the vertical sheet and forms the meniscus.

Figure 3.10. Snapshots of the Trillium profiles on the wetting aluminum surface captured from experiment (a) Initial condition. (b) Equilibrium condition; θ=23°.
The triple line kinetics for the molten Al-Si alloy spreading over the wetting aluminum substrate is shown in Figure 3.12(a). In the initial stages of the capillary flow, the capillary forces dominate (Fu et al. 2017, Dehsara et al. 2017). We fit the triple line mobility $b$ to the initial tangent to the triple line position history, and the results are benchmarked against the remainder of the experiments. The comparison between computational and experimental equilibrium
meniscus profiles is shown in Figure 3.12(b). The computational results match the experimental measurements very closely.

3.4.2.2 Spreading of the molten Trillium metal over the non-wetting surface

The experimental procedure involves in situ real time monitoring of the capillary flow over the assembly of two surfaces forming a wedge-T configuration (Figure 3.13(a)). The horizontal non-wetting surface is alumina (Al₂O₃), while the vertical surface is a thin aluminum sheet. The Al-Si alloy (a folded foil) in the solid state is positioned over the horizontal alumina substrate. The equipment used and the associated procedure are described in detail in Fu (2016) and Fu et al. (2016). During the melting process, the molten metal moves up the vertical sheet. Since the horizontal substrate is a polished non-wetting alumina plate, there is no molten liquid metal on the horizontal surface in the equilibrium state shown in Figure 3.13(b). Note that the vertical solid surface is nominally the same Al alloy (AA3003). Nevertheless, the equilibrium contact
angle with the vertical surface is now much smaller: about 6°. While we don’t have sufficient information to fully explain this difference, we speculate that this is the effect of different flux availability in the two cases. In the previous case (wetting horizontal surface), the flux was partially used up in reaction with the horizontal surface, so that the remainder is insufficient to fully remove the oxide from the vertical surface. The resulting contact angle of 23° is the effective contact angle with heterogeneous Al+oxide surface. In the non-wetting case (Figure 3.13) all the flux embedded in the Al-Si alloy is available for removing the oxide on the vertical surface, so that the contact angle is between two Al alloys and close to zero.

**Figure 3.13.** Snapshots of the Trillium profiles on the non-wetting alumina surface captured from experiment. (a) Initial condition, (b) Equilibrium condition.

The comparison between computational and experimental results is shown in Figure 3.14. First, note that the motion of triple line on the vertical surface (Figure 3.14(c)) is characterized by the inflection point at about \( t=30 \text{s} \). This is a qualitative difference from all previous results,
those reported in Fu et al. (2016) and those shown in Figure 3.12. In all previous cases, the initial rapid acceleration is followed by the gradual deceleration and approach to equilibrium. In this case, the triple line accelerates gradually for the first 30 seconds. To explain this qualitative difference, we first note that the receding contact line (Figure 3.14(c)) moves rapidly from the very start. Thus, the mobility of the receding contact line is not the factor preventing the expected rapid surge of the advancing contact line. What remains is the viscosity of the melting alloy.

Since in the early stages of the non-wetting experiment, the melting process and the capillary flow over the non-wetting substrate are happening simultaneously, the kinetics of the non-wetting case can be split into two stages: before and after the inflection point. In the first stage, the Al-Si coupon is gradually melting and its viscosity varies between solid and liquid phases, so that the viscous forces have significant impact on the flow. After the inflection point, the melting process is completed and pure liquid Trillium is climbing over the vertical substrate by dominant capillary forces. Thus, the phase transition from solid phase to liquid phase is finalized at the inflection time 30s and the solid alloy viscosity has been reduced to the molten alloy viscosity.

We assume that a high value of viscosity of the initial coupon of about 1 Pa-s, while melted alloy has viscosity 0.001 Pa-s (Fu et al. 2016). In order to model the variation of the viscosity between the solid and molten phases before the inflection time, we use the simplest linear interpolation. After the inflection time, when the alloy is completely melted, we consider a constant molten viscosity such that:

\[
\mu(t) = \begin{cases} 
1.001 - t/30, & t \leq 30s \\
0.001, & t > 30s 
\end{cases}
\] (3.9)
The atomistic level processes resulting in shear flow and surface diffusion are very similar. In the absence of better information, we model the time dependence of the triple line mobility $b$ by imposing the condition that the ratio between the characteristic times of viscous flow ($2\mu h/3\Gamma$) and of the triple line motion ($2/3b\Gamma$) is constant (Dehsara et al. 2017). The triple line mobility function is thus obtained as:

$$b(t) = \frac{t/60}{1.001 - t/30} + 0.5, \quad t \leq 30s$$ (3.10)

The numerical and experimental results illustrated in Figure 3.14 are in a good agreement.

**Figure 3.14.** Numerical, and experimental data of the Trillium spreading over the non-wetting surface. (a) Initial phase field contour. (b) Meniscus profiles in equilibrium condition. (c) Triple line kinetics.
CHAPTER 4
SUMMARAY AND CONCLUSIONS

To develop a reliable computational model for capillary flows, we consider two types of phase field models, the compositionally compressible (CC) and the incompressible (IC) model. The motion of the triple line is assumed to take place by diffusion of fluid atoms on the solid surface. The diffusive boundary condition (Jacqmin 2000) is implemented into the finite element framework for both models.

The thermodynamically consistent CC model exhibits computational instability, initiated by appearance of ghost velocity divergence at some distance from the phase field interface, and further exacerbated by spreading of the triple line. The mathematical nature of this instability is not well understood at present and investigation of existence and uniqueness of solutions for the CC model is suggested for future research. Nevertheless, the physical explanation can be constructed based on the observation that a change of mass density in a small part of the computational domain requires movement of a large mass of surrounding incompressible (pure) phase. This problem does not appear when compositional compressibility of the fluid mix is abandoned and the fluids everywhere are considered fully incompressible.

The IC model was derived as approximation to the CC model. The key to the approximation is the alternative definition of the continuum velocity field, such that the new velocity field has vanishing divergence and matches the old velocity outside the interface layer.

