NTNU – Trondheim
Norwegian University of
Science and Technology

Modeling of Heat Transfer in Two-Phase Flow Using the Level-Set Method

Magnus Aashammer
Gjennestad

Master of Science in Physics and Mathematics
Submission date: June 2013
Supervisor: Jon Andreas Støvneng, IFY
Co-supervisor: Svend Tollak Munkejord, SINTEF Energi AS

Norwegian University of Science and Technology
Department of Physics

An electronic version of this document (navigable and in color) should be available at `http://mastersthesis.aashammer.net/thesis.pdf`.

Abstract

The implementation of a two-phase flow model, developed at SINTEF Energy Research and relying on the level-set method, was extended through the discretization and implementation of an advection-diffusion equation for temperature and a Boussinesq coupling between the temperature and velocity fields. In two-phase flow, both the continuum surface force method and the ghost-fluid method was employed for handling jumps at the interface. Results from simulated cases indicated that the implementation for both single- and two-phase flow with the ghost-fluid method was correct, with second- and first-order convergence, respectively.

Also, a model for phase transition was implemented to allow for vaporization and condensation mass transport between the phases. Results from simulated one-dimensional cases indicated that the implementation of this model was correct in one dimension, with first-order convergence. These results from one dimension and the qualitatively correct results from two dimensions gave reason to believe that the implementation was correct also in two dimensions.

Through the introduction of heat-transport physics in the implementation of the two-phase flow model, this implementation has been developed in direction of performing more detailed simulations that are relevant for natural gas liquefaction processes.

Sammendrag

Implementasjonen av en modell for tofasestrømning, som er utviklet ved SINTEF Energi og basert på level set-metoden, ble utvidet gjennom å diskretisere og implementere en adveksjons-diffusjonsligning for temperaturen og en Boussinesq-kobling mellom temperatur- og hastighetsfeltene. For tofasestrømning ble både continuum surface force-metoden og ghost fluid-metoden brukt for å håndtere grenseflatehopp. Resultatene fra simuleringene som ble utført tydet på at implementasjonen for både en- og tofasestrømning med ghost fluid-metoden var korrekt, med henholdsvis andre- og førsteordens konvergens.

I tillegg ble en modell for faseovergang implementert for å kunne ta høyde for massetransport som følge av fordamping og kondensering mellom fasene. Simuleringsresultater fra endimensjonale problemer tydet på at implementasjonen var rett i en dimensjon, med førsteordens konvergens. Disse resultatene og de kvalitativt rette resultatene fra to dimensjoner ga grunn til å tro at implementasjonen var korrekt også i to dimensjoner.

Gjennom å inkludere varmetransportfysikk i implementasjonen av tofasemodellen har denne blitt utviklet i retning av kunne utføre mer detaljerte simuleringer som er relevante for flytendegjøring av naturgass.

Preface

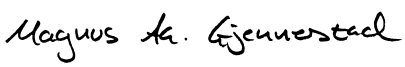
This work will be submitted as a thesis for the degree Master of Science in Physics and Mathematics at The Norwegian University of Science and Technology.

I have had the privilege of conducting this work under the supervision of Chief Scientist Svend Tollak Munkejord at SINTEF Energy Research, whose experience, insight, advice and feedback have all been indispensable to me, and Associate Professor Jon Andreas Støvneng at The Department of Physics at The Norwegian University of Science and Technology, who has been very helpful and has always found the time to answer my questions. I am most grateful to both of them.

Through the supervision by Svend Tollak Munkejord, this thesis is part of the "Enabling Low-Emission LNG Systems" project at SINTEF Energy Research and The Norwegian University of Science and Technology. I want to acknowledge contributions from Statoil, GDF SUEZ E&P Norge and the Petromaks program of the Research Council of Norway (193062/S60).

I wish to thank PhD candidate Karl Yngve Lervåg for good advice and for lending me his Python scripts for parsing `.tec`-files and I must thank my office mate MSc Eskil Aursand for productive discussions, comments and coffee breaks and for challenging me at explaining concepts I may not have understood as well as I initially thought. Also, I am very grateful to my brother Morten and my dear Ida for proofreading and, especially to Ida, for all encouragement and support.

Of course, in spite of all assistance, the responsibility for any errors in this work, both typographic and more severe, is mine.



Magnus Aashammer Gjennestad
Trondheim, June 2013

Nomenclature

Latin letters

$\vec{a}$	Intermediate vector field, see (3.2)	$\mathrm{m\,s^{-2}}$
c_p	Specific heat capacity at constant pressure	$\mathrm{J\,kg^{-1}\,K^{-1}}$
d	Signed Euclidean distance-function, see (3.12)	m
d'	Euclidean distance-function, see (3.13)	m
E_{abs}	Absolute error, see (3.165)	
$E_{\mathrm{ss}}(\varphi)$	Upper bound on simulated $\partial_t \varphi$, see (3.170)	
E_1	Scaled 1-norm error, see (3.166)	
$\vec{f}_{\mathrm{b}}$	Body force density	$\mathrm{N m^{-3}}$
$\vec{f}_{\mathrm{s}}$	Singular interfacial force density	$\mathrm{N m^{-3}}$
$\vec{f}_{\mathrm{sfd}}$	Interfacial force density	$\mathrm{N m^{-3}}$
$\vec{g}$	Gravitational acceleration	$\mathrm{m\,s^{-2}}$
H	Heaviside function	
$\tilde{H}$	Smeared Heaviside function, see (3.57)	
h	Specific enthalpy	$\mathrm{J\,kg^{-1}}$
k	Interface curvature	$\mathrm{m^{-1}}$
L	Fluid domain length	m
$\vec{m}$	Mass flux density	$\mathrm{kg\,m^{-2}\,s^{-1}}$
N	Number of grid cells	1
n	Order of convergence, see (3.167), and time step number	1
$\hat{n}$	Unit normal vector	1
p	Pressure	Pa
Pr	Prandtl number, see (4.2)	1
$\vec{q}$	Heat flux density	$\mathrm{W\,m^{-2}}$
Ra	Rayleigh number, see (4.1) and (4.37)	1

S	Smeared sign function, see (3.17)	1
T	Temperature	K
t	Time	s
$\hat{t}$	Unit tangential vector	1
$\vec{u}$	Velocity	$\mathrm{m\,s^{-1}}$
u	x-component of velocity	$\mathrm{m\,s^{-1}}$
V	Viscous stress tensor	Pa
v	y-component of velocity	$\mathrm{m\,s^{-1}}$
$\vec{w}$	Interface velocity	$\mathrm{m\,s^{-1}}$
$\vec{x}$	Position vector	m
$\hat{x}$	Unit vector in x-direction	1
$\hat{y}$	Unit vector in y-direction	1
$\hat{z}$	Unit vector in z-direction	1

Greek letters

α	Thermal diffusivity	$\mathrm{m^2\,s^{-1}}$
β	Coefficient of thermal expansion	$\mathrm{K^{-1}}$
Γ	Interface between phases	
δ	Dirac and Kronecker delta function	
η	Interface location parameter	
θ	Cell division parameter	1
θ_c	Threshold cell division parameter	1
κ	Thermal conductivity	$\mathrm{W\,m^{-1}\,K^{-1}}$
μ	Dynamic viscosity	Pa s
ν	Kinematic viscosity	$\mathrm{m^2\,s^{-1}}$
ξ	Second viscosity	Pa s
ρ	Density	$\mathrm{kg\,m^{-3}}$
σ	Coefficient of interfacial tension	$\mathrm{N\,m^{-1}}$
τ	Pseudo time and generic integration variable	
ϕ	Level-set function	m

ψ	Stream function	$\mathrm{m}^2\,\mathrm{s}^{-1}$
Ω	Fluid domain	
$\partial\Omega$	Boundary of fluid domain	

Norms and Operators

$[\cdot]$	Difference operator (across interface)
$\nabla\cdot$	Divergence operator
∇	Gradient operator
∇^2	Laplacian operator
$\nabla\times$	Curl operator
$\mathcal{J}(\cdot,\cdot)$	Binary differential operator, see (4.22)
∂_i	Partial derivative operator with respect to spatial coordinate i
∂_n	Partial derivative operator, where the derivative is taken in the direction of some normal vector $\hat{n}$
∂_t	Partial derivative operator with respect to time
$\|\cdot\|_1$	1-norm
$\|\cdot\|_2$	2-norm (Euclidean norm)
$(\cdot)^{\mathrm{T}}$	Transpose

Abbreviations

BC	Boundary condition
CD	Central differences
CFL	Courant–Friedrichs–Lewy
CSF	Continuum surface force
ENO	Essentially non-oscillatory

FUP	First-order upwind
GFM	Ghost-fluid method
LNG	Liquefied natural gas
MAC	Marker-and-cell
ODE	Ordinary differential equation
PDE	Partial differential equation
RK	Runge–Kutta
SI	International system of units
SSP	Strong-stability-preserving
WENO	Weighted essentially non-oscillatory

Contents

1 Introduction

During the past few years, liquefied natural gas (LNG) has become an increasingly important product of the Norwegian petroleum industry. Since the start of extraction in 2007, the Snøhvit field alone has yielded $2.0 \cdot 10^{10}$ standard m^3 oil equivalents of natural gas, $4.65 \cdot 10^9$ standard m^3 oil equivalents of these in 2012 [Norwegian Petroleum Directorate, 2013]. Due to Snøhvit's remote location, natural gas transport by pipeline all the way to the markets was deemed unfeasible as the pipelines would have to be too expensive. Instead, the well stream from Snøhvit is transported through a 145 km pipeline* to the processing facility at Melkøya where it is cooled down to become LNG, which is more efficiently transported by ship [Gisvold, 2004].

The liquefaction process is performed in heat exchangers that can liquefy $11 \cdot 10^6$ kg of natural gas per day [Gisvold, 2004]. A large amount of the greenhouse gas emissions from production, transport and combustion of LNG is incurred at the liquefaction stage [Tamura et al., 2001]. Therefore there is a great potential for enabling a more energy-efficient and less emission-intense LNG life cycle by optimizing the liquefaction stage. Any reduction in energy consumption in the processing would also be beneficial from a financial point of view.

Even though some optimizations can be performed with more conventional engineering approaches, a better and more detailed understanding of the underlying physical behavior of the fluid streams involved in the gas liquefaction is essential. To contribute to such an understanding, SINTEF Energy Research and The Norwegian University of Science and Technology are running the five-year (2009-2014) research program "Enabling Low-Emission LNG Systems" [Løvseth, 2013] with industry partners Statoil and GDF SUEZ E&P Norge.

According to Kunugi [2012], there seems to be a general consensus

*Still, according to Gisvold [2004], the longest of its kind in the world.

that direct numerical simulation is one of the most promising approaches towards clarifying heat transfer characteristics and condensation/boiling phenomena and discussing their mechanisms. Therefore this thesis aims to describe and implement the discretization of a transport equation for temperature into an existing implementation of a two-phase flow model, developed at SINTEF Energy Research, and thus enable the numerical study of heat transfer in two-phase flow. Also, the ambition is to describe and implement a phase-transition model to enable detailed numerical studies of the liquefaction process. Performing such physically realistic LNG simulations, however, is beyond the scope of the thesis.

The flow model relies on the level-set method to keep track of the location of the interface between the phases under consideration. The level-set method was first described by Osher and Sethian [1988] and has since been used in two-phase flow models; both compressible and incompressible [Osher and Fedkiw, 2003]. A review of some of the early advances in the use of this method to study boiling phenomena is given by Dhir [2001]. In this thesis, however, we consider a more recent level-set-based phase-transition model similar to that proposed by Gibou et al. [2007].

1.1 Notation

In this section, we comment on some important points regarding the notation used in this work.

- Vector quantities are distinguished from scalars using arrows. The exception is unit vectors who wear hats. The velocity vector field is therefore denoted by $\vec{u}$, whereas the unit vector in the x-direction is $\hat{x}$ and the scalar pressure field is p.
- We will use derivative notations in full, e.g. $\frac{\partial}{\partial x}$, and short, e.g. ∂_x, forms interchangeably.
- Superscripts $+$ and $-$ most often indicate that the superscripted quantity is associated with one of the two phases. For instance, ρ^- is the density of Phase 1 and ρ^+ is the density of Phase 2.
- When we refer to field values, e.g. of the velocity field, at discrete temporal or spatial points, we use indices enclosed by parenthesis.

Subscripted indices are used for spatial indices and superscripted indices to indicate time step numbers.

- We will sometimes use the repeated index summing convention, also known as Einstein's summing convention, where repeated indices denote an implicit summation over these indices. For example, in this notation, $\partial_i u_i$ is equivalent to $\nabla \cdot \vec{u}$.
- We will apply the gradient operator ∇ to vector fields like $\vec{u}$ with the understanding that the gradient operator is to be applied in a component-by-component fashion, producing a second-order tensor (the Jacobian matrix).
- In analogy to the previous point we will also apply the Laplacian operator ∇^2 to vector fields with the understanding that the operator is to be applied to each component.
- A subscripted g indicates that the subscripted quantity is a ghost value.

1.2 Survey of Thesis

This thesis is divided into five chapters. Chapter 1 is the introduction that you have been reading now. In Chapter 2, we present the mathematical model for the two-phase flow physics whose most significant parts are the Navier–Stokes equations, the temperature transport equation and the phase-transition model. In Chapter 3, we describe the numerical methods used to solve the model from Chapter 2. To test the implementation and the methods in Chapter 3, a few test cases were simulated. These test cases and their results are presented and discussed in Chapter 4. In Chapter 5, we make some concluding remarks and a few suggestions for further work with heat-transfer modeling using the SINTEF-developed implementation of the two-phase flow model.

2 The Mathematical Model for the Two-Phase Flow Physics

The fluid and heat flow physics that is to be simulated is formulated as a mathematical model for two-phase immiscible and incompressible fluid flow and a transport equation for the temperature. The model consists of a set of partial differential equations (PDEs). They are equations of motion and describe the time development of the velocity $\vec{u}$, pressure p and temperature T fields from a known initial configuration subject to a set of known boundary conditions. In this chapter, this model is presented.

In addition, we present a model for phase transition. From the temperature field T we calculate the vaporization and/or condensation mass flux density across the interface and use this to find the correct fluid velocities at the interface and calculate the velocity of the interface itself.

2.1 The Navier–Stokes Equations

From the conservation of mass, one of the most fundamental principles in classical physics, one can derive the continuity equation for a fluid's density ρ,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0. \tag{2.1}$$

In (2.1), $\vec{u}$ is the fluid velocity. The interested reader is referred to Landau and Lifshitz [1987, §1] for a derivation of this equation.

We will assume incompressible flow throughout this work, that is we will assume that

$$\nabla \cdot \vec{u} = 0. \tag{2.2}$$

The continuity equation for the fluid density (2.1) then becomes

$$\frac{\partial \rho}{\partial t} + \vec{u} \cdot \nabla \rho = 0. \tag{2.3}$$

This is an advection equation for the density, indicating that we can have a spatially varying density under the incompressible flow assumption (2.2), but that time evolution of the density field is restricted to advection with the fluid flow. A control surface containing a constant number of fluid particles may not expand or be compressed. Here we will assume that ρ is uniform within each phase.

From the principle of conservation of momentum, one can derive the balance equation for the i-component of a fluid's momentum

$$\rho\left(\frac{\partial u_i}{\partial t}+u_k\frac{\partial u_i}{\partial x_k}\right)=-\frac{\partial p}{\partial x_i}+(f_\mathrm{b})_i+\frac{\partial V_{ik}}{\partial x_k}, \tag{2.4}$$

where p is the pressure, $(f_\mathrm{b})_i$ is the i-component of the body force density* and V is the viscous stress tensor. The derivation of this equation can be found in Landau and Lifshitz [1987, §15].

According to Landau and Lifshitz [1987, §15], the general form of the viscous stress tensor V for an isotropic fluid and small velocity gradients is

$$V_{ik}=\mu\left(\frac{\partial u_i}{\partial x_k}+\frac{\partial u_k}{\partial x_i}-\frac{2}{3}\delta_{ik}\frac{\partial u_\ell}{\partial x_\ell}\right)+\xi\delta_{ik}\frac{\partial u_\ell}{\partial x_\ell}, \tag{2.5}$$

where μ is the dynamic viscosity and ξ is the second viscosity. Since V_{ik} is a linear combination of derivatives $\partial_i u_k$, the fluids we consider may be classified as Newtonian fluids [White, 2008, Section 4.3].

Under assumption of incompressible flow, $\partial_i u_i = 0$ (2.2), the balance equations for all momentum components (2.4) reduce to

$$\rho\left\{\frac{\partial\vec{u}}{\partial t}+(\vec{u}\cdot\nabla)\,\vec{u}\right\}=-\nabla p+\nabla\cdot\mu\left(\{\nabla\vec{u}\}+\{\nabla\vec{u}\}^{\mathrm{T}}\right)+\vec{f}_\mathrm{b}, \tag{2.6}$$

in vector notation. This is commonly known as the incompressible momentum balance equation. The form of (2.6) is that of a continuity equation for the momentum, with the gradient of the pressure field, the body force density and the viscous term $\nabla\cdot\mu\left(\{\nabla\vec{u}\}+\{\nabla\vec{u}\}^{\mathrm{T}}\right)$ as source terms. It is worth mentioning that the form of the viscous term, under the assumption of incompressible flow and constant and uniform viscosity, is identical to

*Body forces is the collective name given to all external forces, such as gravity or electric forces, that are acting on the fluid. In the case where we consider only gravitational external forces, $\vec{f}_\mathrm{b}=\rho\vec{g}$.

a diffusion term for the momentum and that the process of viscous action thus can be viewed as diffusion of momentum.

Equations (2.2) and (2.6) together are often referred to as the incompressible Navier–Stokes equations.

When considering two-phase immiscible flows, we need to model the interface between the two fluids and the interfacial forces acting on it. According to Hansen [2005], the interface can be modeled as a material membrane that flows with the fluids. We will refer to the interface as the set of coordinates

$$\Gamma = \{\vec{x} \mid \vec{x} \text{ is on the interface}\}. \tag{2.7}$$

The interface Γ separates the fluid domain Ω in two regions Ω^+ and Ω^-.

The interfacial forces are modeled by a singular force

$$\vec{f}_{\mathrm{s}}(\vec{x},t) = \int_{\Gamma} \vec{f}_{\mathrm{sfd}}\left(\vec{s},t\right) \delta\left(\vec{x} - \vec{x}_{\mathrm{i}}\left\{\vec{s}\right\}\right) d\vec{s}, \tag{2.8}$$

where $\vec{f}_{\mathrm{sfd}}$ is a force density, δ is the Dirac delta function and $\vec{x}_{\mathrm{i}}$ is a parametrization of the interface. In this thesis, $\vec{f}_{\mathrm{sfd}}$ is modeled by

$$\vec{f}_{\mathrm{sfd}} = \sigma k \hat{n}, \tag{2.9}$$

where σ is the coefficient of interfacial tension, k is the local interface curvature and $\hat{n}$ is a unit normal vector to the interface, pointing into Ω^+.

The singular interfacial force is added as another source term in the momentum balance equation (2.6), so that the equation becomes

$$\rho\left\{\frac{\partial \vec{u}}{\partial t} + \left(\vec{u}\cdot\nabla\right)\vec{u}\right\} = -\nabla p + \nabla\cdot\mu\left(\left\{\nabla\vec{u}\right\} + \left\{\nabla\vec{u}\right\}^{\mathrm{T}}\right) + \vec{f}_{\mathrm{s}} + \vec{f}_{\mathrm{b}}. \tag{2.10}$$

2.2 The Advection-Diffusion Equation for Temperature

In order to model heat transfer, we need a heat-transport equation. In this section, we rely on an argument from Landau and Lifshitz [1987] that reduces the general equation for heat transfer to a transport equation for the temperature and we present this equation.

According to Landau and Lifshitz [1987, §50], we can neglect the temperature dependence of the fluid's viscosity μ, thermal conductivity κ and specific heat capacity c_p if the temperature differences in a fluid are small. We may also suppose that the density is independent of temperature in the equations of motion and regard the fluid as incompressible in the sense that $\partial_i u_i = 0$ if the fluid velocities are small with respect to the speed of sound. With these assumptions, and neglecting viscous heat dissipation, the general equation for heat transfer reduces to an advection-diffusion equation for temperature

$$\frac{\partial T}{\partial t} + (\vec{u} \cdot \nabla)\, T = \frac{1}{\rho c_p} \nabla \cdot (\kappa \nabla T)\,. \tag{2.11}$$

This equation is derived from the general equation for heat transfer by Landau and Lifshitz [1987, §50] and more directly from considering a list of selected physical effects by Griebel et al. [1998, Section 9.2].

The source term on the right hand side of (2.11) is the diffusion term. It models direct molecular transfer of energy from areas with high temperature to areas with low temperature by a heat flux proportional to the temperature difference. This term is independent of the fluid velocity $\vec{u}$ and may lead to heat transfer even when the fluid is macroscopically at rest.

Within each phase, we will assume that κ is uniform and we can rewrite (2.11) as

$$\frac{\partial T}{\partial t} + (\vec{u} \cdot \nabla)\, T = \alpha \nabla^2 T, \tag{2.12}$$

where the thermal diffusivity [Griebel et al., 1998, Section 9.2] is

$$\alpha = \frac{\kappa}{\rho c_p}. \tag{2.13}$$

2.3 The Boussinesq Approximation

Time integration of the temperature transport equation (2.11) alongside the Navier–Stokes equations gives a temperature field that evolves in time subject to diffusion and advection with the fluid flow. What it does not give is any kind of mechanism through which the temperature can couple to the velocity field and thus be a driving force of the flow. Physically

such a coupling would happen through compression and expansion of the fluid as it is heated and cooled. With a few notable exceptions, fluid volumes containing a constant number of fluid particles expand, become less dense and rise when heated, and contract, become more dense and sink when cooled. In our incompressible flow model, however, we do not allow such volumes to expand or contract. Therefore we have to couple the temperature to the velocity field by some other means than a temperature-dependent density, if we want to model temperature-driven flows. In constructing such a coupling, we will follow a widely used strategy and employ the Boussinesq approximation. Based on Griebel et al. [1998, Section 9.1], we state it as follows.

- The fluid density is constant and uniform within each phase, except in the buoyancy part of the body force density $\vec{f}_{\mathrm{b}}$ in the Navier–Stokes equations, where it has a linear dependence on temperature.
- All other fluid properties are constant and uniform, though they also may be discontinuous at the interface.

The linear relationship between density and temperature in the buoyancy part of $\vec{f}_{\mathrm{b}}$ is often expressed as

$$\rho\left(T\right) = \rho_{\infty}\left(1 - \beta\left\{T - T_{\infty}\right\}\right), \tag{2.14}$$

where ρ_{∞} and T_{∞} are a reference density and a reference temperature, respectively, such that $\rho\left(T_{\infty}\right) = \rho_{\infty}$, and β is the coefficient of thermal expansion.

When we consider gravitational body forces only, the body force density becomes $\vec{f}_{\mathrm{b}} = \rho(T)\vec{g}$. The body force term we will use in the Navier–Stokes equations with the Boussinesq coupling is thus

$$\vec{f}_{\mathrm{b}} = \rho_{\infty}\vec{g}\left(1 - \beta\left\{T - T_{\infty}\right\}\right). \tag{2.15}$$

Since $\vec{g}$ appears only in the body force term in the Navier–Stokes equations, we can transfer the temperature dependence of the density in this term to the gravitational acceleration and still expect the same solutions for the $\vec{u}$ and p fields. In other words, we can replace the buoyancy force density $\rho\left(T\right)\vec{g}$ with $\rho_{\infty}\vec{g}\left(T\right)$. The temperature-dependent gravitational acceleration now becomes

$$\vec{g}\left(T\right) = \vec{g}_{\infty}\left(1 - \beta\left\{T - T_{\infty}\right\}\right), \tag{2.16}$$

where $\vec{g}_\infty$ is the usual gravitational acceleration constant. This means that using the Boussinesq approximation presented here is equivalent to introducing a temperature-dependent gravity, where the temperature dependence is given by (2.16).

2.4 A Model for Phase Transition

Consider now the case where we have two phases, liquid and gas, of the same substance and there is mass transport, i.e. vaporization and/or condensation, between the two phases. To simulate this situation, we will use a model very similar to that proposed by Gibou et al. [2007]. In this section, we present this model.

The general idea is to impose that the interface temperature must be equal to the saturation temperature of the liquid T_{sat}. Then we use any resulting interfacial jump in the component of the conductive heat flux density $\vec{q} = -\kappa \nabla T$ that is normal to the interface to find the mass flux density across the interface that is consistent with this jump. Specifically, when more heat flows into a point on the interface than flows out, we assume that this heat has been absorbed in the vaporization of some of the liquid around that point, and vice versa when more heat flows out than in.

To formalize this idea, we first define the interface velocity $\vec{w}$ and the mass flux density across the interface $\vec{m}$. The latter is always normal to the interface, so we can write it in terms of the scalar quantity m as

$$\vec{m} = m\hat{n}. \tag{2.17}$$

We will adopt the convention that the liquid phase occupies Ω^- and the gaseous phase occupies Ω^+. Since $\hat{n}$ is defined as pointing into Ω^+, local mass flux from Ω^- to Ω^+, vaporization, is characterized by a positive m.

Applying mass conservation across the interface gives the following expression for the mass flux density m in terms of the fluid velocities on each side of the interface and the interface velocity,

$$\begin{aligned} m &= \rho^+ \left(\vec{u}^+ \cdot \hat{n} - \vec{w} \cdot \hat{n} \right), && (2.18) \\ &= \rho^- \left(\vec{u}^- \cdot \hat{n} - \vec{w} \cdot \hat{n} \right). && (2.19) \end{aligned}$$

Solving for $\vec{w} \cdot \hat{n}$ gives an expression for the normal component of the interface velocity. Assuming that the tangential component of $\vec{w}$ is equal to the tangential component of $\vec{u}$, we get

$$\vec{w} = \vec{u}^- - \frac{m}{\rho^-}\hat{n}. \tag{2.20}$$

From these equations we see that when there is no mass flux $m = 0$, the velocity field is continuous at the interface and $\vec{w}$ is equal to the fluid velocity at the interface.

Eliminating $\vec{w}$ from (2.19) and (2.20) gives

$$\vec{u}^+ \cdot \hat{n} - \vec{u}^- \cdot \hat{n} = m\left(\frac{1}{\rho^+} - \frac{1}{\rho^-}\right). \tag{2.21}$$

Assuming that $\vec{u} \cdot \hat{t}$ is continuous at the interface, we get

$$\vec{u}^+ - \vec{u}^- = m\left(\frac{1}{\rho^+} - \frac{1}{\rho^-}\right)\hat{n}. \tag{2.22}$$

As the interface condition for temperature, we demand that the temperature in both phases should be equal to the saturation temperature,

$$T^+ = T^- = T_{\text{sat}}. \tag{2.23}$$

Now, we equate the difference in the normal component of the conductive heat flux on each side of the interface with the heat needed to obtain a mass flux density m,

$$m\Delta h = \left(-\kappa^+ \nabla T^+ \cdot \hat{n}\right) - \left(-\kappa^- \nabla T^- \cdot \hat{n}\right). \tag{2.24}$$

Δh is here the specific enthalpy difference (latent heat) associated with the phase change.

The strategy for modeling phase transition is now to evolve the temperature field T in time with (2.11), subject to the interface condition (2.23). Equation (2.24) is used to find the mass flux density m from the temperature field, which is again used to find the interface velocity $\vec{w}$ (2.20) and enforce the interfacial conditions on $\vec{u}$ (2.22).

2.5 The Jump Conditions

When considering two-phase flow, we will assume that all fluid properties such as the viscosity μ, density ρ, thermal conductivity κ and so on, are constant and uniform within each phase. They may, however, have different values in one phase compared to the other. We say that they *jump* at the interface and denote their jumps as

$$[\mu] \quad = \quad \mu^+ - \mu^-, \tag{2.25}$$

for the viscosity and so on for other fluid and flow properties.

So far, the presented equations have been valid in any number of spatial dimensions, depending on the definitions of the vector fields and the spatial derivative operators. In the following, however, we will restrict our attention to two spatial dimensions.

In order to treat the interface sharply, with the method we will later call the ghost-fluid method (GFM), we need to be able to compute the jumps in various quantities. How we do that will depend on whether we include the phase-transition model or not. In the following two subsections we will first state the required jumps for two-phase immiscible flow and then state those for two-phase flow with phase transition.

2.5.1 Two-Phase Immiscible Flow

The required jump conditions for treating the interface sharply in the case of two-phase immiscible flow are

$$\begin{aligned}
[\vec{u}] &= 0, && (2.26)\\
[p] &= 2\,[\mu]\,\hat{n}\cdot\nabla\vec{u}\cdot\hat{n} + \vec{f}_{\text{sfd}}\cdot\vec{n}, && (2.27)\\
[\mu\nabla\vec{u}] &= [\mu]\left\{\left(\hat{n}\cdot\nabla\vec{u}\cdot\hat{n}\right)\hat{n}\hat{n} + \left(\hat{n}\cdot\nabla\vec{u}\cdot\hat{t}\right)\hat{n}\hat{t}\right.\\
&\quad \left. - \left(\hat{n}\cdot\nabla\vec{u}\cdot\hat{t}\right)\hat{t}\hat{n} + \left(\hat{t}\cdot\nabla\vec{u}\cdot\hat{t}\right)\hat{t}\hat{t}\right\}\\
&\quad - \left(t\cdot\vec{f}_{\text{sfd}}\right)\hat{t}\hat{n}, && (2.28)\\
[\nabla p] &= 0, && (2.29)\\
[T] &= 0, && (2.30)\\
[\kappa\nabla T] &= \hat{t}\,[\kappa]\left(\nabla T\cdot\hat{t}\right). && (2.31)
\end{aligned}$$

The jump conditions (2.26) to (2.29) were derived in Lervåg [2008]. The jumps in temperature $[T]$ and heat flux $[\kappa\nabla T]$ at the interface are derived in this section.

According to Landau and Lifshitz [1987, §50], the temperature is continuous at any interface or boundary. The temperature jump then quite simply becomes

$$[T] = 0. \tag{2.32}$$

From the temperature jump (2.32), it immediately follows that

$$[\nabla T] \cdot \hat{t} = 0. \tag{2.33}$$

If this were not the case, the temperature could vary independently along each side of an interface and thus violate (2.32).

Also according to Landau and Lifshitz [1987, §50], the local conductive heat flux out of one phase must be equal to the local conductive heat flux into the other. This amounts to demanding that the component of the heat flux normal to the interface is continuous at the interface,

$$[\kappa\nabla T] \cdot \hat{n} = 0. \tag{2.34}$$

To get an expression for the jump in $\kappa\nabla T$ at the interface, we decompose it into normal and tangential components,

$$[\kappa\nabla T] \quad = \quad \hat{n}\,[\kappa\nabla T] \cdot \hat{n} + \hat{t}\,[\kappa\nabla T] \cdot \hat{t}. \tag{2.35}$$

Using (2.34) and (2.33), we get

$$\begin{aligned} [\kappa\nabla T] &= \hat{t}\,[\kappa\nabla T] \cdot \hat{t}, & (2.36)\\ &= \hat{t}\,[\kappa]\left(\nabla T \cdot \hat{t}\right). & (2.37) \end{aligned}$$

This means that the conductive heat flux tangential to an interface may be discontinuous if the phases have different thermal conductivities.

2.5.2 Two-phase Flow with Phase Transition

When modeling phase transition, it is necessary to modify the jump conditions with respect to those stated in the previous section.

Due to the mass flux across the interface, the velocity is no longer continuous there and the jump condition (2.26) is no longer valid. Instead, we replace it with an expression derived from mass conservation at the interface (2.22).

Because of the heat absorbed or released by the phase transition, the conductive heat flux is not continuous at the interface either, and we replace the interface condition (2.31) with (2.24).

As a jump condition for pressure, we use (13) from Gibou et al. [2007]. In this equation, the jump contribution from the viscous stress has changed with respect to (2.27) because we can no longer assume that $\vec{u}$ is continuous at the interface and there is an additional term to account for the pressure jump from the acceleration of the mass that crosses the interface. The jump conditions on T and ∇p are the same as in the previous section.

Stated together, the jump conditions with phase transition are

$$[\vec{u}] = m\left[\frac{1}{\rho}\right]\hat{n}, \tag{2.38}$$

$$[p] = [\mu\hat{n}\cdot\nabla\vec{u}\cdot\hat{n}] + \vec{f}_{\text{sfd}}\cdot\vec{n} - m^2\left[\frac{1}{\rho}\right], \tag{2.39}$$

$$[\nabla p] = 0, \tag{2.40}$$

$$[T] = 0, \tag{2.41}$$

$$[-\kappa\nabla T\cdot\hat{n}] = m\Delta h. \tag{2.42}$$

Note that we have not stated a jump condition for $\mu\nabla\vec{u}$. This is because, according to Gibou et al. [2007], it is not clear how to treat the jump in viscous stress sharply when modeling phase transitions and we will use the continuum surface force (CSF) method rather than the GFM when computing the viscous stress in these cases.

2.6 Cartesian vs. Cylindrical Coordinates

On regular rectangular grids, we will use the standard Cartesian definitions of the divergence, gradient and Laplacian operators in two-dimensions [Kreyszig, 2006, Appendix 3.4],

$$\nabla\cdot\vec{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}, \tag{2.43}$$

$$\nabla^2 \vec{u} = \hat{x}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) u + \hat{y}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) v, \tag{2.44}$$

$$\nabla T = \hat{x}\frac{\partial T}{\partial x} + \hat{y}\frac{\partial T}{\partial y}, \tag{2.45}$$

$$\nabla^2 T = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}. \tag{2.46}$$

However, when we do three-dimensional simulations under the assumption of rotational symmetry around the y-axis, we use a different grid and different definitions of the above operators. We now let x be the radial coordinate and u be the radial component of the velocity and use the cylindrical-coordinate operators and an axisymmetric grid. With the assumption of rotational symmetry around the y-axis, these operators are [Kreyszig, 2006, Appendix 3.4]

$$\nabla \cdot \vec{u} = \frac{1}{x}\frac{\partial}{\partial x}(xu) + \frac{\partial v}{\partial y}, \tag{2.47}$$

$$\nabla^2 \vec{u} = \hat{x}\left(\frac{\partial^2}{\partial x^2} - \frac{1}{x^2} + \frac{\partial^2}{\partial y^2}\right) u + \hat{y}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) v, \tag{2.48}$$

$$\nabla T = \hat{x}\frac{\partial T}{\partial x} + \hat{y}\frac{\partial T}{\partial y}, \tag{2.49}$$

$$\nabla^2 T = \frac{\partial^2 T}{\partial x^2} + \frac{1}{x}\frac{\partial T}{\partial x} + \frac{\partial^2 T}{\partial y^2}. \tag{2.50}$$

Note that because we have assumed rotational symmetry, there is no contribution from the derivatives with respect to the azimuthal angle in (2.47) to (2.50) and our three-dimensional problem is reduced to a two-dimensional one.

2.7 Boundary Conditions

In order to evolve the $\vec{u}$, p and T fields in time, we need well-posed boundary conditions (BCs) for them. What these are depend, of course, on what kind of physical of properties the boundaries have. In this section, we present the different BCs used in this thesis.

2.7.1 Velocity Boundary Conditions

No-Slip

When we demand that a fluid must be at rest at a boundary, we apply the no-slip BC, and set

$$\vec{u} = 0 \tag{2.51}$$

at that boundary.

Free-Slip

When a boundary is considered a solid surface that cannot supply shear force, we impose that shear stress is zero at the boundary and that there is no flow normal to it. This is expressed mathematically as

$$\vec{u} \cdot \hat{n} = 0, \tag{2.52}$$

$$\partial_n(\vec{u} \cdot \hat{t}) = 0. \tag{2.53}$$

Outflow

At a boundary where the fluids can flow freely out of and into the domain, we impose that the derivative in the normal direction of both velocity components is zero,

$$\partial_n \vec{u} = 0. \tag{2.54}$$

2.7.2 Pressure Boundary Conditions

Dirichlet

With Dirichlet BCs for the pressure, we set

$$p = 0 \tag{2.55}$$

along the boundary.

Neumann

We demand that the normal component of the pressure gradient is zero at the boundary,

$$\partial_n p = 0. \tag{2.56}$$

This choice of BC is not physically founded, but merely an artifact from the projection method. For further details, see Griebel et al. [1998, Section 3.2.3]. With this BC on all boundaries, p is only determined up to the addition of a constant. Therefore we must also supply the value of p at given point.

2.7.3 Temperature Boundary Conditions

Dirichlet

When a boundary can be considered an infinitely large reservoir that can absorb or supply an infinite amount of heat at a constant temperature, we set the temperature equal to a constant value,

$$T = T_s, \tag{2.57}$$

where $s \in \{\text{west}, \text{east}, \text{north}, \text{south}\}$, at that boundary.

Neumann

If we wish to model a boundary with a constant heat flux density q, we set

$$\kappa \partial_n T = q \tag{2.58}$$

at that boundary. With the convention that we take the derivative ∂_n along the outward normal to the boundary, heat flux *into* the fluid domain is associated with a positive q. A thermally insulating boundary is a special case of the Neumann BC with $q = 0$.

2.8 Summary

In this chapter, the mathematical model for the two-phase flow physics was presented. The most significant parts of the model were the incompressible Navier–Stokes equations (2.2) and (2.10) for the velocity $\vec{u}$ and pressure p fields (Section 2.1) and the advection-diffusion equation (2.11) for temperature T (Section 2.2). The body force term in the Navier–Stokes equations was given a temperature dependence (2.15) according to Boussinesq approximation (Section 2.3) to enable simulations with temperature-driven flows and a phase-transition model was presented (Section 2.4) to allow

for vaporization and condensation mass transfer between phases. Jump conditions that are required to treat the interfaces sharply (Section 2.5) and the boundary conditions used in this work were given (Section 2.7).