A detailed benchmarking of the IC model is successfully performed against analytical solutions to the sharp interface problem and against experimental kinetics of water and silicone oil spreading over glass plates at the ambient temperature without any interaction. The choice of physical and computational parameters is discussed. It is shown that the choice of the bulk...
diffusional mobility has little effect on the results. We find that the IC model very accurately represents both, the capillary equilibrium and the capillary flow kinetics.

Further, two high-temperature capillary-driven phenomena are considered with the aim to predict the wetting behavior of the molten metal on two distinct solid substrates. The fluid is the molten Al-Si alloy. The geometric configuration of mating surfaces is a T-joint structure. In the 1\textsuperscript{st} case, the horizontal solid surface is rough Al (AA3003) with good wetting properties with Al-Si alloy. In the 2\textsuperscript{nd} case, the horizontal solid surface is ceramics (Al\textsubscript{2}O\textsubscript{3}) which is not wetted by Al-Si alloy. In both cases, the vertical solid surface is Al (AA3003).

In the rough wetting surface case, the roughness of the substrate is formed at the onset of melting and subsequently causes break-ups within the interface, so that a part of initial available Al-Si melt is separated from the meniscus. With the parameter identification procedure, and with the horizontal surface roughness defined so that it matches the observed final volume of the meniscus, the experimental kinetics and equilibrium are well matched by computations.

In the non-wetting case, the concurrent melting and capillary flow produces a time-varying viscosity of the partially molten metal, which plays an important role. It slows down the initial flow, so that the speed of the advancing triple line increases and reaches maximum after the alloy is fully melted. The obvious question is why such effect is not apparent in the first experiment. The answer probably lies in: (1) different composition of the alloy prior to melting, and, (2) large differences in the total displacements of fluid mass. In the wetting case, the Al-Si alloy suffers Si depletion from the bottom prior to melting. The result is that the top melts first and this is sufficient to feed the small initial displacement of the triple line. Since only the melted fluid flows, the effective viscosity is always small (albeit on variable geometry of the flow). In the non-wetting case, the alloy is homogeneous but the melting is non-homogeneous and the
fluid that flows is a mix of melted alloy with suspended solids. Our rough model with time-varying viscosity captures this phenomenon reasonably well.

In conclusion, while incompressible phase-field model for capillary flow with diffusive triple line motion can be applied for the high-temperature wetting of a liquid metal, such application requires careful analysis. The underlying assumption is that the reactivity of the liquid metal with the substrate is limited to the small region near the triple line so that a single parameter – the effective triple line mobility captures the kinetics. In addition to reactivity (Wheeler at al 2010), further research should address: the 3D nature of the flow over the rough surface, and, the concurrent melting and capillary flow.

4.1 Future work

Future research directions include:

- Developing IC phase-field model by considering additional physical effects such as chemical reactions take place at the high-temperature capillary flows.

- Surface topography has been an important factor impacting the spreading kinetics. In the current work, we applied flat- and sinusoidal-bed shape to study the spreading kinetics. Various surface roughness should be identified as another variable in future study.

- The mathematical nature of the instability in the compositionally compressible phase-field model is not well understood at present and investigation of existence and uniqueness of solutions for the CC model is suggested for future research.
APPENDIX
APPENDIX A

PHASE FIELD DIFFUSION AS A CONSEQUENCE OF BARYCENTRIC VELOCITY

The material rate of change of $\rho^A$ can be written in two ways. From (2.4-5):

$$\frac{D\rho^A}{Dt} = \rho \frac{D\phi}{Dt} + \phi \frac{D\rho}{Dt}.$$  \hspace{1cm} (A.1)

On the other hand, using the balance for $\rho^A$ and the total mass balance (2.3):

$$\frac{D\rho^A}{Dt} = \frac{\partial \rho^A}{\partial t} + \mathbf{u} \cdot \nabla \rho^A = \nabla \cdot (\rho^A \mathbf{u}_A) + \mathbf{u} \cdot \nabla \rho^A$$

$$= -\nabla \cdot (\rho^A (\mathbf{u}_A - \mathbf{u})) - \rho^A \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla \cdot (\rho^A (\mathbf{u}_A - \mathbf{u})) + \frac{\rho^A}{\rho} \frac{D\rho}{Dt}.$$ \hspace{1cm} (A.2)

Equating (A.1) and (A.2) yields:

$$\rho \frac{D\phi}{Dt} = -\nabla \cdot \mathbf{q}; \quad \mathbf{q} = \rho^A \mathbf{w}_A.$$ \hspace{1cm} (A.3)

Thus, the diffusion law (A.3) is the direct consequence of: (i) the choice of barycentric velocity as the material velocity, and, (ii) the functional dependence $\rho(\phi)$.
APPENDIX B

ENERGY BALANCE AND THE PVP FOR THE CC MODEL

The total potential energy is

$$\mathcal{P} = \int_V \rho [\omega(\phi, \nabla \phi) - \psi(\mathbf{x})] dV + \int_{\partial V} \gamma(\phi) \partial \mathcal{V}$$

(B.1)

With the gravitational potential $\rho \psi(\mathbf{x})$, the material rate of change of the potential energy is:

$$\frac{D\mathcal{P}}{Dt} = \int_V \rho \left\{ \frac{\partial \omega}{\partial \phi} \frac{D\phi}{Dt} + \frac{\partial \omega}{\partial \nabla \phi} \frac{D\nabla \phi}{Dt} - \nabla \psi \cdot \mathbf{u} \right\} dV + \int_{\partial V} \gamma \frac{D\phi}{Dt} d\mathcal{V}.$$  

(B.2)

The phase field variable is advected with the flow, so that

$$\frac{D\nabla \phi}{Dt} = \frac{\partial \nabla \phi}{\partial t} + \mathbf{u} \cdot \nabla \nabla \phi = \nabla \frac{\partial \phi}{\partial t} + \nabla (\mathbf{u} \cdot \nabla \phi) - \nabla \phi \cdot (\mathbf{u} \nabla) = \nabla \frac{D\phi}{Dt} - \nabla \phi \cdot (\mathbf{u} \nabla).$$  

(B.3)

Upon substituting (B.3) into (B.2), and after standard manipulation, we obtain

$$\frac{D\mathcal{P}}{Dt} = \int_V \rho \left\{ -\kappa \nabla (\nabla \phi) \cdot (\mathbf{u} \nabla) - \rho \nabla \psi \cdot \mathbf{u} + \rho M_0 \frac{D\phi}{Dt} \right\} dV + \int_{\partial V} \chi_0 \frac{D\phi}{Dt} d\mathcal{V},$$

(B.3)

where

$$M_0 = \frac{d\gamma}{d\phi} - \frac{1}{\rho} \nabla \cdot (\kappa \rho \nabla \phi); \quad \chi_0 = \kappa \rho \mathbf{n} \cdot \nabla \phi + \frac{d\gamma}{d\phi}.$$  

(B.4)

Dissipation in the phase-field fluid arises from viscosity and diffusional flux. Viscous dissipation density is the work rate of viscous stresses, with phase-dependent viscosity $\mu(\phi)$. The viscous stress tensor $\tau$ is symmetric and deviatoric (traceless) and will therefore expand no power on volumetric and skew-symmetric portions of the velocity gradient:

$$\tau = 2\mu(\phi) \left[ \frac{1}{2} (\mathbf{u} \nabla + \nabla \mathbf{u}) - \frac{1}{3} (\nabla \cdot \mathbf{u}) \mathbf{l} \right] = \mu(\phi) \nabla \mathbf{u}.$$  

(B.5)

Diffusional dissipation in the bulk is expressed as the power expanded the rate of change of the phase. With the vanishing normal flux on the boundary:

$$-\int_V \rho M \frac{D\phi}{Dt} dV = \int_V M \nabla \cdot \mathbf{q} dV = -\int_V \nabla (M \cdot \mathbf{q}) dV.$$  

(B.6)

The motion of the triple line along the solid boundary is also dissipative. We assume the form analogous to (B.6):
\[
\int_V \frac{\partial \phi}{\partial t} d\Omega V; \quad \chi = 0 \text{ on } A. \quad (B.7)
\]

For isothermal processes, the 2\textsuperscript{nd} law of thermodynamics requires a non-negative total dissipation rate:

\[
\mathcal{D} = \int_V \left[ \tau : (u \nabla) - \rho M \frac{\partial \phi}{\partial t} \right] dV + \int_{\partial V} \frac{\partial \phi}{\partial t} d\partial V \geq 0,
\]

which is satisfied if

\[
q = -B \nabla M; \quad (B > 0) \quad \text{in } V; \quad \frac{\partial \phi}{\partial t} = b \chi; \quad (b > 0) \quad \text{on } S. \quad (B.9)
\]

The energy balance is then written as

\[
\int_{\partial V} t \cdot u d\partial V = \frac{\partial \mathcal{P}}{\partial t} + \frac{\partial \mathcal{K}}{\partial t} + \mathcal{D}; \quad \frac{\partial \mathcal{K}}{\partial t} = \int_V \rho \frac{\partial u}{\partial t} \cdot u dV. \quad (B.10)
\]

Upon substitution of all the components:

\[
\int_V \left[ \rho \nabla \psi - \rho \frac{\partial u}{\partial t} \right] \cdot u dV + \int_{\partial V} t \cdot u d\partial V = \int_V \left[ \tau - \kappa \rho (\nabla \phi \nabla \phi) \right] \cdot (u \nabla) - \rho (M - M_0) \frac{\partial \phi}{\partial t} dV + \int_S \left( \chi + \chi_0 \right) \frac{\partial \phi}{\partial t} d\partial V. \quad (B.11)
\]

Upon interpreting the inertial forces as external body forces, the left-hand side of (B.11) represents power expenditure of external forces on \( u \), while the right-hand side represents power expenditure of generalized internal forces on \( u \nabla \) and \( \frac{\partial \phi}{\partial t} \). Consider small variations \( \delta u \) and \( \delta \left( \frac{\partial \phi}{\partial t} \right) \). The statement of PVP can be formulated as follows: The power of external forces expanded on virtual velocities and fluxes equals the power of internal generalized forces (stresses) on the virtual internal kinematic gradients (deformation rates). Formally, the limit when variations approach zero is taken to preserve only the first order infinitesimals. Otherwise, the variations are arbitrary except for the requirement that they be \textit{kinematically allowable}, i.e., vanish wherever essential boundary condition is prescribed. With the constraint representing compositional compressibility:

\[
\int_V \pi \left( \nabla \cdot \delta u + \frac{\rho'}{\rho} \delta \frac{\partial \phi}{\partial t} \right) dV = 0, \quad (B.12)
\]

the statement of the PVP is:
\[
\int_{V} \rho \mathbf{g} - \rho \frac{\partial \mathbf{u}}{\partial t} \cdot \delta \mathbf{u} dV + \int_{\partial V} \mathbf{t} \cdot \delta \mathbf{u} d\partial V = 0 \quad (B.13)
\]

\[
\int_{V} \left( \Sigma : (\delta \mathbf{uV}) - \rho \left( M - M_0 + \frac{\pi}{\rho^2} \frac{d\rho}{d\phi} \right) \delta \frac{\tilde{D} \phi}{\partial t} \right) dV + \int_{S} (\chi + \chi_0) \delta \frac{\tilde{D} \phi}{\partial t} d\partial V.
\]

where \( \Sigma = \tau - \kappa \phi \nabla \phi \nabla \phi - \pi I \), and \( \mathbf{g} = \nabla \psi \). This yields two independent variational statements:

\[
\int_{V} \rho \left( \frac{\partial \mathbf{u}}{\partial t} - \rho \mathbf{g} - \nabla \Sigma \right) \cdot \delta \mathbf{u} dV = \int_{\partial V} (\mathbf{t} - \mathbf{n} \cdot \Sigma) \cdot \delta \mathbf{u} d\partial V; \quad (B.14)
\]

\[
\int_{V} \rho \left( M - M_0 + \frac{\pi}{\rho^2} \frac{d\rho}{d\phi} \right) \delta \frac{\tilde{D} \phi}{\partial t} dV = \int_{S} (\chi + \chi_0) \delta \frac{\tilde{D} \phi}{\partial t} d\partial V.
\]