3 Numerical Methods

In the previous chapter, the mathematical model for the two-phase flow physics was presented. In this chapter, we present the methods used to numerically solve it. The general scheme is similar to those used by e.g. Kang et al. [2000] and Gibou et al. [2007] and can be summarized as follows.

We will use explicit Runge–Kutta (RK) methods to integrate the equations of motion and thus approximate the velocity, pressure and temperature fields discrete at points in time. This can be done if we are able to compute the time derivatives of the velocity and temperature fields at every time step (and RK stage) and the pressure at the next. To this avail we will make use of a variant of Chorin's projection method. This method yields explicit expressions for the required time derivatives and a Poisson equation for the pressure. To compute these time derivatives and solve the Poisson equation, we also need schemes to approximate spatial derivative operators. We will use the weighted essentially non-oscillatory (WENO) scheme for advective operators and central differencing (CD) schemes for Laplacian, gradient and divergence operators. This again requires knowledge of the interface location at each time step and to keep track of that, we employ the level-set method. Also, when computing approximations to the spatial derivatives, we need to treat discontinuities at the interface. For this we use the GFM, the CSF method or a combination of them, depending on the case at hand.

3.1 The Projection Method

In order to approximate the time evolution of the velocity and pressure fields with our explicit RK schemes, we need explicit expressions or solvable equations instead of Navier–Stokes equations (2.2) and (2.10).

In this thesis, we use a variant of the Chorin's projection method [Griebel et al., 1998, Section 3.2.1]. This method gives an explicit expression for

the time derivative of the velocity field $\partial_t \vec{u}^{(n)}$ at time step n and a Poisson equation the pressure $p^{(n+1)}$ at the next time step (or time stage for higher-order RK methods). It is thus explicit in all fields except the pressure where it is implicit. In this section, we present the projection method by giving a brief derivation of it.

We start our derivation by defining an intermediate vector field $\vec{a}^{(n)}$. This contains the entire right hand side of the momentum balance equation (2.10) when solved for $\partial_t \vec{u}$ at time step n, except the contribution from the gradient of the pressure,

$$\partial_t \vec{u}^{(n)} = \vec{a}^{(n)} - \frac{\nabla p^{(n+1)}}{\rho}. \tag{3.1}$$

Explicitly stated then, $\vec{a}^{(n)}$ is

$$\begin{aligned} \vec{a}^{(n)} = &- \left(\vec{u}^{(n)} \cdot \nabla\right) \vec{u}^{(n)} + \nabla \cdot \mu \left(\left\{ \nabla \vec{u}^{(n)} \right\} + \left\{ \nabla \vec{u}^{(n)} \right\}^{\mathrm{T}} \right) \\ &+ \vec{f}_{\mathrm{b}}^{(n)} + \vec{f}_{\mathrm{s}}^{(n)}. \end{aligned} \tag{3.2}$$

This equation for $\vec{a}^{(n)}$ can be directly evaluated, with proper discretization of the spatial differential operators, since the right hand side contains known fields only. However, in order to use (3.1) to get $\partial_t \vec{u}^{(n)}$, we must find the as of yet unknown pressure $p^{(n+1)}$. To do this, we discretize the time derivative in (3.1) using the forward Euler method and get

$$\frac{\vec{u}^{(n+1)} - \vec{u}^{(n)}}{\Delta t} = \vec{a}^{(n)} - \frac{\nabla p^{(n+1)}}{\rho}. \tag{3.3}$$

The Δt is to be understood as the time-step length from time step n to $n+1$. We do not yet know the new velocity field $\vec{u}^{(n+1)}$, but we can demand that it should satisfy (2.2) and be divergence free and then let $p^{(n+1)}$ be whichever pressure field that makes $\nabla \cdot \vec{u}^{(n+1)} = 0$. Applying $\nabla\cdot$ to both sides of (3.3), and demanding that $\nabla \cdot \vec{u}^{(n+1)} = 0$, we get

$$\nabla \cdot \left(\frac{\nabla p^{(n+1)}}{\rho} \right) = \nabla \cdot \vec{a}^{(n)} + \frac{\nabla \cdot \vec{u}^{(n)}}{\Delta t}. \tag{3.4}$$

This is a Poisson equation for the pressure $p^{(n+1)}$, with a known right-hand side. According to Hansen [2005], it can be solved with the appropriate boundary conditions.

In this derivation of Chorin's projection method we have used the pressure to enforce the divergence-freeness of the velocity field. We must therefore expect that the pressure field will adapt such as to make (3.1) give any $\partial_t \vec{u}^{(n)}$ that will make the velocity field divergence free in the next time step. This behavior of the pressure can be seen, from a physical point of view, as an infinite speed of sound. Information about local changes in the velocity field must be instantaneously propagated to the entire domain by the pressure to give a divergence free velocity field. As a consequence, the pressure we calculate in this incompressible flow model will not necessarily correspond to a thermodynamic state variable.

3.1.1 Solving the Poisson Equation

The Poisson equation (3.4) is discretized as described by Kang et al. [2000]. Solving the discretized equation for $p^{(n+1)}$ can be done by solving a linear system of equations corresponding to a symmetric matrix. In this thesis, the implementations of the conjugate gradient method with incomplete Cholesky preconditioning and the biconjugate gradient stabilized method with incomplete LU preconditioning from the PETSc package [Balay et al., 2013] were used. The tolerance of the solvers was set to 10^{-9} for all cases.

The Compatibility Condition

If we integrate both sides of the Poisson equation (3.4) over the entire fluid domain Ω,

$$\int_\Omega \nabla \cdot \left(\frac{\nabla p^{(n+1)}}{\rho} \right) d\vec{x} = \int_\Omega \left(\nabla \cdot \vec{a}^{(n)} + \frac{\nabla \cdot \vec{u}^{(n)}}{\Delta t} \right) d\vec{x}, \tag{3.5}$$

and apply the divergence theorem of Gauss [Kreyszig, 2006, Section 10.7], we get

$$\int_{\partial\Omega} \frac{\nabla p^{(n+1)}}{\rho} \cdot \hat{n} \; d\vec{s} = \int_{\partial\Omega} \left(\vec{a}^{(n)} + \frac{\vec{u}^{(n)}}{\Delta t} \right) \cdot \hat{n} \; d\vec{s}. \tag{3.6}$$

When imposing Neumann BCs on the pressure, we set $\nabla p^{(n+1)} \cdot \hat{n} = 0$ along the boundary. Remembering that we also always set $\vec{a}^{(n)} \cdot \hat{n} = 0$ along the boundary, we see that we must have

$$\int_{\partial\Omega} \vec{u}^{(n)} \cdot \hat{n} \; d\vec{s} \;\; = \;\; 0. \tag{3.7}$$

to satisfy (3.5) when we impose the Neumann BCs. Therefore, (3.7) is our condition for *compatibility* between the right hand side of the Poisson equation (3.4) and the Neumann BCs on p. According to Griebel et al. [1998, Section 3.2.1], compatibility is necessary for the Poisson equation (3.4) to have a solution (determined up to the addition of a constant).

Looking more closely at (3.7), we see that we have compatibility whenever the net flow of fluid across the boundaries and into the domain is zero. Therefore, as long as there is no phase transition, we can follow Griebel et al. [1998] and Kang et al. [2000] and say that (3.7) is satisfied because of the BCs we impose on $\vec{u}$.

When we do have a phase transition, however, these arguments are no longer valid. The easiest way to ensure that the Poisson equation (3.4) still has a solution is to resort to Dirichlet BCs for the pressure and therefore this is what we will do in this thesis.

3.2 The Level-Set Method

When employing finite-difference approximations to the spatial derivative operators in (3.1), (3.2) and (3.4), it is necessary to know the location of the interface between the two phases under consideration. To this avail we need a method to keep track of the interface location at each time step. Several approaches, such as the volume of fluid method [Scardovelli and Zaleski, 1999], the level-set method [Osher and Fedkiw, 2003] and front tracking methods [Tryggvason et al., 2001], can be used. In this work, we use the level-set method and in this section it is briefly presented.

With the level-set method, the interface Γ is implicitly defined as the zero isocontour of the scalar level-set function ϕ,

$$\Gamma(t) = \{\vec{x} \mid \phi(\vec{x}, t) = 0\}. \tag{3.8}$$

The level-set function thus divides the fluid domain Ω in two parts, Ω^+ and Ω^-, where

$$\Omega^+(t) = \{\vec{x} \mid \phi(\vec{x}, t) > 0\}, \tag{3.9}$$

$$\Omega^-(t) = \{\vec{x} \mid \phi(\vec{x}, t) < 0\}. \tag{3.10}$$

We refer to the fluid occupying the region Ω^- as Phase 1 and the fluid occupying Ω^+ as Phase 2.

The temporal evolution of the level-set function is given by taking the total time derivative on both sides of $\phi(\vec{x}, t) = 0$,

$$\frac{\partial \phi}{\partial t} + \vec{w} \cdot \nabla \phi = 0, \tag{3.11}$$

where $\vec{w}$ is the velocity of the interface. In cases without phase transition, $\vec{w}$ is the fluid velocity at the interface. Then (3.11) can be viewed as an advection equation for ϕ and the temporal evolution of the interface is advection with the velocity field $\vec{u}$.

There are many scalar functions that could serve as suitable level-set functions. We will, however, stick to a particularly useful and widely used choice, the signed distance-function

$$d(\vec{x}) = \begin{cases} -d'(\vec{x}) & \text{if } \vec{x} \in \Omega^- \\ d'(\vec{x}) & \text{if } \vec{x} \in \Omega^+ \end{cases}, \tag{3.12}$$

where $d'(\vec{x})$ is the Euclidean distance from the point $\vec{x}$ to its closest point on the interface,

$$d'(\vec{x}) = \min_{\vec{x}' \in \Gamma} \left(\left\| \vec{x} - \vec{x}' \right\|_2 \right). \tag{3.13}$$

The advantages of the level-set method are many. Two of them are the relative ease with which it can be implemented and its ability to handle topological changes such as merging and pinching of drops without any special treatment. Another advantage is the straightforward definition of the normal vector $\hat{n}$ to the interface, pointing from Ω^- into Ω^+ [Lervåg, 2008, Gibou et al., 2007],

$$\hat{n} = \frac{\nabla \phi}{\|\nabla \phi\|_2}, \tag{3.14}$$

and the interface curvature k,

$$k = -\nabla \cdot \hat{n}. \tag{3.15}$$

One of the most significant drawbacks of the level-set method is that it does not conserve mass [Tanguy et al., 2007, Gibou et al., 2007]. That is, the global volume fractions of phases 1 and 2 may change during the time evolution, without this being a result of physical mass transport between the phases. This problem is especially pronounced when the grid is coarse and/or the interface exhibits high curvature.

3.2.1 Reinitialization

In the discretization of some of the spatial derivative operators, we will rely on ϕ being a signed distance-function. However, accumulating numerical errors in the time integration of (3.11) and a non-uniform velocity field $\vec{w}$ will cause ϕ to gradually lose this property during the course of a simulation. To mend this, we employ the reinitialization procedure by Sussmann et al. [1994] at regular time-step intervals. With this procedure, we take our current level-set function ϕ_0 and iterate the reinitialization equation,

$$\frac{\partial \phi}{\partial \tau} + S(\phi_0)\left(\|\nabla \phi\|_2 - 1\right) = 0, \tag{3.16}$$

towards steady-state in pseudo time τ. This eventually produces the steady state solution ϕ^*. In (3.16) S is a smeared sign function,

$$S(\phi) = \frac{\phi}{\sqrt{\phi^2 + (2\Delta x)^2}}. \tag{3.17}$$

Since ϕ^* is the steady-state solution to (3.16), it must satisfy $\|\nabla \phi^*\|_2 = 1$. According to Osher and Fedkiw [2003, Section 2.4] this is a property only signed distance-functions possess. Thus, by replacing ϕ_0 with ϕ^*, the level-set function has regained its signed distance-function property.

3.2.2 Velocity Extrapolation

As showed by Zhao et al. [1996], another measure we can take to make ϕ keep its signed distance-function property upon time integration of (3.11) is to define $\vec{w}$, in points not on the interface, as being equal to the interface velocity at the closest interface point. In practice this is achieved by extrapolating $\vec{w}$ such that it is constant in directions normally away from the interface before computing $\partial_t \phi$ in (3.11). The extrapolation is done by solving the PDE

$$\frac{\partial \vec{w}}{\partial \tau} + S(\phi)\,\hat{n} \cdot \nabla \vec{w} = 0 \tag{3.18}$$

to steady-state in pseudo time τ. Again, $S(\phi)$ is the smeared sign function defined by (3.17). Since the steady-state solution $\vec{w}^*$ must satisfy $\hat{n} \cdot \nabla \vec{w}^* = 0$, we see that $\vec{w}^*$ is indeed constant in direction normal to the interface.

3.3 Time Integration with Strong-Stability-Preserving Runge–Kutta Methods

The equations of motion considered in this thesis, the extrapolation PDEs and the level-set reinitialization equation are spatially discretized such that the time evolution of their solutions can be approximated by solving initial value problems in the form

$$\frac{\partial \varphi}{\partial t} = f\left(\varphi, t\right);\ t \geq 0, \quad \varphi\left(0\right) = \varphi_0. \tag{3.19}$$

Here, $\varphi(t)$ is a placeholder for an arbitrary smooth function of time and φ_0 is its known initial value.

The solution to (3.19) can be approximated at discrete points in time $t^{(n)}$ with e.g. RK methods. We let $\varphi^{(n)}$ approximate φ at time $t^{(n)}$. According to Ketcheson and Robinson [2005], any irreducible s-stage RK method can be uniquely specified by its Butcher table,

$$\begin{array}{c|ccc} c_1 & a_{11} & \dots & a_{1s} \\ \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & \dots & a_{ss} \\ \hline & b_1 & \dots & b_s \end{array},$$

where

$$c_i = \sum_{j=1}^{s} a_{ij},\ \forall i \in \{1, ..., s\}\,. \tag{3.20}$$

The coefficients in this table then specify the RK method as

$$\varphi^{\{i\}} = \varphi^{(n)} + \Delta t \sum_{j=1}^{s} a_{ij} f\left(\varphi^{\{j\}}, t^{(n)} + c_j \Delta t\right), \quad \forall i \in \{1, .., s\}\,, \tag{3.21}$$

$$\varphi^{(n+1)} = \varphi^{(n)} + \Delta t \sum_{j=1}^{s} b_j f\left(\varphi^{\{j\}}, t^{(n)} + c_j \Delta t\right). \tag{3.22}$$

In these equations, $\varphi^{\{i\}}$ denotes the value of φ at the ith RK stage.

If $a_{ij} = 0$ for $j \geq i$, the RK method is explicit. In this work, only explicit RK methods with the strong-stability-preserving (SSP) property* are used. Strong-stability-preserving methods are higher-order RK methods that preserve the stability of the forward Euler method. This property is obtained by ensuring that the RK method can be written as a convex combination for Forward Euler steps. For a more complete discussion of RK methods and the SSP property and its practical importance, the reader is referred to Süli and Mayers [2006, Section 12.5] and Ketcheson and Robinson [2005] respectively.

In this thesis, the forward Euler method with Butcher table

$$\begin{array}{c|c} 0 & \\ \hline & 1 \end{array},$$

the two-stage second-order method (SSP-RK 2()2),

$$\begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array},$$

and the four-stage third-order method (SSP-RK 3()4),

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1 & 1/2 & 1/2 & & \\ 1/2 & 1/6 & 1/6 & 1/6 & \\ \hline & 1/6 & 1/6 & 1/6 & 1/2 \end{array},$$

were used. All equations of motion (2.11), (3.1) and the level-set equation (3.11), were integrated with the SSP-RK 2()2 method and all extrapolation equations and the level-set reinitialization equation (3.16) were integrated with SSP-RK 3()4 where otherwise is not explicitly stated.

3.4 Spatial Discretization

We use equations (2.11), (3.1), (3.2), (3.4) and (3.11) to approximate the pressure and the time derivatives of the level-set function and of the

*Strong-stability-preserving methods are also known as total variation diminishing (TVD) methods in the literature [Ketcheson and Robinson, 2005].

velocity and temperature fields. Also, we have PDEs that we solve to steady-state to do extrapolation and reinitialization. However, in order to do so, we need schemes to estimate the spatial derivative operators in these equations. Most of these schemes can be found in Kang et al. [2000], Gibou et al. [2007] and the references therein. In this section we discuss some of the most important and non-standard ones in detail. For simplicity, we restrict our discussion to rectangular grids.

Throughout this thesis, we work on a staggered marker-and-cell (MAC) grid, where scalar fields such as the pressure p and the temperature T are defined at the cell centers and vector fields such as the velocity $\vec{u}$ are defined at the cell edges. This is illustrated in Figure 3.1.

3.4.1 Discretization of the Advective Operator

When evaluating (3.2) and the right-hand sides of the level-set advection equation (3.11) and the advection-diffusion equation for temperature (2.11), it is necessary to discretize the advective operator $(\vec{u} \cdot \nabla)$. In this work, the weighted essentially non-oscillatory (WENO) scheme is used and in this section we present it and motivate its use in disfavor of the simpler first-order central differences (CD), first-order upwind (FUP) and third-order essentially non-oscillatory (ENO) schemes.

For simplicity and concreteness, we will consider the convective operator in one dimension and use temperature as the example field. That is, we will consider the discretization of

$$u \, \frac{\partial T}{\partial x}, \tag{3.23}$$

since the extension to multiple dimensions can be done in a dimension by dimension fashion. We will also assume uniformly spaced grid nodes. This section follows roughly the arguments from Osher and Fedkiw [2003, Chapter 3].

The Central Differences Scheme

The CD scheme is a natural first naive choice of discretizing the spatial derivative in (3.23),

$$\left(\frac{\partial T}{\partial x} \right)_{(i)} \approx \frac{T_{(i+1)} - T_{(i-1)}}{2\Delta x}. \tag{3.24}$$

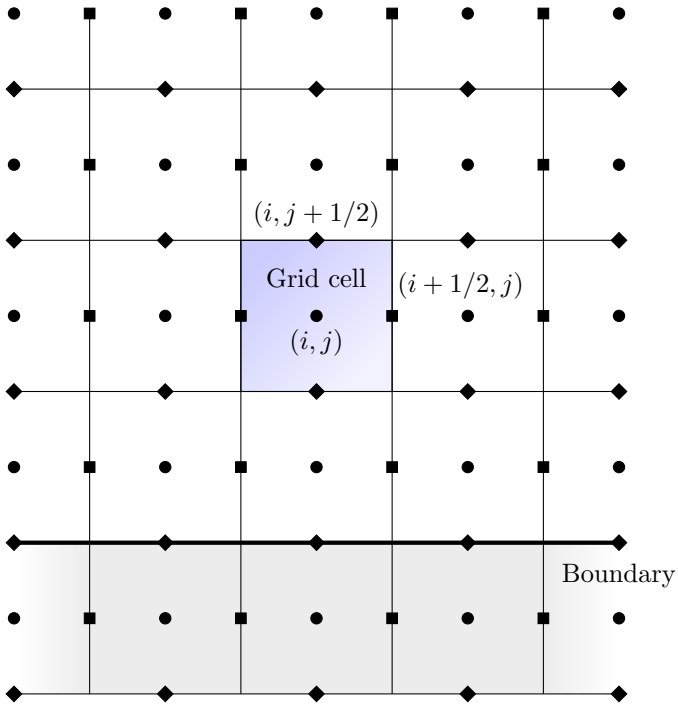


Figure 3.1: Schematic illustration of the MAC grid. The solid lines mark the cell edges. Scalar values are stored at the cell centers (i, j), marked by circles. Vector quantities have their x-components stored at the vertical cell edges $(i+1/2, j)$, marked by squares, and their y-components at the horizontal cell edges $(i, j+1/2)$, marked with diamonds. Boundaries lie along cell edges and boundary conditions are applied by setting the field values in the gray ghost cells.

If the velocity field is interpolated to the cell centers by simple averaging and multiplied with (3.24), we obtain a discretization of (3.23). Unfortunately, however, this scheme will become unstable fail or at best produce spurious oscillations in T, as described by Osher and Fedkiw [2003, Section 3.2] and Lervåg [2008].

The First-Order Upwind Scheme

There are ways to construct better behaving schemes than CD, e.g. following the principle of upwind differencing. If $u_{(i)} > 0$, then values of T move from left to right around (i), advected with the fluid flow. Since information about the temperature flows from left to right, it is then natural to use values located predominantly to the left of (i) when determining what value of T that will be advected to (i) in the next time step, that is, when approximating $\partial_x T$ at (i). We name this approximation to $\partial_x T$, using values of T predominantly to the left, T_x^-. Analogously, if $u_{(i)} \leq 0$ we use values that lie predominantly to the right of (i) to approximate $\partial_x T$ at (i) and name the approximation T_x^+.

One possible choice for the approximations T_x^- and T_x^+ are the backward $D_{(i)}^- T$ and forward $D_{(i)}^+ T$ differences respectively. This results in the FUP scheme,

$$\left(\frac{\partial T}{\partial x}\right)_{(i)} \approx \begin{cases} (T_x^+)_{(i)} = D_{(i)}^+ T = \frac{T_{(i+1)} - T_{(i)}}{\Delta x} & \text{if } u_{(i)} \leq 0 \\ (T_x^-)_{(i)} = D_{(i)}^- T = \frac{T_{(i)} - T_{(i-1)}}{\Delta x} & \text{if } u_{(i)} > 0 \end{cases}. \tag{3.25}$$

This scheme is first-order accurate and produces much better results than CD, according to Osher and Fedkiw [2003, Section 3.2].

The Essentially Non-Oscillatory Scheme

A scheme of higher than first-order accuracy would be an obvious improvement over the FUP scheme. The ENO scheme accomplishes increased accuracy by calculating the approximations T_x^- and T_x^+ in a more elaborate fashion compared to FUP. The general idea is to use divided differences to construct a Newton interpolation polynomial Q to T and then compute the derivative of this polynomial. See e.g. Kreyszig [2006, Section 19.3] for an introduction to Newton interpolation polynomials and divided differences.

The first four divided differences, here indexed 0 to 3, can be expressed as

$$D^0_{(i)}T = T_{(i)}, \tag{3.26}$$

$$D^1_{(i+1/2)}T = \frac{D^0_{(i+1)}T - D^0_{(i)}T}{\Delta x}, \tag{3.27}$$

$$D^2_{(i)}T = \frac{D^1_{(i+1/2)}T - D^1_{(i-1/2)}T}{2\Delta x}, \tag{3.28}$$

$$D^3_{(i+1/2)}T = \frac{D^2_{(i+1)}T - D^2_{(i)}T}{3\Delta x}. \tag{3.29}$$

Note that the zeroth and second divided differences are defined at the cell centers and that the first and third divided differences are defined at the cell edges. We notice also that $D^1_{(i-1/2)}T = D^-_{(i)}T$, that is, the first divided difference at $(i-1/2)$, is the backward difference approximation to $\partial_x T$ at (i) and that $D^1_{(i+1/2)}T = D^+_{(i)}T$ is the forward difference approximation to $\partial_x T$ at (i).

From the divided differences, we can construct a Newton interpolation polynomial Q approximating T,

$$Q(x) = Q_0(x) + Q_1(x) + Q_2(x) + Q_3(x), \tag{3.30}$$

where Q_i is the ith-order contribution. The derivative of Q is

$$Q'(x) = Q'_1(x) + Q'_2(x) + Q'_3(x), \tag{3.31}$$

where we note that the constant zeroth-order contribution has vanished.

The first-order contribution Q_1 is

$$Q_1(x) = \left(D^1_{(k+1/2)}T\right)\left(x - x_{(i)}\right), \tag{3.32}$$

and the derivative is

$$Q'_1(x) = D^1_{(k+1/2)}T. \tag{3.33}$$

We set $k = i-1$ if $u_{(i)} > 0$ and $k = i$ if $u_{(i)} \leq 0$. Observe now that if we ignore the higher order contributions to Q', we recover the first-order FUP scheme. To get the desired third-order accuracy, however, we must include the contributions Q'_2 and Q'_3.

When calculating Q_2 we can choose to make use of either of $D^2_{(k)}T$ and $D^2_{(k+1)}T$. Wanting to construct the smoothest interpolation polynomial, we opt for the one with the smallest absolute value and define

$$C_2 = \begin{cases} D^2_{(k)}T & \text{if } |D^2_{(k)}T| < |D^2_{(k+1)}T| \\ D^2_{(k+1)}T & \text{otherwise} \end{cases}, \tag{3.34}$$

$$\ell = \begin{cases} k-1 & \text{if } |D^2_{(k)}T| < |D^2_{(k+1)}T| \\ k & \text{otherwise} \end{cases}. \tag{3.35}$$

Then

$$Q_2(x) = C_2\left(x - x_{(k)}\right)\left(x - x_{(k+1)}\right), \tag{3.36}$$

and differentiating at $x_{(i)}$ we get

$$Q_2'\left(x_{(i)}\right) = C_2\left(2\left\{i-k\right\}-1\right)\Delta x. \tag{3.37}$$

The third order contribution Q_3 is found from a similar argument. We let

$$C_3 = \begin{cases} D^3_{(\ell+3/2)}T & \text{if } |D^3_{(\ell+3/2)}T| < |D^3_{(\ell+1/2)}T| \\ D^3_{(\ell+1/2)}T & \text{otherwise} \end{cases}. \tag{3.38}$$

Then

$$Q_3(x) = C_3\left(x - x_{(\ell)}\right)\left(x - x_{(\ell+1)}\right)\left(x - x_{(\ell+2)}\right), \tag{3.39}$$

and differentiating at $x_{(i)}$ we get

$$Q_3'\left(x_{(i)}\right) = C_3\left(3\left\{i-\ell\right\}^2 - 6\left\{i-\ell\right\} + 2\right)(\Delta x)^2. \tag{3.40}$$

Using this procedure to find Q' and evaluating it at $x_{(i)}$ to approximate $\partial_x T$ at (i) is the ENO method.

The Weighted Essentially Non-Oscillatory Scheme

It turns out that the ENO scheme produces one of three possible approximations T_x^- or one of another three approximations T_x^+, depending on the

relative magnitudes of the divided differences and the sign of $u_{(i)}$. Defining $\upsilon_j = D^-_{(i-3+j)}T,\ \forall j \in \{1,2,3,4,5\}$ the three possible approximations T^-_x are

$$T^1_x = \frac{\upsilon_1}{3} - \frac{7\upsilon_2}{6} + \frac{11\upsilon_3}{6}, \tag{3.41}$$

$$T^2_x = -\frac{\upsilon_2}{6} + \frac{5\upsilon_3}{6} + \frac{\upsilon_4}{3}, \tag{3.42}$$

$$T^3_x = \frac{\upsilon_3}{3} + \frac{5\upsilon_4}{6} - \frac{\upsilon_5}{6}. \tag{3.43}$$

If we instead define $\upsilon_j = D^-_{(i+3-j)}T,\ \forall j \in \{1,2,3,4,5\}$, (3.41) to (3.43) give the three possible approximations T^+_x.

While the ENO method chooses one of the three possible approximations, the WENO method uses a convex linear combination of them and thus achieves fifth-order accuracy in smooth regions. Introducing the weights $\omega_1, \omega_2, \omega_3 \in [0,1]$ such that $\sum_{j=1}^3 \omega_j = 1$, the WENO scheme choice of T^-_x is

$$T^-_x = \omega_1 T^1_x + \omega_2 T^2_x + \omega_3 T^3_x. \tag{3.44}$$

In the rest of this section we will consider this choice of T^-_x, noting that the following formulas apply also for the choice of T^+_x using the appropriate definition of υ_j.

In Jiang and Shu [1996], it was shown that the WENO scheme produces fifth-order accurate results in smooth regions if the weights are chosen as $\omega_1 = 0.1$, $\omega_2 = 0.6$, and $\omega_3 = 0.3$. Unfortunately, this choice of weights can be very inaccurate in regions where T is not smooth. The key to mending this is to choose the weights so that they are close to the optimal weights when T is smooth around (i) and T^1_x, T^2_x and T^3_x all correspond to smooth interpolation polynomials, and, when T is not smooth around (i), to give less weight to those of T^1_x, T^2_x and T^3_x that correspond to the least smooth interpolation polynomials. This results in third order-accuracy in non-smooth regions.

The non-smoothness measures associated with each of the approximations T^1_x, T^2_x and T^3_x are

$$Z_1 = \frac{13}{12}\left(\upsilon_1 - 2\upsilon_2 + \upsilon_3\right)^2 + \frac{1}{4}\left(\upsilon_1 - 4\upsilon_2 + 3\upsilon_3\right)^2, \tag{3.45}$$

$$Z_2 = \frac{13}{12}\left(v_2 - 2v_3 + v_4\right)^2 + \frac{1}{4}\left(v_2 - v_4\right)^2, \tag{3.46}$$

$$Z_3 = \frac{13}{12}\left(v_3 - 2v_4 + v_5\right)^2 + \frac{1}{4}\left(3v_3 - 4v_4 + v_5\right)^2. \tag{3.47}$$

Next, we define weights that are small when the corresponding measure of non-smoothness is large,

$$\zeta_1 = \frac{0.1}{\left(Z_1 + \epsilon\right)^2}, \tag{3.48}$$

$$\zeta_2 = \frac{0.6}{\left(Z_2 + \epsilon\right)^2}, \tag{3.49}$$

$$\zeta_3 = \frac{0.3}{\left(Z_3 + \epsilon\right)^2}, \tag{3.50}$$

where $\epsilon = 10^{-6}$ to make sure that we never attempt division by zero. These weights are normalized and chosen as the weights in (3.44),

$$\omega_1 = \frac{\zeta_1}{\zeta_1 + \zeta_2 + \zeta_3}, \tag{3.51}$$

$$\omega_2 = \frac{\zeta_2}{\zeta_1 + \zeta_2 + \zeta_3}, \tag{3.52}$$

$$\omega_3 = \frac{\zeta_3}{\zeta_1 + \zeta_2 + \zeta_3}. \tag{3.53}$$

The WENO scheme is thus preferable to the ENO scheme and, by extension, to the other schemes considered in this section because of its fifth-order accuracy in smooth regions. It also has another advantage over the ENO scheme in that a smaller computational effort is needed for each approximation to $\partial_x T$. This was demonstrated e.g. by Lervåg [2008] in his Table 3.1.

3.4.2 Interface-Capturing Methods

In the spatial discretizations of the Laplacian operator (viscous term) in (3.2), the Laplacian (diffusive term) in (2.11), the Laplacian in (3.4) and the gradient operator in the projection equation (3.1) for two-phase simulations, we need a way to handle interfacial discontinuities in the fields we apply these operators to and/or their spatial derivatives.

In this section, we present two strategies for handling the interfacial discontinuities in the diffusive term in the advection-diffusion equation (2.11). These are the continuum surface force (CSF) method and the ghost-fluid method (GFM). The GFM scheme is discussed twice, once for two-phase immiscible flow and once for two-phase flow with phase transition.

For a presentation of these methods in the context of the viscous term in (3.2), the Poisson equation (3.4) and the projection equation (3.1), the interested reader is referred to Kang et al. [2000] and Osher and Fedkiw [2003].

The Continuum Surface Force Method

The CSF method handles discontinuities by smearing out the discontinuous properties around the interface in a controlled way.

Consider the discretization of the term

$$\frac{1}{\rho c_p}\frac{\partial}{\partial x}\left(\kappa\frac{\partial T}{\partial x}\right). \tag{3.54}$$

The discretization of the y-term will be completely analogous to this. We discretize it at the grid point (i,j) using a standard CD scheme

$$\left\{\frac{1}{\rho c_p}\frac{\partial}{\partial x}\left(\kappa\frac{\partial T}{\partial x}\right)\right\}_{(i,j)} \approx 2\alpha\left(\phi_{(i,j)}\right)\frac{\frac{T_{(i+1,j)}-T_{(i,j)}}{\Delta x_{(i)}} - \frac{T_{(i,j)}-T_{(i-1,j)}}{\Delta x_{(i-1)}}}{\Delta x_{(i)}+\Delta x_{(i-1)}}, \tag{3.55}$$

where $\Delta x_{(i)}$ is the distance between grid points (i,j) and $(i+1,j)$ and α is smeared out in the vicinity of the interface. The smearing of α is given by

$$\alpha(\phi) = \alpha^- + (\alpha^+ - \alpha^-)\tilde{H}(\phi), \tag{3.56}$$

where $\tilde{H}(\phi)$ refers to a smeared Heaviside function, which guarantees that the thermal diffusivity α is smooth,

$$\tilde{H}(\phi) = \begin{cases} 0 & \text{if } \phi < -\epsilon \\ \frac{1}{2} + \frac{\phi}{2\epsilon} + \frac{1}{2\pi}\sin\left(\frac{\pi\phi}{\epsilon}\right) & \text{if } -\epsilon \le \phi \le \epsilon \\ 1 & \text{if } \epsilon < \phi \end{cases}. \tag{3.57}$$

The smoothness of $\tilde{H}(\phi)$ is illustrated in Figure 3.2.

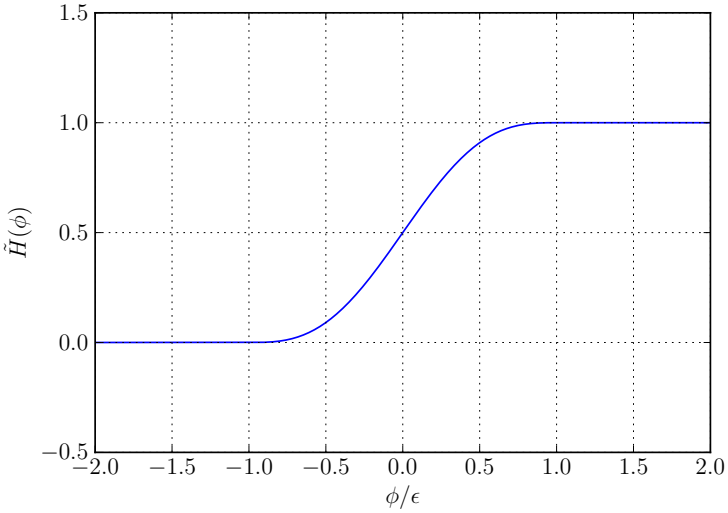


Figure 3.2: A plot of the smeared Heaviside function (3.57).

The Ghost-Fluid Method for Immiscible Two-Phase Flow

It would be advantageous to have a way of sharply discretizing terms like (3.54) without numerical smearing of κ. For this purpose, we will use the GFM. Again, we will consider only the discretization of the x-term, as discretization of the y-term will be completely analogous to this. We discretize the derivative in (3.54) at the grid point (i, j) follows,

$$\left(\frac{\partial}{\partial x}\left\{\kappa\frac{\partial T}{\partial x}\right\}\right)_{(i,j)} \approx 2\frac{(\kappa\partial_x T)_{(i+1/2,j)} - (\kappa\partial_x T)_{(i-1/2,j)}}{\Delta x_{(i)} + \Delta x_{(i-1)}}. \tag{3.58}$$

Now, if $\phi_{(i-1,j)}$, $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ are either all positive or all negative, the discretization of the numerator above is straightforward. If all are positive, then

$$\left(\kappa\frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx \kappa^+\frac{T_{(i+1,j)} - T_{(i,j)}}{\Delta x_{(i)}}, \tag{3.59}$$

$$\left(\kappa\frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx \kappa^+\frac{T_{(i,j)} - T_{(i-1,j)}}{\Delta x_{(i-1)}}. \tag{3.60}$$

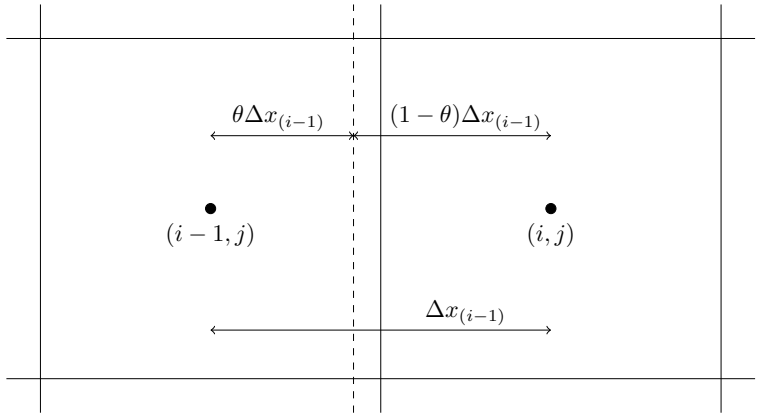


Figure 3.3: An illustration of the sub-cell resolution of the interface location. The dashed line shows the interface position.

For the case where $\phi_{(i-1,j)}$, $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ are all negative, we replace κ^+ with κ^- in the above equations,

$$\left(\kappa\frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx \kappa^- \frac{T_{(i+1,j)} - T_{(i,j)}}{\Delta x_{(i)}}, \tag{3.61}$$

$$\left(\kappa\frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx \kappa^- \frac{T_{(i,j)} - T_{(i-1,j)}}{\Delta x_{(i-1)}}. \tag{3.62}$$

If $\phi_{(i-1,j)}$ and/or $\phi_{(i+1,j)}$ have a different sign from $\phi_{(i,j)}$, we must take the presence of an interface into account.

Consider first the case where there is an interface located between the grid points $(i-1,j)$ and (i,j), that is when $\phi_{(i-1,j)}$ and $\phi_{(i,j)}$ have opposite signs. Assume also that $\phi_{(i-1,j)} \leq 0$ and $\phi_{(i,j)} > 0$. Relying on the distance-function property of the level-set function, we will assume that the interface is located at

$$x_{\mathrm{i}} = x_{(i-1)} + \theta\Delta x_{(i-1)}, \tag{3.63}$$

where θ is given by

$$\theta = \frac{\left|\phi_{(i-1,j)}\right|}{\left|\phi_{(i-1,j)}\right| + \left|\phi_{(i,j)}\right|}. \tag{3.64}$$

In this way, we resolve the position of the interface on a scale smaller than the cell sizes, see Figure 3.3.