The 1st variational statement (B.14) yields the Cauchy equation of motion with standard boundary conditions:

\[
\rho \frac{\partial \mathbf{u}}{\partial t} = \rho \mathbf{g} + \nabla \Sigma \quad \text{in } V; \quad \mathbf{u} = \mathbf{u}^0 \quad \text{on } \partial V. \quad (B.15)
\]

The 2nd variational statement (B.14) implies

\[
M = M_0 - p \frac{d\rho}{\rho^2} \frac{d\phi}{d\phi}; \quad \chi = -\chi_0. \quad (B.16)
\]

After substituting (B.4) into (B.16), and with (B.9) and (2.6) describing the diffusion, this completes the derivation of governing equations (2.16-17).
APPENDIX C

RELATIONSHIP BETWEEN THE TWO DEFINITIONS OF DIFFUSIONAL FLUX

From (2.27):

\[
\frac{D\rho}{Dt} = \frac{\bar{D}\rho}{Dt} - \alpha q \cdot \nabla \rho = -\rho \nabla \cdot (v + \alpha q) - \alpha q \cdot \nabla \rho = -\alpha \nabla \cdot (\rho q). \tag{C.1}
\]

On the other hand, using the linear interpolation (2.28):

\[
\frac{D\rho}{Dt} = \frac{d\rho}{dc} \frac{Dc}{Dt} = (\rho_0^A - \rho_0^B) \frac{Dc}{Dt} = \frac{\rho_0^A - \rho_0^B}{\rho_0^A \rho_0^B} \nabla \cdot (\rho q). \tag{C.2}
\]

Upon equating (C.1) and (C.2):

\[
\frac{Dc}{Dt} = -\nabla \cdot \left( \frac{\rho}{\rho_0^A \rho_0^B} q \right) = -\nabla \cdot J. \tag{C.3}
\]
APPENDIX D

ENERGY BALANCE AND THE PVP FOR THE IC MODEL

We write the total potential energy as:

$$\mathcal{P} = \int_V \rho \left[ \frac{\omega(c, \nabla c)}{\rho} - \psi(x) \right] dV + \int_{\partial V} \gamma(c) d\partial V \quad (D.1)$$

The material rate of change of the potential energy is:

$$\frac{D\mathcal{P}}{Dt} = \int_V \left\{ \rho \left[ \frac{D\omega}{Dt} \left( \frac{\omega}{\rho} \right) - \nabla \psi \cdot \mathbf{v} \right] \right\} dV + \int_{\partial V} \frac{d\gamma}{dc} \frac{Dc}{Dt} d\partial V \quad (D.2)$$

Noting that $c$ is advected with $\mathbf{v}$, c.f. (B.3), the first term in (D.2) is:

$$\int_V \rho \frac{D\omega}{Dt} \left( \frac{\omega}{\rho} \right) dV = \int_V \rho \left[ \frac{\partial(\omega/\rho)}{\partial c} - \nabla \left( \rho \frac{\partial(\omega/\rho)}{\partial c} \right) \right] \frac{Dc}{Dt} dV$$

Then, consider the first two terms on the right hand side of (D.3):

$$\int_V \left[ \rho \frac{\partial(\omega/\rho)}{\partial c} - \nabla \left( \rho \frac{\partial(\omega/\rho)}{\partial c} \right) \right] \frac{Dc}{Dt} dV = \int_V \left( \frac{df}{dc} - \kappa \nabla^2 c \right) \frac{Dc}{Dt} dV - \int_V \omega \frac{D\rho}{Dt} dV \quad (D.4)$$

The 2nd term is the change in potential energy arising from the change in density, which is neglected, consistent with the assumption of incompressibility. The change in total potential energy is obtained as:

$$\frac{D\mathcal{P}}{Dt} = \int_V \left\{ \mathcal{M}_0 \frac{Dc}{Dt} - \kappa (\nabla c \nabla c) : (\nabla \psi - \rho \mathbf{g} \cdot \mathbf{v}) \right\} dV + \int_{\partial V} \mathcal{X}_0 \frac{Dc}{Dt} d\partial V; \quad (D.5)$$

$$\mathcal{M}_0 = \frac{df}{dc} - \kappa \nabla^2 c; \quad \mathcal{X}_0 = \frac{d\gamma}{dc} + \kappa \mathbf{n} \cdot \nabla c.$$  

For incompressible fluids, the viscous stress tensor is:

$$\mathbf{\tau} = \mu(\phi)(\mathbf{u} \nabla + \nabla \mathbf{u}) \quad (D.6)$$

Dissipation is defined in a manner analogous to the definitions in Appendix B:
\[ \mathcal{D} = \int_V \left[ \tau : (u \nabla) - \mathbf{M} \frac{Dc}{Dt} \right] dV - \int_{\partial V} \frac{Dc}{Dt} d\partial V \geq 0; \]  

\[ J = -B \nabla \mathcal{M}; \quad (B > 0) \text{ in } V; \quad \frac{Dc}{Dt} = b \chi; \quad (b > 0) \text{ on } S. \]  

The remainder of the derivation (energy balance and the PVP) is identical to the derivation in Appendix B, except that the condition enforced by the Lagrange multiplier \( \pi \), is now:

\[ \int_V \pi \nabla \cdot u dV = 0. \]  

(D.8)
APPENDIX E
USER’S MANUAL

1. Introduction

This chapter describes the use of the mathematics interface of COMSOL Multiphysics, which is applied for equation-based modeling of the current thesis. COMSOL Multiphysics can be used as an engineering tool for predefined modeling situations, but also as a numerical solver for partial differential equations, for which the user explicitly defines the equation or the associated variational form. COMSOL can either be run from its own Graphical User Interface (GUI) or can be set up to interface with MATLAB. In this study, we used both the GUI and MATLAB interface.

COMSOL Multiphysics provides three types of PDE interfaces for equation-based modeling, distinguished by the equation form used for entering the equations, namely, Coefficient Form, General Form or Weak Form. In the coefficient form, the problem is specified through the setting of individual coefficients in a quite general system of partial differential equations and associated boundary conditions. In the general form, the equation is specified through the definition of a so-called flux function in a what is known as a “conservation-law” formulation of the equation. The weak form is the most general and precise way to specify the equations in a finite element context. In the weak form, the equation is stated by specifying the integrands that occur when writing the equation in a variational form.
2. Implementation with COMSOL Multiphysics

When using COMSOL Multiphysics, there are conceptually six consecutive steps that needs to be taken for each problem:

1. Parameters and variables definitions,
2. Geometry definition,
3. Meshing,
4. Problem specification,
5. Solver definition,
6. Post processing and visualization.

2.1 Parameters and variables definitions

The parameters and variables nodes are located under the Definitions node. Enter values in the parameters table to define parameters used throughout the whole model. Use the Variables node to define expressions as user-defined variables. In the both sections you can enter parameters/variables manually or import them from a text file.

2.2 Geometry definition

The Geometry Node, representing the geometry sequence of the model component, is located under the definitions node. In order to learn how to carry out parameterized solid geometry modeling, work through the Working with Geometry Sequences from the COMSOL
Multiphysics Modeling Guide. Once you have created the geometry as described in the above document, save it to file, since you will need it for the steps below.

### 2.3 Meshing

In this study, we only need to create simple so-called unstructured triangular mesh. Add the Free Triangular node in the mesh section to create an unstructured triangular mesh on domains. You can control the number, size, and distribution of elements by using Size and Distribution nodes.

### 2.4 Problem specification

There are three conceptually different ways in the “expert mode” to specify equations and boundary conditions, namely through the coefficient form, the general form, or the weak form. Here we used General Form and Weak Form to run the simulations.

#### 2.4.1 General Form PDE Interface

Assuming that you are working with a single dependent variable $u$, then a transient problem in general form reads:

$$
e_a \frac{\partial^2 u}{\partial t^2} + d_a \frac{\partial u}{\partial t} + \nabla \cdot \Gamma = f \quad \text{in} \; \Omega, \quad (E.1)$$
where $e_a$, $d_a$, $\Gamma$, and $f$ are the mass coefficient, a damping coefficient or a mass coefficient, the conservative flux vector, and the source term, respectively. The terms $\Gamma$, and $f$ can be functions of the spatial coordinates, the solution $u$, and the space derivatives of $u$. The coefficient $f$ is scalar, whereas $\Gamma$ is the flux vector.

To apply the General PDE interface, we need to define dependent variables for each equation and rewrite the governing equations in the form of the General PDE interface to identify the $e_a$, $d_a$, $\Gamma$, and $f$ coefficients. So, we have:

\[
\text{NS-alt: } \rho \frac{Dv}{Dt} = \rho g + \nabla \cdot \mu(v \nabla + \nabla v) - \nabla \tilde{\pi} + \mathcal{M} \nabla c \\
\Rightarrow \rho \frac{\partial v}{\partial t} - \nabla \cdot \mu(v \nabla + \nabla v) - \nabla \tilde{\pi} I = -\rho \nabla \cdot v + \mathcal{M} \nabla c + \rho g.
\]

The dependent variables are $u$ and $v$. The coefficients are:

\[
e_a = 0; \quad d_a = \rho; \quad \Gamma = - \mu(v \nabla + \nabla v) - \nabla \tilde{\pi} I; \quad f = -\rho \nabla \cdot v + \mathcal{M} \nabla c + \rho g.
\]

(E.3)

The dependent variable is $p$. The coefficients are:

\[
e_a = 0; \quad d_a = 0; \quad \Gamma = 0; \quad f = -\nabla \cdot v.
\]

(E.5)

\[
\text{CH1: } \frac{Dc}{Dt} = \nabla \cdot (B \nabla \mathcal{M}) \Rightarrow \frac{\partial c}{\partial t} - \nabla \cdot (B \nabla \mathcal{M}) = -\nabla \cdot \kappa \nabla c; \\
\text{CH 2: } \mathcal{M} = \frac{df}{dc} - \kappa \nabla^2 c \Rightarrow -\nabla \cdot \kappa \nabla c = \mathcal{M} - \frac{df}{dc}.
\]

(E.6)
The dependent variables are \( c \) and \( M \). The coefficients are:

\[
\begin{align*}
e_{a,CH_1} &= 0 \text{ and } e_{a,CH_2} = 0; \\
d_{a,CH_1} &= 1 \text{ and } d_{a,CH_2} = 0; \\
\Gamma_{CH_1} &= -BH \nabla M \text{ and } \Gamma_{CH_2} = -\kappa \nabla c; \\
f_{CH_1} &= -v \cdot \nabla c \text{ and } f_{CH_2} = M - df / dc.
\end{align*}
\] (E.7)

### 2.4.2 Weak Form PDE Interface

The Weak Form PDE provides a general interface for specifying and solving PDEs in the weak form. To apply the Weak Form, we need to convert the strong form of the governing equations to the weak form using proper test functions:

\( \text{NS-alt:} \quad \int_V \left[ \left( \rho \frac{Dv}{Dt} - \rho g + \nabla \bar{p} \right) \cdot \delta v - \{ \nabla \cdot [\mu (v \nabla + \nabla v)] \} \cdot \delta v - M \nabla c \cdot \delta v \right] dV = 0 \)

\[
\Rightarrow \int_V \left[ \left( \rho \frac{Dv}{Dt} - \rho g + \nabla \bar{p} \right) \cdot \delta v + [\mu (v \nabla + \nabla v)] \cdot (\delta v \nabla) - M \nabla c \cdot \delta v \right] dV
\]

\[
- \int_A \mu n \cdot (v \nabla + \nabla v) \cdot \delta v \partial V = 0;
\] (E.8)

\( \text{IC:} \quad \int_V (\nabla \cdot v) \delta \bar{v} dV = 0; \) (E.9)

\( \text{CH1:} \quad \int_V \left[ \frac{Dc}{Dt} \delta M - \nabla \cdot (B \nabla M) \delta M \right] dV = 0; \)

\[
\Rightarrow \int_V \left[ \frac{Dc}{Dt} \delta M + (B \nabla M) \cdot \nabla \delta M \right] dV - \int_s n \cdot (B \nabla M) \delta M d \partial V = 0; \] (E.10)