To estimate the temperature at the interface T_i, we discretize the jump condition (2.31) at the interface as follows and call it J_i.

$$J_\mathrm{i} \approx \kappa^+ \frac{T_{(i,j)} - T_\mathrm{i}}{(1-\theta)\,\Delta x_{(i-1)}} - \kappa^- \frac{T_\mathrm{i} - T_{(i-1,j)}}{\theta \Delta x_{(i-1)}}. \tag{3.65}$$

The jump condition itself is interpolated to the interface as

$$J_\mathrm{i} \approx (1-\theta)\, J_{(i-1,j)} + \theta J_{(i,j)}. \tag{3.66}$$

Equation (3.65) can be solved for the interface temperature,

$$T_\mathrm{i} \approx \frac{\kappa^+ T_{(i,j)}\theta + \kappa^- T_{(i-1,j)}\,(1-\theta) - J_\mathrm{i}\theta\,(1-\theta)\,\Delta x_{(i-1)}}{\kappa^+\theta + \kappa^-\,(1-\theta)}, \tag{3.67}$$

and this expression for T_i can be used to discretize $\kappa\ \partial_x T$ at $(i-1/2, j)$,

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx \kappa^+ \frac{T_{(i,j)} - T_\mathrm{i}}{(1-\theta)\Delta x_{(i-1)}}, \tag{3.68}$$

$$\approx \kappa' \frac{T_{(i,j)} - T_{(i-1,j)}}{\Delta x_{(i-1)}} + \kappa' \frac{J_\mathrm{i}\theta}{\kappa^-}. \tag{3.69}$$

In these equations we have defined the effective thermal conductivity,

$$\kappa' = \frac{\kappa^+\kappa^-}{\kappa^+\theta + \kappa^-\,(1-\theta)}. \tag{3.70}$$

By a similar argument one finds, for the case where $\phi_{(i-1,j)} > 0$ and $\phi_{(i,j)} \le 0$,

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx \kappa^- \frac{T_{(i,j)} - T_\mathrm{i}}{(1-\theta)\Delta x_{(i-1)}}, \tag{3.71}$$

$$\approx \kappa'' \frac{T_{(i,j)} - T_{(i-1,j)}}{\Delta x_{(i-1)}} + \kappa'' \frac{J_\mathrm{i}\theta}{\kappa^+}, \tag{3.72}$$

where the effective thermal conductivity is

$$\kappa'' = \frac{\kappa^+\kappa^-}{\kappa^-\theta + \kappa^+\,(1-\theta)}. \tag{3.73}$$

Now we can insert the appropriate one of (3.69) and (3.72) for the discretization of $\kappa\ \partial_x T$ at $(i-1/2, j)$ to evaluate (3.58) in the case where there is an interface between $(i-1, j)$, and (i, j).

Next, consider the situations where there is an interface located between (i, j) and $(i+1, j)$. Then we can interpolate the jump condition to the interface as

$$J_{\mathrm{i}} \approx (1-\theta) J_{(i+1,j)} + \theta J_{(i,j)}, \tag{3.74}$$

where θ is now given by

$$\theta = \frac{\left|\phi_{(i+1,j)}\right|}{\left|\phi_{(i+1,j)}\right| + \left|\phi_{(i,j)}\right|}. \tag{3.75}$$

For the case where $\phi_{(i,j)} > 0$ and $\phi_{(i+1,j)} \leq 0$, we use

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx \kappa^{+} \frac{T_{\mathrm{i}} - T_{(i,j)}}{(1-\theta)\Delta x_{(i)}}, \tag{3.76}$$

$$\approx \kappa' \frac{T_{(i+1,j)} - T_{(i,j)}}{\Delta x_{(i)}} + \kappa' \frac{J_{\mathrm{i}} \theta}{\kappa^{-}}, \tag{3.77}$$

in the evaluation of (3.58). And for the case where $\phi_{(i,j)} \leq 0$ and $\phi_{(i+1,j)} > 0$, we use

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx \kappa^{-} \frac{T_{\mathrm{i}} - T_{(i,j)}}{(1-\theta)\Delta x_{(i)}}, \tag{3.78}$$

$$\approx \kappa'' \frac{T_{(i+1,j)} - T_{(i,j)}}{\Delta x_{(i)}} + \kappa'' \frac{J_{\mathrm{i}} \theta}{\kappa^{+}}, \tag{3.79}$$

in (3.58).

The Ghost-Fluid Method for Two-Phase Flow with Phase Transition

When simulating two-phase flow with phase transition, we replace the interface condition (2.31) on the temperatures with the condition (2.23); that the interface temperature should be equal to the saturation temperature T_{sat}. This leads to a slightly different GFM treatment of the Laplacian operator in (2.11), which we will proceed to describe in this section. This new scheme is similar to that described by Gibou et al. [2002] for treating Stefan problems.

We choose the discretization of the Laplacian operator like in the immiscible two-phase flow case, according to (3.58). Again, if $\phi_{(i-1,j)}$, $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ are all positive, we plug (3.59) and (3.60) into (3.58) and, as before, if $\phi_{(i-1,j)}$, $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ are all negative we use (3.61) and (3.62).

However, when $\phi_{(i-1,j)}$ and $\phi_{(i,j)}$ do not have the same sign, we need to approximate $\partial_x T$ at $(i-1/2, j)$ in a way that enforces the condition that the interface temperature is equal to T_{sat}. First, define the cell division parameter θ,

$$\theta = \frac{\left|\phi_{(i,j)}\right|}{\left|\phi_{(i-1,j)}\right| + \left|\phi_{(i,j)}\right|}, \tag{3.80}$$

and consider the case when $\phi_{(i,j)} > 0$. Then, the straightforward way to approximate $\partial_x T$ at $(i-1/2, j)$ is

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx \kappa^{+} \frac{T_{(i,j)} - T_{\text{sat}}}{\theta \Delta x_{(i-1)}}. \tag{3.81}$$

According to Gibou et al. [2002], this scheme becomes unstable when θ is small, because we end up subtracting two numbers of similar magnitude and dividing the result by a small number. Therefore, we use it only when $\theta > \theta_{\text{c}}$, for some specified constant $\theta_{\text{c}} \in \langle 0, 1 \rangle$. To get good approximations to $\kappa\, \partial_x T$, we should choose θ_{c} reasonably small.

If instead $\theta \leq \theta_{\text{c}}$ and $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ have the same sign, we choose

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx \kappa^{+} \frac{T_{(i+1,j)} - T_{\text{sat}}}{\theta \Delta x_{(i-1)} + \Delta x_{(i)}}, \tag{3.82}$$

and thus avoid the problem that occurs when θ is small. From this equation it is evident that the cost of choosing a finite θ_{c} to keep this scheme stable is a smearing out of the approximated $\kappa\, \partial_x T$ at the interface.

When $\theta \leq \theta_{\text{c}}$ and $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ do not have the same sign, we give up and set

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i-1/2,j)} \approx 0. \tag{3.83}$$

According to Gibou et al. [2002], this is acceptable since we now have two interfaces close together, both have temperatures T_{sat} and the temperature derivative must therefore be small.

Similarly, we need to account for the interface condition also when $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ have different signs. Assume for now that $\phi_{(i,j)} > 0$ and define

$$\theta = \frac{\left|\phi_{(i,j)}\right|}{\left|\phi_{(i,j)}\right| + \left|\phi_{(i+1,j)}\right|}. \tag{3.84}$$

When $\theta > \theta_c$, we approximate

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx \kappa^+ \frac{T_{\text{sat}} - T_{(i,j)}}{\theta \Delta x_{(i)}}, \tag{3.85}$$

and when $\theta \leq \theta_c$ and $\phi_{(i-1,j)}$ and $\phi_{(i,j)}$ have the same sign, we use

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx \kappa^+ \frac{T_{\text{sat}} - T_{(i-1,j)}}{\theta \Delta x_{(i)} + \Delta x_{(i-1,j)}}. \tag{3.86}$$

Again, when $\theta \leq \theta_c$ and $\phi_{(i-1,j)}$ and $\phi_{(i,j)}$ have different signs we give up and set

$$\left(\kappa \frac{\partial T}{\partial x}\right)_{(i+1/2,j)} \approx 0. \tag{3.87}$$

In the case where $\phi_{(i,j)} \leq 0$, we use the same procedure, but replace κ^+ with κ^- in the equations above.

3.5 Numerical Strategy for Handling Phase Transition

The phase-transition model was presented in Section 2.4. In this section, we give our strategy for handling it numerically. This strategy is similar to that by Gibou et al. [2007] and can be roughly divided into three parts. First, we have to calculate the mass flux density m from the temperature field T. Second, we need to enforce the interface condition (2.38) on the velocities and third, we must calculate the interface velocity in a way that is consistent with mass conservation.

Note that in this work, we do not use the Boussinesq coupling when simulating phase-transition cases. In these cases, the temperature may drive the flow by initiating mass transfer between the phases, but not through a temperature-dependent buoyancy force.

3.5.1 Computing the Mass Flux Density from the Temperature Field

Our first challenge when incorporating phase transitions in our model is to accurately and robustly approximate $\hat{n} \cdot \nabla T$ in cell centers on both sides of the interface so that we can use (2.42) to get the mass flux density m.

We will use the following scheme to obtain approximations $[T_x, T_y]$ to ∇T at every cell center and then project them onto the interface normal vector $\hat{n}$ and extrapolate the projected values across the interface in a constant fashion to compute m. This is similar to the scheme presented by Gibou et al. [2002] for Stefan problems.

If $|\phi_{(i-1,j)}| < |\phi_{(i+1,j)}|$, the point $(i-1, j)$ lies closer to the interface than $(i+1, j)$. Then, if $\phi_{(i-1,j)}$ and $\phi_{(i,j)}$ are either both positive or both negative, we discretize according to

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = \frac{T_{(i,j)} - T_{(i-1,j)}}{\Delta x_{(i-1)}}. \tag{3.88}$$

On the other hand, if $\phi_{(i-1,j)}$ and $\phi_{(i,j)}$ have different signs, there is an interface between $(i-1, j)$ and (i, j) and we need to use that the interface temperature is equal to the saturation temperature (2.23), when approximating $\partial_x T$. Let

$$\theta = \frac{\left|\phi_{(i,j)}\right|}{\left|\phi_{(i-1,j)}\right| + \left|\phi_{(i,j)}\right|}. \tag{3.89}$$

Then, if $\theta > \theta_c$ for some specified constant $\theta_c \in \langle 0, 1\rangle$, we use the approximation

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = \frac{T_{(i,j)} - T_{\text{sat}}}{\theta \Delta x_{(i-1)}}. \tag{3.90}$$

By the same arguments as in Section 3.4.2, this scheme becomes unstable when θ is small. This is precisely why we use it only when $\theta > \theta_c$. If instead $\theta \leq \theta_c$ and $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ have the same sign, we choose

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = \frac{T_{(i+1,j)} - T_{\text{sat}}}{\theta \Delta x_{(i-1)} + \Delta x_{(i)}}. \tag{3.91}$$

In the case where $\theta \leq \theta_c$ and $\phi_{(i,j)}$ has a different sign than $\phi_{(i+1,j)}$, we set

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = 0. \tag{3.92}$$

If instead $|\phi_{(i-1,j)}| \geq |\phi_{(i+1,j)}|$, then the point $(i+1,j)$ lies closer to the interface than $(i-1,j)$ and we follow an analogous procedure to approximate $\partial_x T$ at (i,j). If $\phi_{(i+1,j)}$ and $\phi_{(i,j)}$ are both positive or both negative, we discretize according to

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = \frac{T_{(i+1,j)} - T_{(i,j)}}{\Delta x_{(i)}}. \tag{3.93}$$

On the other hand, if $\phi_{(i,j)}$ and $\phi_{(i+1,j)}$ have different signs, there is an interface between (i,j) and $(i+1,j)$ and we need to enforce the interface condition (2.23). Let

$$\theta = \frac{\left|\phi_{(i,j)}\right|}{\left|\phi_{(i,j)}\right| + \left|\phi_{(i+1,j)}\right|}. \tag{3.94}$$

Then, if $\theta > \theta_{\mathrm{c}}$ for the specified constant $\theta_{\mathrm{c}} \in \langle 0,1\rangle$, we use the approximation

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = \frac{T_{\mathrm{sat}} - T_{(i,j)}}{\theta \Delta x_{(i)}}. \tag{3.95}$$

If instead $\theta \leq \theta_{\mathrm{c}}$ and $\phi_{(i,j)}$ and $\phi_{(i-1,j)}$ have the same sign, we choose

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = \frac{T_{\mathrm{sat}} - T_{(i-1,j)}}{\theta \Delta x_{(i)} + \Delta x_{(i-1)}}. \tag{3.96}$$

In the case where $\theta \leq \theta_{\mathrm{c}}$ and $\phi_{(i,j)}$ has a different sign than $\phi_{(i-1,j)}$, we give up and set

$$\left(\frac{\partial T}{\partial x}\right)_{(i,j)} \approx T_x = 0. \tag{3.97}$$

The scheme for approximating $\partial_y T$ is completely analogous to that presented for $\partial_x T$ and will not be discussed in further detail.

Assume now that we have our approximations T_x and T_y in a band around the interface. Then we compute approximations T_n to $\hat{n} \cdot \nabla T$ by taking

$$T_n = \hat{n} \cdot [T_x, T_y]. \tag{3.98}$$

To compute the jump (2.42) we need to extrapolate T_n from Phase 1 into Phase 2 and vice versa to have liquid values T_n^- and gas values T_n^+

at each grid point. We perform the extrapolation by employing the PDE approach presented by Aslam [2003] and solve the equations

$$\frac{\partial T_n^+}{\partial \tau} = -\{1 - H(\phi)\}\,\hat{n} \cdot \nabla T_n^+, \tag{3.99}$$

$$\frac{\partial T_n^-}{\partial \tau} = -H(\phi)\,\hat{n} \cdot \nabla T_n^-, \tag{3.100}$$

to steady-state in pseudo time τ. In these equations, H is the Heaviside function.

Having extrapolated T_n^+ into Ω^- and T_n^- into Ω^+, we can reliably calculate m according to the following approximation to (2.42),

$$m \approx \frac{-\left(\kappa^+ T_n^+ - \kappa^- T_n^-\right)}{\Delta h}. \tag{3.101}$$

3.5.2 Enforcing the Interface Conditions on the Velocity

In order to enforce the jump in fluid velocity at the interface (2.38), we define the liquid and gaseous velocities $\vec{u}^-$ and $\vec{u}^+$ point by point in terms of the physical fluid velocities $\vec{u}$ in their respective phases and the ghost values $\vec{u}_{\mathrm{g}}^-$ and $\vec{u}_{\mathrm{g}}^+$,

$$\vec{u}^+ = \begin{cases} \vec{u} & \text{if } \phi > 0 \\ \vec{u}_{\mathrm{g}}^+ & \text{if } \phi \leq 0 \end{cases}, \tag{3.102}$$

$$\vec{u}^- = \begin{cases} \vec{u}_{\mathrm{g}}^- & \text{if } \phi > 0 \\ \vec{u} & \text{if } \phi \leq 0 \end{cases}. \tag{3.103}$$

The values of the scalar level-set function is obtained at the velocity points by linear interpolation. The ghost values are defined as

$$\vec{u}_{\mathrm{g}}^- = \vec{u}^+ - m\left[\frac{1}{\rho}\right]\hat{n}, \tag{3.104}$$

$$\vec{u}_{\mathrm{g}}^+ = \vec{u}^- + m\left[\frac{1}{\rho}\right]\hat{n}. \tag{3.105}$$

To maintain the discontinuity in $\vec{u}$ as the interface moves across velocity points, we evolve $\vec{u}^+$ and $\vec{u}^-$ separately in time. Specifically, we use $(\vec{u}^+)^{(n)}$

and $(\vec{u}^-)^{(n)}$ to calculate $(\vec{a}^+)^{(n)}$ and $(\vec{a}^-)^{(n)}$, respectively, with (3.2). Then we approximate the corresponding time derivatives of $\vec{u}^+$ and $\vec{u}^-$ with (3.1). The pressure is obtained by solving the Poisson equation (3.4), where the right hand side is discretized using values from $(\vec{u}^-)^{(n)}$ and $(\vec{a}^-)^{(n)}$ for Phase 1 points and $(\vec{u}^+)^{(n)}$ and $(\vec{a}^+)^{(n)}$ for points in Phase 2. Then, after each time step (or stage), we update the physical velocities $\vec{u}^{(n+1)}$ point by point according to

$$\vec{u}^{(n+1)} = \begin{cases} (\vec{u}^+)^{(n+1)} & \text{if } \phi^{(n+1)} > 0 \\ (\vec{u}^-)^{(n+1)} & \text{if } \phi^{(n+1)} \leq 0 \end{cases}. \tag{3.106}$$

3.5.3 Computing the Interface Velocity

When there was no mass flux across the interface, the interface velocity $\vec{w}$ was equal to the fluid velocity at the interface. When there is mass flux across the interface, however, we must account for this and use (2.20) when computing $\vec{w}$. We arrived at this expression in Section 2.4 by imposing mass conservation at the interface. This interface velocity $\vec{w}$ is used in the level-set advection equation (3.11) in cases with phase transition. Note that although we in (2.20) use the liquid (Phase 1) velocities $\vec{u}^-$ to calculate $\vec{w}$, we could, in principle, use either of $\vec{u}^-$ and $\vec{u}^+$.

3.6 Boundary Conditions

This section presents the implementation of the boundary conditions (BCs) described in Section 2.7. The stated formulas are based on those given by Griebel et al. [1998] and are applied in the beginning of each time step (or stage). The fields are defined on an $i_{\max} \times j_{\max}$ rectangular MAC grid and there is a band of ghost cells extending three cells into the far side of each of the four boundaries; west, east, south and north. It is the field values in these ghost cells and on their edges we set when we apply the BCs.