\( \text{CH2:} \quad \int_V \left[ M \delta c - \frac{df}{dc} \delta c + \kappa \nabla^2 c \delta c \right] dV = 0 \)

\[
\Rightarrow \int_V \left[ M \delta c - \frac{df}{dc} \delta c - \kappa \nabla c \cdot \nabla \delta c \right] dV + \int_s \kappa (n \cdot \nabla c) \delta c d \partial V = 0; \] (E.11)
Then, we must use the \textit{test} operator to enter the above weak formulations into the COMSOL and to distinguish between functions and the solutions. The test operator must always occur linearly in each weak form contribution. Contributions or terms without any test operator are ignored, while terms nonlinear in the test operator are considered an error. It is also sufficient to let the test operator act directly on the dependent variables and their derivatives. So that, the weak form equations readable by COMSOL are given as:

NS-alt:

\begin{align*}
\text{\textbf{x- direction:}} & \quad -\rho \times (\text{\textit{ut \ test}(u)) + (p - 2 \times \mu \times \text{\textit{ux}}) \times \text{\textit{test}}(\text{\textit{ux}}) + (-\mu \times \text{\textit{uy}} - \mu \times \text{\textit{vx}}) \times \text{\textit{test}}(\text{\textit{uy}}) \\
& \quad -\rho \times (\text{\textit{ux}} \times \text{\textit{u} + \text{\textit{uy}} \times \text{\textit{v}}}) \times \text{\textit{test}}(\text{\textit{u}}) + \mathbf{M} \times \text{\textit{cx}} \times \text{\textit{test}}(\text{\textit{u}}) = 0; \\
\text{\textbf{y- direction:}} & \quad -\rho \times (\text{\textit{vt \ test}(v)) + (-\mu \times \text{\textit{vx}} - \mu \times \text{\textit{uy}}) \times \text{\textit{test}}(\text{\textit{vx}}) + (p - 2 \times \mu \times \text{\textit{vy}}) \times \text{\textit{test}}(\text{\textit{vy}}) \\
& \quad -\rho \times (g + \text{\textit{vx}} \times \text{\textit{u} + \text{\textit{vy}} \times \text{\textit{v}}}) \times \text{\textit{test}}(\text{\textit{v}}) + \mathbf{M} \times \text{\textit{cy}} \times \text{\textit{test}}(\text{\textit{v}}) = 0; \\
\end{align*}

\begin{align*}
\text{\textbf{IC :}} & \quad -u \times \text{\textit{test}}(\text{\textit{px}}) - v \times \text{\textit{test}}(\text{\textit{py}}) = 0; \\
\text{\textbf{CH1:}} & \quad -k \times \text{\textit{cx}} \times \text{\textit{test}}(\text{\textit{c x}}) - k \times \text{\textit{cy}} \times \text{\textit{test}}(\text{\textit{c y}}) + (-32 \times c_0 \times c \times (1 - 3 \times c + 2 \times c^2) + \mathbf{M}) \times \text{\textit{test}}(c) = 0; \\
\text{\textbf{CH2:}} & \quad -B \times \mathbf{M} \times \text{\textit{test}}(\mathbf{M x}) - B \times \mathbf{M} \times \text{\textit{test}}(\mathbf{M y}) - c t \times \text{\textit{test}}(\mathbf{M}) \\
& \quad -(u \times \text{\textit{cx}} + v \times \text{\textit{cy}}) \times \text{\textit{test}}(\mathbf{M}) - c \times (u x + v y) \times \text{\textit{test}}(\mathbf{M}) = 0.
\end{align*}

\begin{equation}
\text{Diff BC: } \int_s \left[ \frac{Dc}{Dt} \delta c + b \kappa \mathbf{n} \cdot \nabla c \delta c + b \frac{d}{dc} \delta c \right] d\partial V = 0. \quad (E.12)
\end{equation}
2.4.3 Specifying and Interpreting Boundary Conditions

The types of boundary conditions used in COMSOL are:

- **Dirichlet** Boundary Condition is used to specify a value of dependent variable on the boundary of the domain. No slip/penetration boundary condition and pressure distribution are imposed by Dirichlet B.C.

- Inlet boundary condition is applied by **Weak Contribution** which contains the test operator acting on any dependent variable in the model.

- The **Flux/Source** boundary condition adds a flux or source \( g \) on the boundary and it is used to impose diffusive boundary condition.

- The **Constraint** boundary condition specifies an expression \( R \) which is constrained to be equal to zero on the selection, \( R = 0 \). The outlet boundary condition is imposed as a constraint.

- The **Zero Flux** boundary condition is applied to vanish normal diffusional flux and normal gradient of phase field variable.

2.5 Solver definition

The Time Dependent solver is selected in the study node since the problem is transient and field variables change over time. The time-stepping is based on the backward differentiation formula which is a family of implicit methods for the numerical integration of ordinary differential equations. The maximum and minimum allowed degrees of the interpolating polynomial of the BDF method are chosen 2 and 1. Nonlinear controller check box is selected for more efficient
time-step control in the BDF method, especially for highly nonlinear problems such as multiphase flows. When nonlinear failures occur the nonlinear controller becomes active and uses a more careful time step regulation. The nonlinear controller acknowledges that the step size for Newton stability might be smaller than the step size for BDF accuracy.

Backward Euler method is applied to perform consistent initialization. The Fraction of initial step value for Backward Euler field is changed from 0.001 (default value) to 1. This value is a dimensionless quantity that determines the size of the time step for the backward Euler method (in terms of the initial step). Adjusting this value can improve the accuracy of the initialization step but can also affect the start-up of some models.

For solving sparse linear systems of equations Direct method based on Gaussian elimination is used with PARDISO-Nested Dissection. Nested dissection is a method of finding an elimination ordering. In Fully Coupled node, constant Newton’s method is selected. And the adaptive mesh consisting of 2nd order triangular elements finest at the interface is added in the solver section. The element size is about 0.1h in all runs. The adaptive mesh density is determined based on the norm of the gradient of the phase field variable.

2.6 Post processing and visualization

The Results branch contains tools and functionality for post-processing and visualizing of the results. 2D Plot and 1D Plot groups are used for illustrating the contours, interface kinetics and pressure gradient.
3. Implementation with MATLAB

The COMSOL Multiphysics can be connected to the MATLAB scripting environment using LiveLink™ for MATLAB. For each operation done in the COMSOL Desktop there is a corresponding command that is entered at the MATLAB prompt, so that a model can be implemented from scratch using the MATLAB scripting environment. Current research has been also implemented by writing a source code in MATLAB M-file using the COMSOL command language. Following commands have been tested and applied:

3.1 The General Utility Functionality

The model object utility methods are available with the ModelUtil object. These methods can be used, for example, to create or remove a new model object, but also to enable the progress bar or list the model object available in the COMSOL server.

3.1.1 Managing the COMSOL model object

Use the method ModelUtil.create to create a new model object in the COMSOL server:

- `model = ModelUtil.create('Model')`

This command creates a model object Model on the COMSOL server and a MATLAB object model that is linked to the model object.
It is possible to have several model objects on the COMSOL server, each with a different name. To access each model object requires different MATLAB variables linked to them and each MATLAB variable must have a different name.

Create a MATLAB variable linked to an existing model object with:

- \[ \text{model} = \\text{ModelUtil.model('Model')} \]

To remove a specific model object, use:

- \[ \text{ModelUtil.remove('Model')} \]

Alternatively remove all the COMSOL objects stored in the COMSOL server with the command:

- \[ \text{ModelUtil.clear} \]

List the names of the model objects available on the COMSOL server with the command:

- \[ \text{list} = \text{ModelUtil.tags} \]

### 3.1.2 Activating The Progress Bar

By default, no progress information is displayed while running COMSOL with MATLAB. To manually enable a progress bar and visualize the progress of operations (such as loading a model, creating a mesh, assembling matrices, or computing the solution), enter the command:

- \[ \text{ModelUtil.showProgress(true)} \]

To deactivate the progress bar, enter:

- \[ \text{ModelUtil.showProgress(false)} \]
3.2 The Geometry Sequence Syntax

Create a geometry sequence using the syntax:

- `model.geom.create(<geomtag>, sdim)`

where `<geomtag>` is a string used to refer to the geometry. The integer `sdim` specifies the space dimension of the geometry and it can be either 0, 1, 2, or 3.

To add an operation to a geometry sequence, use the syntax:

- `model.geom(<geomtag>).feature.create(<ftag>, operation)`

where `<geomtag>` is the string defined when the geometry is created. The string `<ftag>` is used to refer to the operation.

To set the feature property with different values than the default, use the set method:

- `model.geom(<geomtag>).feature(<ftag>).set(property, <value>)`

where `<ftag>` is the string defined when creating the operation.

To build the geometry sequence, enter:

- `model.geom(<geomtag>).run`

Alternatively, to build the geometry sequence up to a given feature `ftag` enter:

- `model.geom(<geomtag>).run(<ftag>)`

3.3 The Meshing Sequence Syntax

Create a meshing sequence by using the syntax:

- `model.mesh.create(<meshtag>, <geomtag>)`
where `<meshtag>` is a string that you use to refer to the sequence. The tag `geomtag` specifies the geometry to use for this mesh node.

To add an operation to a sequence, use the syntax:

- `model.mesh(<meshtag>).feature.create(<ftag>, operation)`

where the string `<ftag>` is a string that you use to refer to the operation.

To set a property to a value in an operation, enter:

- `model.mesh(<meshtag>).feature(<ftag>).set(property, <value>)`

To build the mesh sequence, enter:

- `model.mesh(<meshtag>).run`

To run the mesh node up to a specified feature node `<ftag>`, enter:

- `model.mesh(<meshtag>).run(ftag)`

### 3.4 The Physics Interface Syntax

Create a physics interface instance using the syntax:

- `model.physics.create(<phystag>, physint, <geomtag>)`

where `<phystag>` is a string that identifies the physics interface. Once defined, you can always refer to a physics interface, or any other feature, by its tag. The string `physint` is the constructor name of the physics interface. To get the constructor name, the best way is to create a model using the desired physics interface in the GUI and save the model as an M-file. The string `<geomtag>` refers to the geometry where you want to specify the interface.

To add a feature to a physics interface, use the syntax:

- `model.physics(<phystag>).feature.create(<ftag>, operation)`
where the \texttt{<phystag>} string refers to a physics interface. \texttt{<ftag>} is a string that you use to refer to the operation. To set a property to a value in an operation, enter:

- \texttt{model.physics(<phystag>).feature(<ftag>).set(property, <value>)}

where \texttt{<ftag>} is the string that identifies the feature.

To disable or remove a feature node, use the methods active or remove, respectively.

The command:

- \texttt{model.physics(<phystag>).feature(<ftag>).active(false)}

disables the feature \texttt{<ftag>}.

To activate the feature node, you can set the active method to true:

- \texttt{model.physics(<phystag>).feature(<ftag>).active(true)}

To remove a feature from the model, use the method remove:

- \texttt{model.physics(<phystag>).feature.remove(<ftag>)}

### 3.5 The Material Syntax

In addition to changing material properties directly inside the physics interfaces, materials available in the entire model can also be created. Such a material can be used by all physics interfaces in the model.

Create a material using the syntax:

- \texttt{model.material.create(<mattag>)}

where \texttt{<mattag>} is a string that you use to refer to a material definition.
A Material is a collection of material models, where each material model defines a set of material properties, material functions, and model inputs. To add a material model, use the syntax:

- `model.material(<mattag>).materialmodel.create(<mtag>)`

where `<mattag>` is the string identifying the material defined when creating the material. The string `<mtag>` refers to the material model.