3.6.1 Velocity Boundary Conditions

No-Slip

With the no-slip BC, both the normal and the tangential component of the velocity are to be zero. At a specific boundary, we set the normal

component with the appropriate one of

$$u_{(0,j)} = 0, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)}, \tag{3.107}$$

$$u_{(i_{\max},j)} = 0, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)}, \tag{3.108}$$

$$v_{(i,0)} = 0, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)}, \tag{3.109}$$

$$v_{(i,j_{\max})} = 0, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)}. \tag{3.110}$$

The tangential velocity components do not lie directly on the boundary, so we enforce the no-slip condition by setting the ghost values so that the tangential velocities become zero when linearly interpolated to the boundary. That is, we set

$$v_{(0,j)} = -v_{(1,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)}, \tag{3.111}$$

$$v_{(i_{\max}+1,j)} = -v_{(i_{\max},j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)}, \tag{3.112}$$

$$u_{(i,0)} = -u_{(i,1)}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)}, \tag{3.113}$$

$$u_{(i,j_{\max}+1)} = -u_{(i,j_{\max})}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)}. \tag{3.114}$$

Free-Slip

With the free-slip BCs, the normal components of the velocities are zero and we can set them with (3.107) to (3.110). The normal derivative of the tangential velocity components are to be zero and we achieve this by mirroring their values across the boundary,

$$v_{(1-\ell,j)} = v_{(\ell,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(West)}, \tag{3.115}$$

$$v_{(i_{\max}+\ell,j)} = v_{(i_{\max}+1-\ell,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(East)}, \tag{3.116}$$

$$u_{(i,1-\ell)} = u_{(i,\ell)}, \qquad \forall i \in \{1, \ldots, i_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(South)}, \tag{3.117}$$

$$u_{(i,j_{\max}+\ell)} = u_{(i,j_{\max}+1-\ell)}, \qquad \forall i \in \{1, \ldots, i_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(North)}. \tag{3.118}$$

Outflow

With the outflow BC, the normal derivatives of both the normal and tangential velocity components are to be zero. We accomplish this by

mirroring both the normal,

$$u_{(0,j)} = u_{(1,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)}, \tag{3.119}$$
$$u_{(i_{\max}+1,j)} = u_{(i_{\max},j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)}, \tag{3.120}$$
$$v_{(i,0)} = v_{(i,1)}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)}, \tag{3.121}$$
$$v_{(i,j_{\max}+1)} = v_{(i,j_{\max})}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)}, \tag{3.122}$$

and the tangential velocity components,

$$v_{(0,j)} = v_{(1,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)}, \tag{3.123}$$
$$v_{(i_{\max}+1,j)} = v_{(i_{\max},j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)}, \tag{3.124}$$
$$u_{(i,0)} = u_{(i,1)}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)}, \tag{3.125}$$
$$u_{(i,j_{\max}+1)} = u_{(i,j_{\max})}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)}, \tag{3.126}$$

at the boundaries in question.

3.6.2 Level-set Function Boundary Conditions

As in Lervåg [2008], the level-set function is mirrored at all boundaries,

$$\phi_{(1-\ell,j)} = \phi_{(\ell,j)} \qquad \forall j \in \{1, \ldots, j_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(West)} \tag{3.127}$$
$$\phi_{(i_{\max}+\ell,j)} = \phi_{(i_{\max}+1-\ell,j)} \qquad \forall j \in \{1, \ldots, j_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(East)} \tag{3.128}$$
$$\phi_{(i,1-\ell)} = \phi_{(i,\ell)} \qquad \forall i \in \{1, \ldots, i_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(South)} \tag{3.129}$$
$$\phi_{(i,j_{\max}+\ell)} = \phi_{(i,j_{\max}+1-\ell)} \qquad \forall i \in \{1, \ldots, i_{\max}\}, \ell \in \{1, 2, 3\} \quad \text{(North)} \tag{3.130}$$

This results in an interface that is always normal to the boundary.

3.6.3 Pressure Boundary Conditions

Dirichlet

With this BC we set the pressure equal to a constant value along the boundaries in question. Since we are free to choose the zero-level of the

pressure, we set

$$p_{(0,j)} = 0 \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)} \tag{3.131}$$
$$p_{(i_{\max}+1,j)} = 0 \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)} \tag{3.132}$$
$$p_{(i,0)} = 0 \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)} \tag{3.133}$$
$$p_{(i,j_{\max}+1)} = 0 \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)} \tag{3.134}$$

Neumann

Due to the choice of projection method we can in most cases set the normal component of the pressure gradient equal to zero at the boundaries [Griebel et al., 1998, Section 3.2.10],

$$p_{(0,j)} = p_{(1,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)}, \tag{3.135}$$
$$p_{(i_{\max}+1,j)} = p_{(i_{\max},j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)}, \tag{3.136}$$
$$p_{(i,0)} = p_{(i,1)}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)}, \tag{3.137}$$
$$p_{(i,j_{\max}+1)} = p_{(i,j_{\max})}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)}. \tag{3.138}$$

3.6.4 Temperature Boundary Conditions

Dirichlet

With Dirichlet BCs, we want to set the temperature at the west, east, south or north boundary equal to the constant values T_{west}, T_{east}, T_{south} or T_{north}, respectively. However, as the temperatures are not located directly on the boundaries, we resort to setting them such that they have the desired value at the boundary when linearly interpolated. We set

$$T_{(0,j)} = 2T_{\text{west}} - T_{(1,j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(West)}, \tag{3.139}$$
$$T_{(i_{\max}+1,j)} = 2T_{\text{east}} - T_{(i_{\max},j)}, \qquad \forall j \in \{1, \ldots, j_{\max}\} \quad \text{(East)}, \tag{3.140}$$
$$T_{(i,0)} = 2T_{\text{south}} - T_{(i,1)}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(South)}, \tag{3.141}$$
$$T_{(i,j_{\max}+1)} = 2T_{\text{north}} - T_{(i,j_{\max})}, \qquad \forall i \in \{1, \ldots, i_{\max}\} \quad \text{(North)}. \tag{3.142}$$

Neumann

With Neumann BCs, we want to impose heat flux densities q_{west}, q_{east}, q_{south} or q_{north} *into* the fluid domain from the west, east, south or north

boundaries respectively. We set

$$T_{(0,j)} = T_{(1,j)} + \frac{q_{\text{west}}}{\kappa}\Delta x_{(0)}, \qquad \forall j \in \{1,\ldots,j_{\max}\} \quad \text{(West)}, \tag{3.143}$$

$$T_{(i_{\max}+1,j)} = T_{(i_{\max},j)} + \frac{q_{\text{east}}}{\kappa}\Delta x_{(i_{\max})}, \quad \forall j \in \{1,\ldots,j_{\max}\} \quad \text{(East)}, \tag{3.144}$$

$$T_{(i,0)} = T_{(i,1)} + \frac{q_{\text{south}}}{\kappa}\Delta y_{(0)}, \qquad \forall i \in \{1,\ldots,i_{\max}\} \quad \text{(South)}, \tag{3.145}$$

$$T_{(i,j_{\max}+1)} = T_{(i,j_{\max})} + \frac{q_{\text{north}}}{\kappa}\Delta y_{(j_{\max})}, \quad \forall i \in \{1,\ldots,i_{\max}\} \quad \text{(North)}. \tag{3.146}$$

3.7 Time-Step Restrictions

To ensure stability of the SSP RK schemes, we use standard Courant–Friedrichs–Lewy (CFL) criteria to calculate the safe length of each time step. These are limited by a safety factor, the CFL number $C_{\text{CFL}} \in \langle 0, 1\rangle$. In this work, all simulations were run with $C_{\text{CFL}} = 0.5$.

As Griebel et al. [1998], we calculate a safe time step for integration of the velocities $(\Delta t)_{\vec{u}}$ and a safe time step for the temperatures $(\Delta t)_T$ separately and let the time-step length be

$$\Delta t = \min\left\{(\Delta t)_{\vec{u}}, (\Delta t)_T\right\}. \tag{3.147}$$

The manner in which we calculate $(\Delta t)_{\vec{u}}$ and $(\Delta t)_T$ depends on whether we have a single- or two-phase case and, if we have a two-phase case, whether we model phase transition or not. Before we consider these different cases, we define

$$\Delta x_{\min} = \min_i \left(\Delta x_{(i)}\right), \tag{3.148}$$

$$\Delta y_{\min} = \min_j \left(\Delta y_{(j)}\right). \tag{3.149}$$

3.7.1 Single-Phase Simulations

The stability criterion for $\vec{u}$ in single-phase incompressible flow states that no fluid particle may travel longer than across one grid cell in the time

interval Δt [Griebel et al., 1998, Section 3.2.4]. This is ensured by selecting a Δt satisfying

$$\Delta t \leq C_x = \min_{i,j} \left(\frac{\Delta x_{(i)}}{|u_{(i,j)}|} \right), \tag{3.150}$$

$$\Delta t \leq C_y = \min_{i,j} \left(\frac{\Delta y_{(j)}}{|v_{(i,j)}|} \right). \tag{3.151}$$

In cases of viscous flow, we also need to have

$$\Delta t \leq C_\nu = \frac{1}{2\nu} \left\{ \min_{i,j} \left(\frac{1}{\left(\Delta x_{(i)}\right)^2} + \frac{1}{\left(\Delta y_{(j)}\right)^2} \right) \right\}^{-1}. \tag{3.152}$$

We ensure that Δt satisfies all of (3.150) to (3.152) by setting

$$(\Delta t)_{\vec{u}} = C_{\mathrm{CFL}} \min \left(C_x, C_y, C_\nu \right). \tag{3.153}$$

Looking to Griebel et al. [1998, Section 9.4], we need to make Δt satisfy

$$2\Delta t \alpha \min_{i,j} \left(\frac{1}{\left(\Delta x_{(i)}\right)^2} + \frac{1}{\left(\Delta y_{(j)}\right)^2} \right) \leq 1 \tag{3.154}$$

to ensure stability of the temperature integration. In this work, we accomplish that by setting

$$(\Delta t)_T = C_{\mathrm{CFL}} \frac{\left(\min \left\{ \Delta x_{\min}, \Delta y_{\min} \right\}\right)^2}{4\alpha}. \tag{3.155}$$

3.7.2 Two-Phase Simulations

For two-phase flows, we continue in the spirit of Kang et al. [2000] and demand that Δt must satisfy

$$\frac{\Delta t}{2} \left(C_{\mathrm{c}} + C_\nu + \sqrt{(C_{\mathrm{c}} + C_\nu)^2 + 4C_g^2 + 4C_{\mathrm{s}}^2} \right) \leq 1, \tag{3.156}$$

where

$$C_{\mathrm{c}} = \max_{i,j} \left(\frac{|u_{(i,j)}|}{\Delta x_{(i)}} \right) + \max_{i,j} \left(\frac{|v_{(i,j)}|}{\Delta y_{(j)}} \right), \tag{3.157}$$

$$C_\nu = 2\max\left(\nu^+, \nu^-\right) \max_{i,j} \left(\frac{1}{\left(\Delta x_{(i)}\right)^2} + \frac{1}{\left(\Delta y_{(j)}\right)^2} \right), \quad (3.158)$$

$$C_\mathrm{s} = \sqrt{\frac{\sigma \max_{i,j}\left(k_{(i,j)}\right)}{\max\left(\rho^+, \rho^-\right)\left(\min\left\{\Delta x_\mathrm{min}, \Delta y_\mathrm{min}\right\}\right)^2}}, \quad (3.159)$$

$$C_g = \sqrt{\frac{|g_x|}{\Delta x_\mathrm{min}} + \frac{|g_y|}{\Delta y_\mathrm{min}}}. \quad (3.160)$$

Note that in (3.159), Kang et al. [2000] use $\min\left(\rho^+, \rho^-\right)$. However, this was judged overly conservative and replaced with $\max\left(\rho^+, \rho^-\right)$ by Lervåg [2008]. We then let

$$(\Delta t)_{\vec{u}} = \frac{2C_\mathrm{CFL}}{C_\mathrm{c} + C_\nu + \sqrt{(C_\mathrm{c} + C_\nu)^2 + 4C_g^2 + 4C_\mathrm{s}^2}}. \quad (3.161)$$

In a case where we do not model phase transition, we let $(\Delta t)_T$ be an extension of its single-phase counterpart (3.153),

$$(\Delta t)_T = C_\mathrm{CFL} \frac{\left(\min\left\{\Delta x_\mathrm{min}, \Delta y_\mathrm{min}\right\}\right)^2}{4\max\left(\alpha^+, \alpha^-\right)}. \quad (3.162)$$

When we do model phase transition then, according to Gibou et al. [2002], we must demand that Δt satisfies

$$\Delta t \max\left(\alpha^+, \alpha^-\right) \left(\frac{2}{\min_i\left\{(\theta \Delta x_{(i)})^2\right\}} + \frac{2}{\min_j\left\{(\theta \Delta y_{(j)})^2\right\}} \right) \leq 1 \quad (3.163)$$

to guarantee stability of the temperature. In this thesis $\theta > \theta_\mathrm{c}$, so it is safe to set

$$(\Delta t)_T = C_\mathrm{CFL} \frac{\left(\theta_\mathrm{c} \min\left\{\Delta x_\mathrm{min}, \Delta y_\mathrm{min}\right\}\right)^2}{4\max\left(\alpha^+, \alpha^-\right)}. \quad (3.164)$$

If we compare this time-step restriction with (3.162), we see that it is stricter by a factor θ_c^2. We want to choose θ_c reasonably small to get good approximations to the spatial derivatives of T, e.g. in (3.81), but the price for this is safe time steps that are short. In this thesis we have chosen $\theta_\mathrm{c} = 0.25$ to avoid very short time steps.

3.8 Estimation of Error and Convergence Order

In order to quantify the error in an obtained solution with respect to some reference, we will use one of several possible measures of error.

In the case where the solution a and reference a_{ref} are real numbers, $a, a_{\text{ref}} \in \mathbb{R}$, we simply define the error E_{abs} as the absolute difference

$$E_{\text{abs}} = |a - a_{\text{ref}}| \,. \tag{3.165}$$

When we have vectors $\vec{a}, \vec{a}_{\text{ref}} \in \mathbb{R}^n$, we define the scaled 1-norm E_1,

$$E_1 = \frac{\|\vec{a} - \vec{a}_{\text{ref}}\|_1}{N}, \tag{3.166}$$

as a measure of error. For a formal definition of norms and their properties, see e.g. Süli and Mayers [2006].

We will assume that the error as a function of grid size $E(N)$ can be written as a polynomial expansion in $1/N$. For a sufficiently fine grid, that is for large enough values of N, the error of a solution is dominated by the leading term in this polynomial expansion,

$$E(N) \approx \frac{C}{N^n}. \tag{3.167}$$

The number n is what we will refer to as the convergence order.

Taking the logarithm on both sides of (3.167) we get

$$\log_{10}\{E(N)\} \approx \log_{10}(C) - n \log_{10}(N) \,, \tag{3.168}$$

so plotting $\log_{10}\{E(N)\}$ against $\log_{10}(N)$ should give a straight line with slope $-n$. This is one way of estimating the convergence order.

By dividing the error in a solution for one grid size $E(N_1)$ by the error in the solution for another $E(N_2)$ we get an expression that we can solve for n,

$$n \approx \frac{\log_{10}\left(\frac{E(N_1)}{E(N_2)}\right)}{\log_{10}\left(\frac{N_2}{N_1}\right)}. \tag{3.169}$$

Another way of estimating the convergence order is then to evaluate (3.169) for pairs of N.

To have a quantitative notion of how close a simulated solution is to steady-state, we define E_{ss} of the field φ as

$$E_{\text{ss}}(\varphi) = \frac{\max_{i,j}\left(\varphi_{(i,j)}^{(n)} - \varphi_{(i,j)}^{(n-\ell)}\right)}{t^{(n)} - t^{(n-\ell)}}, \tag{3.170}$$

where $\ell \geq 1$ and n is the final time step in the simulation. If the solution converges to steady-state, then E_{ss} should provide an upper bound for the simulated $\partial_t \varphi$ at the end of the simulation.

3.9 Summary

We have presented the numerical methods used to solve the two-phase flow model. Chorin's projection method (Section 3.1) was employed to get an explicit expression (3.1) for the velocity derivatives $\partial_t \vec{u}^{(n)}$ and a Poisson equation (3.4) for the pressure $p^{(n+1)}$. The level-set method, used to keep track of the interface location through the time integration of the level-set equation (3.11), was presented (Section 3.2) and the finite-difference approximations to the spatial derivative operators (Section 3.4), and thus to e.g. the temperature derivative $\partial_t T^{(n)}$, were described. We also described a strategy for handling the phase-transition model numerically (Section 3.5), by calculating the mass flux density m from the temperature field using (3.101) and use m to enforce the interface condition on the velocity field and find the velocity of the interface. The implementation of the boundary conditions (Section 3.6) was presented and to ensure stability of the Runge–Kutta schemes (Section 3.3), we gave expressions for the maximum length of safe time steps (Section 3.7).

4 Numerical Experiments

In this chapter, we describe the eight test cases for which simulations were run in this thesis. The main purpose of most of these numerical experiments is to verify the methods and their implementation by comparing results from simulation against analytical or reliable numerical results. Some of the cases are therefore maybe not so interesting in themselves, but they may reveal interesting features of the method or errors in the implementation.

4.1 De Vahl Davis' Benchmark Case

Before we can hope to get correct numerical results for simulations of heat transport in two-phase flow, we should confirm that we are able to obtain reasonable results for single-phase simulations. To this avail, a widely used* benchmark case was run and the results were compared with results from de Vahl Davis [1983].

In this case, as illustrated in Figure 4.1, we consider a two-dimensional, fluid-filled, square cavity of width and height $L_x = L_y = 1.0\,\mathrm{m}$. We impose no-slip boundary conditions on the walls. There is a constant high temperature $T_{\mathrm{west}} = 294\,\mathrm{K}$ on the western wall, a constant low temperature $T_{\mathrm{east}} = 292\,\mathrm{K}$ on the eastern wall and thermally insulating north and south walls. Initially the fluid is at rest and the temperature in the fluid domain is uniform and equal to T_{east}. The gravitational acceleration vector $\vec{g}_\infty$ is set to $-9.81\,\mathrm{m\,s^{-2}}$ along the y-axis.

The simulations were run with Prandtl number $Pr = 0.71$ and two different Rayleigh numbers, $Ra = 10^3$ (Case A) and $Ra = 10^6$ (Case B), corresponding to the fluid parameters in Table 4.1. They were run to $t = 5 \cdot 10^4\,\mathrm{s}$, where steady-state was verified to have been established.

*At the time of writing de Vahl Davis [1983] had 186 article citations according to Web of Knowledge.

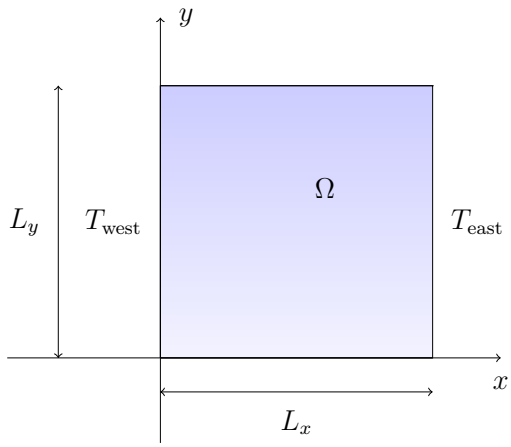


Figure 4.1: Schematic illustration of the de Vahl Davis benchmark case configuration.

The Ra and Pr numbers are

$$Ra = \frac{\|\vec{g}_\infty\|_2 \, \beta (T_{\text{west}} - T_{\text{east}}) L_x^3}{\nu \alpha}, \tag{4.1}$$

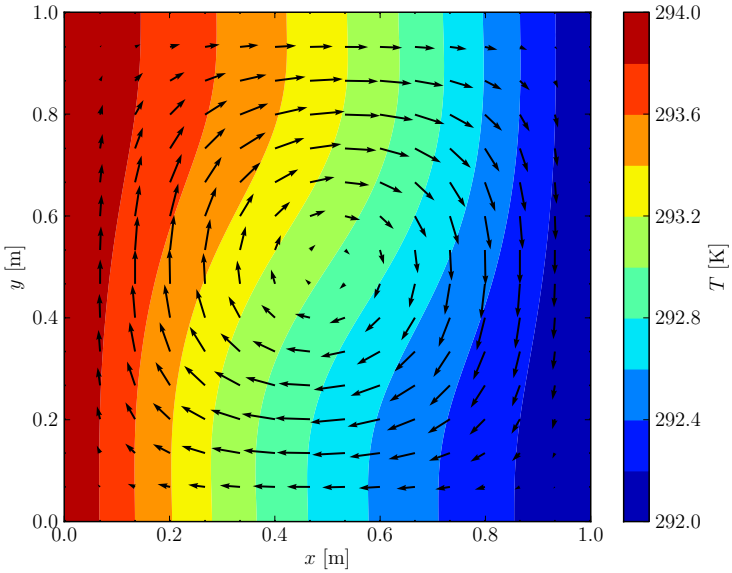
$$Pr = \frac{\nu}{\alpha}, \tag{4.2}$$

for these cases.

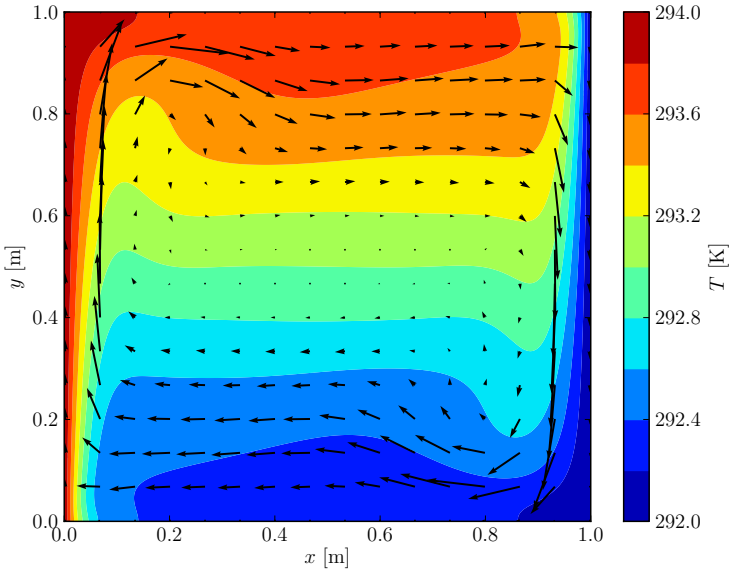
4.1.1 Numerical Results

Contour plots of the temperatures obtained on 301×301 grids at $t = 5 \cdot 10^4$ s for both Case A and Case B are shown in Figure 4.2. These contour plots can be qualitatively compared with Figures 4a and 4d from de Vahl Davis [1983]. It was verified that the magnitude of E_{ss} did not exceed 10^{-7} in the appropriate SI base units for the T and $\vec{u}$ fields. Looking at the vertical velocity distribution along the horizontal center lines for the two cases, we see that it is more sharply peaked at the hot and the cold walls for Case B than for Case A. This is consistent with the higher Ra of Case B, which says that ratio of buoyancy forces to viscous forces are higher for Case B.

A quantitative comparison with the benchmark case was done with respect to the maximum horizontal velocity component $u_{\max}$ along the ver-



(a)



(b)

Figure 4.2: Contour plot of the temperature in the de Vahl Davis bench mark case for (a) Case A, $Ra = 10^3$ and (b) Case B, $Ra = 10^6$. The black arrows illustrate the velocity field.

Table 4.1: Fluid properties used in the de Vahl Davis benchmark case.

	Unit	Case A	Case B
Pr	1	0.71	0.71
Ra	1	10^3	10^6
ρ_∞	$\mathrm{kg\,m^{-3}}$	1.2041	1.2041
μ	$\mathrm{Pa\,s}$	$1.7400 \cdot 10^{-5}$	$1.7400 \cdot 10^{-5}$
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	$1.012 \cdot 10^3$	$1.012 \cdot 10^3$
β	$\mathrm{K^{-1}}$	$1.4990 \cdot 10^{-8}$	$1.4990 \cdot 10^{-5}$
κ	$\mathrm{W\,K^{-1}\,m^{-1}}$	$2.4801 \cdot 10^{-2}$	$2.4801 \cdot 10^{-2}$
T_∞	K	293.0	293.0

tical center line of the fluid domain. In the benchmark solution, this quantity is $7.4268 \cdot 10^{-5}\,\mathrm{m\,s^{-1}}$ for Case A and $1.3154 \cdot 10^{-3}\,\mathrm{m\,s^{-1}}$ for Case B .

In Case A, on the 301×301 grid, $u_{\max}$ was $7.4279 \cdot 10^{-5}\,\mathrm{m\,s^{-1}}$. This value of $u_{\max}$ was obtained, as in de Vahl Davis [1983], by doing a fourth order polynomial interpolation of u around the maximum point on the vertical center line and then finding the maximum of the interpolation polynomial. The obtained value corresponded to an error of approximately 0.01% with respect to the benchmark solution. de Vahl Davis [1983] estimates the error in this benchmark solution to be 0.1%. The error in our solution with respect to the benchmark is then smaller than the estimated error of the benchmark itself. In that sense, our 301×301 solution is as close to the benchmark as one could reasonably hope to get. Therefore we conclude that there was agreement between our solution on the 301×301 grid and the benchmark solution.

Similarly, the simulated $u_{\max}$ was $1.3206 \cdot 10^{-3}\,\mathrm{m\,s^{-1}}$ for Case B on the 301×301 grid, corresponding to an error with respect to the benchmark of 0.4%. This is again smaller than the estimated error of the benchmark solution of 1% and therefore in agreement with the benchmark solution.

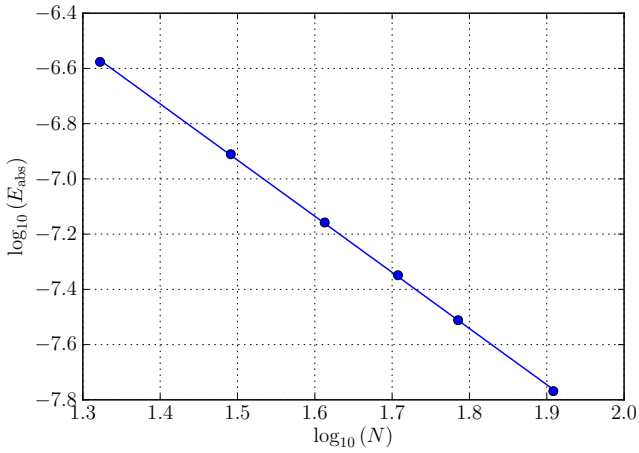
The order of convergence was investigated for both cases A and B, with the 301×301 solutions as reference solutions. The resulting errors E_{abs} and orders of convergence n, estimated using (3.169) from page 51, are

Table 4.2: Convergence table based on the maximum horizontal velocity component $u_{\max}$ along the vertical center line for the de Vahl Davis benchmark Case A with $Ra = 10^3$.

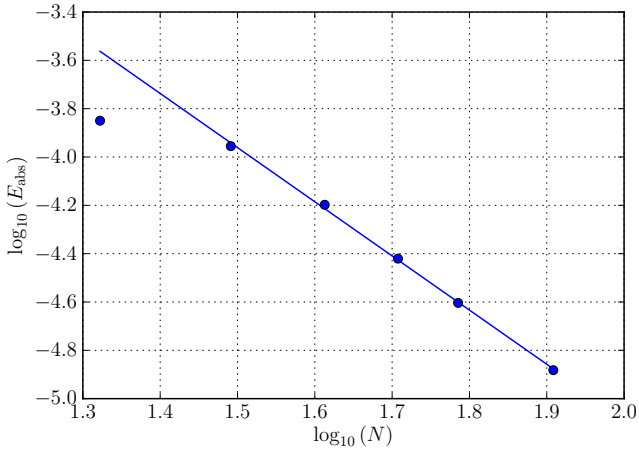
N	$u_{\max}$	E_{abs}	n
-	$\mathrm{m\,s^{-1}}$	$\mathrm{m\,s^{-1}}$	-
21	$7.454 \cdot 10^{-05}$	$2.65 \cdot 10^{-07}$	-
31	$7.440 \cdot 10^{-05}$	$1.23 \cdot 10^{-07}$	1.98
41	$7.435 \cdot 10^{-05}$	$6.95 \cdot 10^{-08}$	2.04
51	$7.432 \cdot 10^{-05}$	$4.48 \cdot 10^{-08}$	2.01
61	$7.431 \cdot 10^{-05}$	$3.08 \cdot 10^{-08}$	2.09
81	$7.430 \cdot 10^{-05}$	$1.70 \cdot 10^{-08}$	2.09

Table 4.3: Convergence table based on the maximum horizontal velocity component $u_{\max}$ along the vertical center line for the de Vahl Davis benchmark Case B with $Ra = 10^6$.

N	$u_{\max}$	E_{abs}	n
-	$\mathrm{m\,s^{-1}}$	$\mathrm{m\,s^{-1}}$	-
21	$1.462 \cdot 10^{-03}$	$1.41 \cdot 10^{-04}$	-
31	$1.431 \cdot 10^{-03}$	$1.11 \cdot 10^{-04}$	0.62
41	$1.384 \cdot 10^{-03}$	$6.34 \cdot 10^{-05}$	2.00
51	$1.359 \cdot 10^{-03}$	$3.80 \cdot 10^{-05}$	2.35
61	$1.346 \cdot 10^{-03}$	$2.49 \cdot 10^{-05}$	2.35
81	$1.334 \cdot 10^{-03}$	$1.31 \cdot 10^{-05}$	2.26



(a)



(b)

Figure 4.3: Log-log plot of the error E_{abs} in $u_{\max}$ against the linear grid size N in the de Vahl Davis bench mark case for (a) Case A, $Ra = 10^3$, where the solid line represents a first order polynomial fit and has a slope of approximately -2.03 and (b) Case B, $Ra = 10^6$, where the solid line represents a first order polynomial fit to the five rightmost points and has a slope of approximately -2.24.

shown in Table 4.2 for Case A and in Table 4.3 for Case B. These tables show a steady decrease of the E_{abs} upon grid refinement and an estimated convergence order that is consistent with the expected value of 2. This steady decrease in error is visualized in Figures 4.3a and 4.3b.

From the results of the simulations of the de Vahl Davis benchmark case we can conclude that the implementation is able to reproduce the benchmark results from de Vahl Davis [1983] on the 301×301 grid. Also, a convergence study indicates that the order of convergence is consistent with the expected value of 2. This gives reason to believe that the current implementation of the discretization of the advection-diffusion equation and the Boussinesq coupling to the Navier–Stokes equations for single-phase flow cases is correct.

4.2 Two-Phase Heat Conduction

A natural first test of the two-phase CSF and GFM discretizations of the diffusion term in the advection-diffusion equation (2.11) on page 8 is to solve this in the absence of convection. In other words, to solve the heat equation. For simplicity we also restrict ourselves to one dimension. In this case (2.11) reduces to

$$\frac{d}{dy}\left(\kappa \frac{dT}{dy}\right) = 0, \tag{4.3}$$

in steady-state.

We consider a cavity of height L containing two phases, Phase 2 on top of Phase 1 and a horizontal boundary between them at $y = \eta L$. The relevant fluid properties are given in Table 4.4. In order to have no convection, the fluid is initially at rest and the gravitational vector $\vec{g}_\infty$ is set to zero. The temperature is initially uniform and equal to 290 K.

This situation was simulated with two sets of boundary conditions, Case A and Case B. First it was run with constant-temperature boundary conditions (Dirichlet), $T_{\text{north}} = 285\,\text{K}$ on the north and $T_{\text{south}} = 295\,\text{K}$ on the south wall and insulating western and eastern walls (Case A) and second with constant heat flux densities $q_{\text{south}} = -q_{\text{north}} = 1.0\,\text{W m}^{-2}$ (Neumann) and again insulating eastern and western walls (Case B). Due to our convention from Section 2.7.3, we have heat flowing into the domain through the southern wall and out through the northern wall.

Table 4.4: Relevant fluid properties used in the two-phase heat conduction test case.

	Unit	Phase 1	Phase 2
ρ_∞	$\mathrm{kg\,m^{-3}}$	2.0	1.0
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	100.0	100.0
κ	$\mathrm{W\,K^{-1}\,m^{-1}}$	0.04	0.01

Although the case was simulated on a two-dimensional grid, the boundary conditions and initial configuration should lead to an x-independent solution and the temperature field in steady-state along any constant-x line should be governed by (4.3). The case was run to $t = 1 \cdot 10^4\,\mathrm{s}$, where steady-state was verified to have been obtained with the GFM. The results were then compared to an analytical solution.

4.2.1 Analytical Solution

One of the appealing aspects of (4.3) is that it can quite easily be solved analytically. By differentiation and insertion it is easy to confirm that the following linear temperature profile is a solution to (4.3),

$$T(y) = \begin{cases} T^-(y) = C_0^- + C_1^- y & \text{if } y \in \Omega^- \\ T^+(y) = C_0^+ + C_1^+ y & \text{if } y \in \Omega^+ \end{cases}. \tag{4.4}$$

We assume that the interface is located at a known location ηL, where $\eta \in [0, 1]$.

To determine the unknown coefficients of (4.4), we need to use the boundary conditions and impose continuity of the temperature field and continuity of the heat flux. From the Dirichlet boundary conditions in Case A, that is constant and known temperatures T_{south} and T_{north} at $y = 0$ and $y = L$ respectively, we have

$$T^-(0) = T_{\text{south}}, \tag{4.5}$$

$$T^+(L) = T_{\text{north}}, \tag{4.6}$$

$$T^-(\eta L) = T^+(\eta L), \tag{4.7}$$

$$\kappa^- \left.\frac{dT^-}{dy}\right|_{\eta L} = \kappa^+ \left.\frac{dT^+}{dy}\right|_{\eta L}, \tag{4.8}$$

which determines the four coefficients C_0^-,C_1^-,C_0^+ and C_1^+ as

$$C_1^- = \frac{T_{\text{north}} - T_{\text{south}}}{L\left(\eta + \{1-\eta\}\frac{\kappa^-}{\kappa^+}\right)}, \tag{4.9}$$

$$C_0^- = T_{\text{south}}, \tag{4.10}$$

$$C_1^+ = \frac{\kappa^-}{\kappa^+} C_1^-, \tag{4.11}$$

$$C_0^+ = T_{\text{north}} - C_1^+ L. \tag{4.12}$$

For the Neumann boundary conditions in Case B, with constant heat flux density $q = |q_{\text{south}}| = |q_{\text{north}}|$ at the top and bottom walls, continuity of heat flux and continuity of temperature, we have

$$q = -\kappa^- \left.\frac{dT^-}{dy}\right|_0 \tag{4.13}$$

$$q = -\kappa^+ \left.\frac{dT^+}{dy}\right|_L \tag{4.14}$$

$$T^-(\eta L) = T^+(\eta L), \tag{4.15}$$

$$\kappa^- \left.\frac{dT^-}{dy}\right|_{\eta L} = \kappa^+ \left.\frac{dT^+}{dy}\right|_{\eta L} \tag{4.16}$$

which determines the three coefficients C_1^-,C_0^+ and C_1^+ as

$$C_1^- = -\frac{q}{\kappa^-}, \tag{4.17}$$

$$C_1^+ = -\frac{q}{\kappa^+}, \tag{4.18}$$

$$C_0^+ = C_0^- + \eta L q\left(\frac{1}{\kappa^+} - \frac{1}{\kappa^-}\right). \tag{4.19}$$

C_0^-, again equal to $T(0)$, is undetermined by these boundary conditions.

4.2.2 Numerical Results

Since there is no gravity, there should be no advection and the fluid velocity should remain zero everywhere at all times. This was also found to be true for all simulations run. Thus the advective term in (2.11) was zero in the whole domain in all simulations.

The analytical steady-state solution for the temperature profile is piecewise linear with a kink at the boundary, both for Case A and Case B. Thus the diffusive term in (2.11) should also be zero in the whole domain. In the absence of advection, we see from (2.11) that a diffusive term that is zero in the entire domain is necessary to get a steady-state solution. If it is not zero then $\partial_t T \neq 0$.

Continuum Surface Force Method

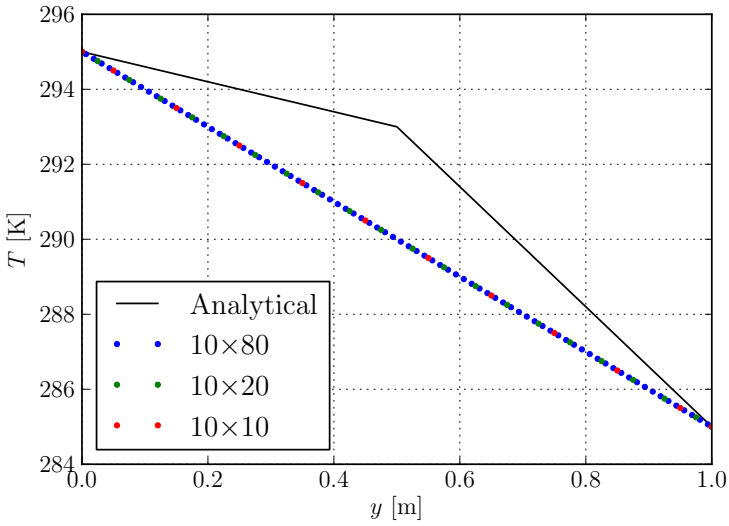
The temperature profiles obtained with the CSF method for Case A are shown in Figure 4.4a for different grid sizes. These simulations did reach steady-state in the sense that $E_{\text{ss}}(T)$ was zero. This is consistent with the linear temperature profile obtained from the simulations.

However, as is apparent from Figure 4.4a, the temperature profile is not in agreement with the analytical solution and grid refinement did not seem to produce better results. Also, the spatial derivative of the temperature profile is physically unacceptable. To see why, consider the heat flux $\kappa\ dT/dy$ at $y = 0$ and at $y = L$. Four times as much heat flows in through the southern wall at $y = 0$ compared to the amount of heat that flows out through the northern wall at $y = L$. So there is a net heat flow into the domain, yet the domain temperatures do not change. This is a clear violation of energy conservation.