To define material properties for the model, set the property value pairs by entering:

- `model.material(<mattag>).materialmodel(<mtag>).set(property, <value>)`

### 3.6 Adding Global Equations

To add a global equation in the model, use the command:

- `model.physics.create(<odetag>, 'GlobalEquations')`

To define the name of the variable to be solved by the global equation, enter:

- `model.physics(<odetag>).set('name', <idx>, <name>)`

where `<idx>` is the index of the global equation, and `<name>` a string with the name of the variable.

Set the expression `<expr>` of the global equation with:

- `model.physics(<odetag>).set('equation', <idx>, <expr>)`

where `<expr>` is defined as a string variable.

Initial value and initial velocity can be set with the commands:

- `model.physics(<odetag>).set('initialValueU', <idx>, <init>)`
- `model.physics(<odetag>).set('initialValueU_t', <idx>, <init_t>)`
where \(<init>\) and \(<init>_t\) are the initial value expression for the variable and its time derivative respectively.

### 3.7 The Study Node

A study node holds the nodes that define how to solve a model. These nodes are divided into these broad categories:

- Study steps, which determines overall settings suitable for a certain study type,
- Solver sequence, and
- Job configurations for distributed parametric jobs, batch jobs, and cluster computing.

Create a study node by using the syntax:

- `model.study.create(<studytag>);`

where `studytag` is a string that is used to define the study node.

The minimal definition for the study node consists in a study step that define the type of study to use to compute the solution. To add a study step to the study node, use the syntax:

- `model.study(<studytag>).feature.create(<ftag>, operation);`

where `<studytag>` is the string identifying the study node. The string `<ftag>` is a string that is defined to refer to the study step. The string operation is one of the basic study types, such as Stationary, Transient, or Eigenfrequency, and more.

To specify a property value pair for a study step, enter:

- `model.study(<studytag>).feature(<ftag>).set(property, <value>);`

where `<ftag>` is the string identifying the study step.
To generate the default solver sequence associated with the physics solved in the model and compute the solution, run the study node with the command:

- `model.study(<studytag>).run`

### 3.7.1 The Solver Sequence Syntax

If you do not want to use the default solver sequence created by the study node, you can manually create one. To create a solver sequence, enter:

- `model.sol.create(<soltag>);`

where `<soltag>` is a string used to refer to the solver sequence associated to a solution object.

A solver sequence has to be connected to a study node, which is done with the command:

- `model.sol(<soltag>).study(<studytag>);`

where `<studytag>` is the tag of the study you want to associate the solver sequence defined with the tag `<soltag>`.

A solver sequence also requires the definition of these nodes:

- Study Step, where the study and study step is specified for compiling the equations and computing the current solver sequence;

- Dependent Variables, this node handles settings for the computation of dependent variables, such as initial value and scaling settings but also the dependent variables not solved for; and

- Solver node, where the type of solver to use is specified to compute the solution.

Add the nodes to the solver sequence with the command:

- `model.sol(<soltag>).feature.create(<ftag>, operation);`
where `<soltag>` is the string defined when creating the solver sequence. The string `ftag` is a string that is defined to refer to the node, for example, a study step. operation can be 'StudyStep', 'Variables', or 'Stationary'.

To specify a property value pair for a solver feature, enter:

- `model.sol(<soltag>).feature(<ftag>).set(property, <value>);`

where `<soltag>` is a string referring to the solver sequence configuration.

### 3.7.2 Run the Solver Sequence

There are different ways to run the solver sequence:

- run the entire sequence,
- run up to a specified feature, or
- run from a specified feature.

Use the methods `run` or `runAll` to run the entire solver configuration node:

- `model.sol(<soltag>).run;`
- `model.sol(<soltag>).runAll;`

You can also use the method `run(<ftag>)` to run the solver sequence up to the solver feature with the tag `<ftag>`:

- `model.sol(<soltag>).run(<ftag>);`

When you want to continue solving a sequence, use the method `runFrom(<ftag>)` to run the solver configuration from the solver feature with the tag `<ftag>`:

- `model.sol(<soltag>).runFrom(<ftag>)`
3.8 Plot While Solving

To activate Plot While Solving, enter the command:

- \( \text{model.study(<studytag>).feature(<studysteptag>).set('plot', 'on');} \)

where \(<\text{studytag}>\) and \(<\text{studysteptag}>\) refer to the study node and study step, respectively.

Specify the plot group to plot by setting the plot group tag:

- \( \text{model.study(<studytag>).feature(<studysteptag>).set('plotgroup', <ptag>);} \)

Only one plot group can be plotted during a computation. Use the probe feature instead if you need to monitor several variables.

To activate Plot While Solving for a probe plot, enter the command:

- \( \text{model.study(<studytag>).feature(<studysteptag>).set('probesel', seltype);} \)

where \( \text{seltype} \) is the type of probe selection, that can be 'none', 'all', or 'manual'.

In case the probe selection is set to manual you have to specify the list of the probe variable to display. Enter the command:

- \( \text{model.study(<studytag>).feature(<studysteptag>).set('probes', <list>);} \)

where \( \text{<list>} \) is a cell array containing the list of the probe to use.

3.8.1 The Plot Group Syntax

Result plots always appear in plot groups, which are added to the model by the create method:

- \( \text{model.result.create(<pgtag>, sdim);} \)

Select the string \(<\text{pgtag}>\) to identify the plot group and the integer \(sdim\) to set the space dimension (1, 2, or 3) of the group.
To add a plot to a plot group, use the syntax:

- `model.result(<pgtag>).feature.create(<ftag>, plottype);`

where the string `<ftag>` identifies the plot, and the string `plottype` defines its type.

Plots can have different attributes that modify the display. For example, the Deformation attribute deforms the plot according to a vector quantity, the Height Expression attribute introduces 3D height on a 2D table surface plot, and the Filter attribute filters the plot using a logical expression. The type of plot determines which attributes are available. Add an attribute to a plot with the command:

- `model.result(<pgtag>).feature(<ftag>).feature.create(<attrtag>, attrtype);`

where `attrtype` is a string that defines the attribute type.

Another way to plot in a COMSOL Graphics window is to use the `run` method:

- `model.result(<pgtag>).run`
REFERENCES


Feng, X. 2006. Fully discrete finite element approximations of the Navier–Stokes–Cahn–Hilliard