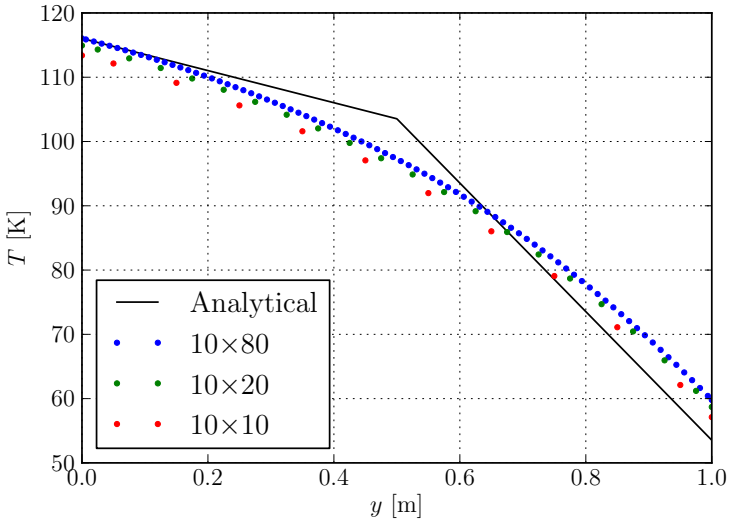
This leads us to conclude that the CSF discretization of the diffusive term in the advection-diffusion equation failed to reproduce the analytical solution in the two-phase heat conduction case with Dirichlet boundary conditions.

The temperature profile obtained with the CSF method in Case B is shown in Figure 4.4b for different grid sizes. With Neumann boundary conditions, the analytical steady-state temperature profile is only determined up to the addition of a constant. Specifically, the constant $C_0^- = T(0)$ is undetermined by the boundary conditions. For the sake of comparison, this constant is here taken to be the calculated south wall temperature of the 10×80 simulation.

In contrast to Case A with CSF, these simulations did not reach steady-state. That is, $\partial_t T$ averaged in the whole domain did not approach zero during the time integration, but seemed to approach a constant, negative value instead. Looking at (2.11), we see that a local negative curvature in



(a)



(b)

Figure 4.4: Comparison between the analytical solution to the one-dimensional heat equation and the numerical solution with different grid sizes and the CSF method for (a) constant temperature (Dirichlet) boundary conditions (Case A) and (b) constant heat flux (Neumann) boundary conditions (Case B).

the temperature field is equivalent to a temperature drain. The drain of temperature is then consistent with the negative curvature of the temperature profile in Figure 4.4b, but still unphysical and not in agreement with the analytical solution. To see why it is unphysical, we again consider the heat flux at the boundaries. The heat flux into the domain at the south wall and out of the domain at the north wall are in this case equal, as posed by the boundary conditions, yet the average domain temperature does not approach a constant value. This is a clear violation of energy conservation. The CSF method seems to introduce an artificial curvature in T that drains temperature from the inner domain.

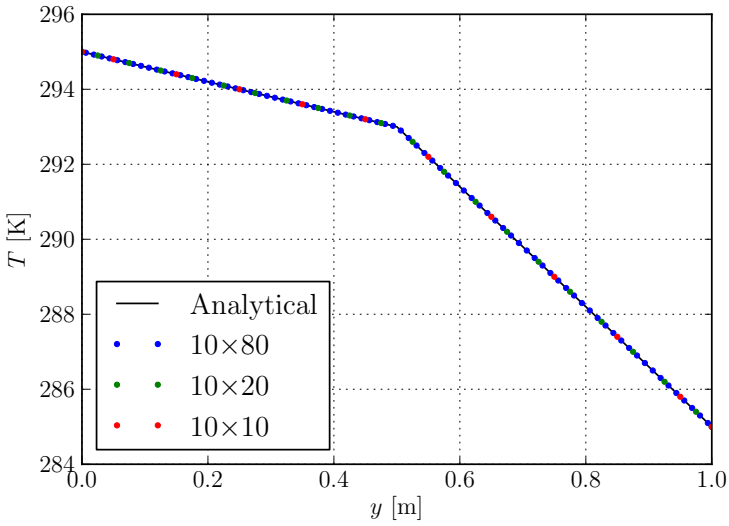
This leads us to conclude that the CSF discretization of the diffusive term in the advection-diffusion equation failed to reproduce the analytical solution in the two-phase heat conduction case with Neumann boundary conditions.

Ghost-Fluid Method

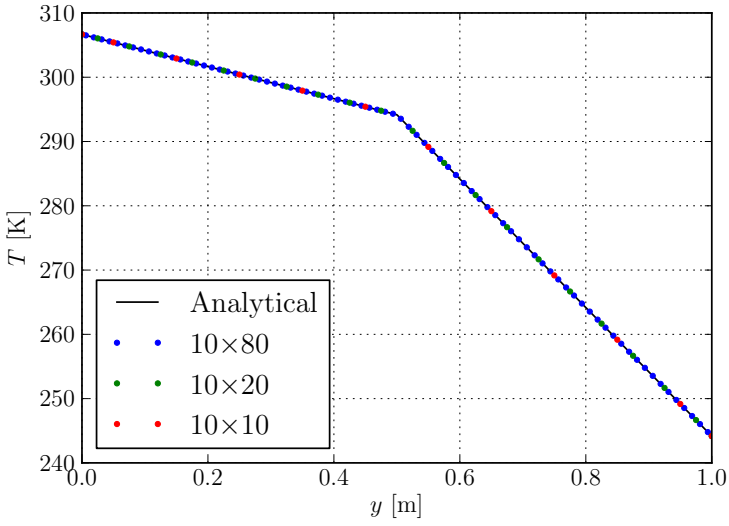
The temperature profile obtained with the GFM in Case A is shown in Figure 4.5a for different grid sizes. These simulations did reach steady-state, $E_{\text{ss}}(T)$ was zero, consistent with the obtained piecewise linear temperature profiles. For all grid sizes E_1 with respect to the analytical solution was zero. The GFM method was thus able to obtain the exact solution of the one dimensional heat equation in Case A[†]. This leads us to conclude that the GFM discretization of the diffusive term in the advection-diffusion equation produced the correct solution to the one dimensional heat equation with Dirichlet boundary conditions.

The temperature profile obtained with the GFM in Case B is shown in Figure 4.5b for different grid sizes. These simulations did also reach steady-state, $E_{\text{ss}}(T)$ was zero, consistent with the obtained piecewise linear temperature profiles. For all grid sizes E_1 was approximately $4 \cdot 10^{-15}\,\text{K}$, so the correct temperature profiles were obtained to machine precision for

[†]It may seem strange and even suspicious that the E_1 turned out to be zero and not a small number on order of magnitude with the machine precision. The reason for it was probably that, for this particular case and choice of grid point locations, the analytical solution evaluated in the grid points consisted of rational numbers that could be exactly represented within the machine precision. Alternatively, the difference in the numerical solution to the analytical solution was small enough to cause underflow in the double precision data type used to store it.



(a)



(b)

Figure 4.5: Comparison between the analytical solution to the one-dimensional heat equation and numerical solutions with different grid sizes and the GFM method for (a) Case A, constant temperature (Dirichlet) boundary conditions and (b) Case B, constant heat flux (Neumann) boundary conditions.

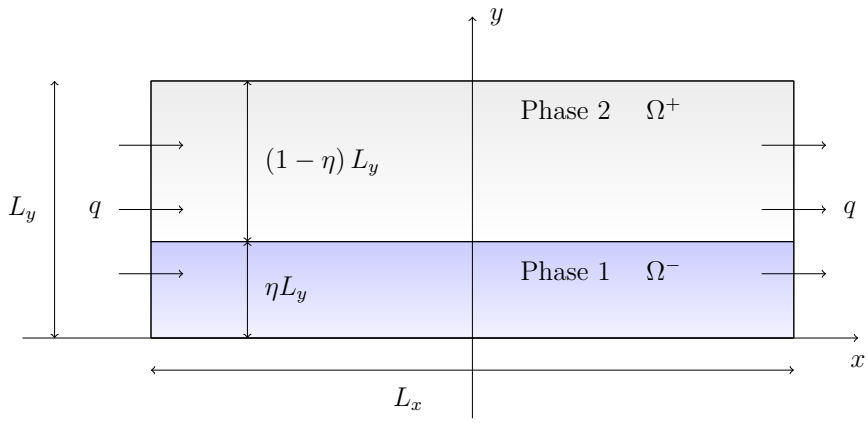


Figure 4.6: Schematic illustration of the two-layer configuration in Wang et al.'s convection case.

all grid sizes. This leads us to conclude that the GFM discretization of the diffusive term in the advection-diffusion equation produced the correct solution to the one-dimensional heat equation with Neumann boundary conditions.

This test case shows the advantage of treating the interface sharply with the GFM instead of with the CSF method when solving the heat equation. The GFM is able to obtain temperature profiles with an error limited by machine precision regardless of grid size while those produced by the CSF method are, at best, questionable.

4.3 Wang et al.'s Two-Layer Convection Case

As a test for temperature driven two-phase flow, the case presented by Wang et al. [1991] was simulated. A two-dimensional, rectangular cavity of length L_x, height L_y and aspect ratio $A = L_x/L_y$ contains two phases, Phase 2 on top of Phase 1 in a two-layer configuration. A constant and uniform heat flux density $q_{\text{west}} = q$ points into the domain on the western wall and a heat flux density $q_{\text{east}} = -q$ points out of the domain on the eastern wall, as illustrated in Figure 4.6. The northern and southern walls are insulating.

The size of the domain used in the numerical simulation was $L_x =$

Table 4.5: Fluid parameters used in the numerical simulation of Wang et al.'s two-layer convection case.

	Unit	Phase 1	Phase 2
ρ	$\mathrm{kg\,m^{-3}}$	1.5	1.0
μ	$\mathrm{Pa\,s}$	$2.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	120.0	100.0
κ	$\mathrm{W\,K^{-1}\,m^{-1}}$	0.013	0.010
β	$\mathrm{K^{-1}}$	$1 \cdot 10^{-4}$	$1 \cdot 10^{-3}$
T_∞	K	290	290

0.8 m and $L_y = 0.1\,\mathrm{m}$, with the interface located at the constant-y line defined by $y = \eta L_y = 0.05\,\mathrm{m}$. The initial temperature was set to 290 K in the entire inner domain and the gravitational acceleration vector $\vec{g}_\infty$ was set to $-9.81\,\mathrm{m\,s^{-2}}$ along the y-axis. The heat flux density q used was $1.0\,\mathrm{W\,m^{-2}}$. The fluid properties are given in Table 4.5.

This case was run to $t = 10^3\,\mathrm{s}$ with the GFM and compared to the analytical solution by Wang et al. [1991].

4.3.1 Analytical Solution

In the limiting case where $A \to \infty$, Wang et al. [1991] showed that one can use the steady-state and the parallel flow assumptions to obtain an analytical solution to the Navier–Stokes equations (2.2) and (2.10) and advection-diffusion equation (2.11) that is valid near the middle of the cavity. Here, we will present this solution and a derivation of the expressions needed to verify it. Thus we elaborate on some of the points in the derivation in more detail than what is done in Wang et al. [1991].

The Partial Differential Equations

We start the derivation by introducing the stream function ψ,

$$u = \frac{\partial \psi}{\partial y}, \tag{4.20}$$

$$v = -\frac{\partial \psi}{\partial x}. \tag{4.21}$$

Inserting this definition of ψ into the incompressible flow assumption (2.2), we see that (2.2) is automatically satisfied through the definition of ψ.

Further, to simplify our notation a bit and to be more consistent with Wang et al. [1991], we introduce the binary differential operator $\mathcal{J}$. This operator is defined for two scalar functions a and b as

$$\mathcal{J}(a,b) = \frac{\partial a}{\partial y}\frac{\partial b}{\partial x} - \frac{\partial b}{\partial y}\frac{\partial a}{\partial x}. \tag{4.22}$$

Consider now the advection-diffusion equation (2.12) on page 8, which is valid within each phase. In steady-state, that is when $\partial_t T = 0$, we can restate it in terms of $\mathcal{J}$ and ψ as

$$\mathcal{J}(\psi, T) = \alpha \nabla^2 T. \tag{4.23}$$

Next, we consider the momentum balance equation (2.6) on page 6, also valid within each phase. In steady-state, $\partial_t \vec{u} = 0$, and (2.6) becomes

$$(\vec{u}\cdot\nabla)\,\vec{u} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\vec{u} + \vec{g}. \tag{4.24}$$

Applying $\nabla\times$ on both sides of (4.24), we get [Landau and Lifshitz, 1987, §15]

$$\nabla\times(\vec{u}\times\{\nabla\times\vec{u}\}) + \nu\nabla^2(\nabla\times\vec{u}) + \nabla\times\vec{g} = 0. \tag{4.25}$$

Considering the first term, we employ standard vector calculus relations [Rottmann, 2003] and (2.2) to rewrite it as

$$\begin{aligned}
\nabla\times(\vec{u}\times\{\nabla\times\vec{u}\}) &= (\{\nabla\times\vec{u}\}\cdot\nabla)\,\vec{u} - (\vec{u}\cdot\nabla)\,\{\nabla\times\vec{u}\} \\
&\quad + \vec{u}\,(\nabla\cdot\{\nabla\times\vec{u}\}) - \{\nabla\times\vec{u}\}\,(\nabla\cdot\vec{u}), && (4.26)\\
&= (\{\nabla\times\vec{u}\}\cdot\nabla)\,\vec{u} - (\vec{u}\cdot\nabla)\,\{\nabla\times\vec{u}\}, && (4.27)
\end{aligned}$$

and then (4.25) becomes

$$(\vec{u}\cdot\nabla)\,\{\nabla\times\vec{u}\} - (\{\nabla\times\vec{u}\}\cdot\nabla)\,\vec{u} = \nu\nabla^2\{\nabla\times u\} + \nabla\times\vec{g}. \tag{4.28}$$

We can easily rewrite $\nabla\times\vec{u}$ in terms of the stream function ψ,

$$\nabla\times\vec{u} = -\nabla^2\psi\hat{z}. \tag{4.29}$$

Now, using that $\vec{g} = -g\hat{y}$ and (4.29) before dividing by $\hat{z}$, we can recast (4.28) as an equation for ψ

$$\frac{\partial \psi}{\partial y}\frac{\partial}{\partial x}\left(\nabla^2\psi\right) - \frac{\partial \psi}{\partial x}\frac{\partial}{\partial y}\left(\nabla^2\psi\right) = \nu\nabla^2\left(\nabla^2\psi\right) + \frac{\partial g}{\partial x}, \tag{4.30}$$

which is equivalent to

$$\mathcal{J}\left(\psi, \nabla^2\psi\right) = \nu\nabla^2\left(\nabla^2\psi\right) + \frac{\partial g}{\partial x}. \tag{4.31}$$

Finally, employing the Boussinesq approximation (2.16) on page 9, we get

$$\begin{aligned} \mathcal{J}\left(\psi, \nabla^2\psi\right) &= \nu\nabla^2\left(\nabla^2\psi\right) + \frac{\partial}{\partial x} g_\infty \left(1 - \beta\left\{T - T_\infty\right\}\right), & (4.32)\\ &= \nu\nabla^2\left(\nabla^2\psi\right) - \beta g_\infty \frac{\partial T}{\partial x}. & (4.33) \end{aligned}$$

We now have two coupled PDEs, (4.23) and (4.33), for ψ and T and we give their solutions different names in the two phases,

$$\psi = \begin{cases} \psi^+ & \text{if } (x,y) \in \Omega^+ \\ \psi^- & \text{if } (x,y) \in \Omega^- \end{cases}, \tag{4.34}$$

$$T = \begin{cases} T^+ & \text{if } (x,y) \in \Omega^+ \\ T^- & \text{if } (x,y) \in \Omega^- \end{cases}. \tag{4.35}$$

In order to use expressions for ψ and T that are notationally easily comparable to those from Wang et al. [1991], we introduce the relative fluid properties

$$\bar{\mu} = \frac{\mu^+}{\mu^-},\ \bar{\kappa} = \frac{\kappa^+}{\kappa^-},\ \bar{\alpha} = \frac{\alpha^+}{\alpha^-}, \tag{4.36}$$

and the dimensionless Rayleigh and Prandtl numbers

$$Ra^+ = \frac{g_\infty \beta^+ q L_y^4}{\nu^+\alpha^+\kappa^+}, \qquad Pr^+ = \frac{\nu^+}{\alpha^+}, \tag{4.37}$$

$$Ra^- = \frac{g_\infty \beta^- q L_y^4}{\nu^-\alpha^-\kappa^-}, \qquad Pr^- = \frac{\nu^-}{\alpha^-}. \tag{4.38}$$

We also introduce dimensionless quantities $\tilde{x}$, $\tilde{y}$, $\tilde{u}$, $\tilde{\psi}$ and $\tilde{T}$,

$$\tilde{x} = \frac{1}{L_y}x,\ \tilde{y} = \frac{1}{L_y}y,\ \tilde{\vec{u}} = \frac{L_y}{\alpha^-}\vec{u},\ \tilde{\psi} = \frac{1}{\alpha^-}\psi,\ \tilde{T} = \frac{\kappa^-}{qL_y}T. \tag{4.39}$$

and obtain dimensionless PDEs for $\tilde{\psi}$ and $\tilde{T}$,

$$\mathcal{J}\left(\tilde{\psi}^-, \nabla^2\tilde{\psi}^-\right) = Pr^-\left\{\nabla^2\left(\nabla^2\tilde{\psi}^-\right) - Ra^-\frac{\partial\tilde{T}^-}{\partial\tilde{x}}\right\}, \tag{4.40}$$

$$\mathcal{J}\left(\tilde{\psi}^+, \nabla^2\tilde{\psi}^+\right) = \bar{\alpha}\ Pr^+\left\{\nabla^2\left(\nabla^2\tilde{\psi}^+\right) - \bar{\alpha}\bar{\kappa}Ra^+\frac{\partial\tilde{T}^+}{\partial\tilde{x}}\right\}, \tag{4.41}$$

$$\mathcal{J}\left(\tilde{\psi}^-, \tilde{T}^-\right) = \nabla^2\tilde{T}^-, \tag{4.42}$$

$$\mathcal{J}\left(\tilde{\psi}^+, \tilde{T}^+\right) = \bar{\alpha}\ \nabla^2\tilde{T}^+. \tag{4.43}$$

The spatial derivatives in these equations are now, of course, taken with respect to the dimensionless coordinates $\tilde{x}$ and $\tilde{y}$.

The Boundary and Interface Conditions

We pose no-slip BCs on the velocity field. Thus,

$$\tilde{u} = \frac{\partial\tilde{\psi}}{\partial\tilde{y}} = 0, \tag{4.44}$$

$$\tilde{v} = -\frac{\partial\tilde{\psi}}{\partial\tilde{x}} = 0, \tag{4.45}$$

at the boundaries.

Since $\tilde{\psi}$ is continuous, otherwise the velocity field would not be well-defined, $\tilde{\psi}$ must have the same constant value along the entire boundary. Then, since the velocity is invariant upon the addition of a constant to $\tilde{\psi}$, we are free to choose the zero-level of $\tilde{\psi}$ such that

$$\tilde{\psi} = 0 \tag{4.46}$$

at the boundary.

At the interface, the velocity field must be continuous and well-defined. Also, the shear stress must be continuous. Assuming that the interface is stationary and located at $\tilde{y} = \eta$, we get the following conditions on $\tilde{\psi}$

$$\tilde{\psi}^- = \tilde{\psi}^+, \tag{4.47}$$

$$\frac{\partial \tilde{\psi}^-}{\partial \tilde{y}} = \frac{\partial \tilde{\psi}^+}{\partial \tilde{y}}, \tag{4.48}$$

$$\frac{\partial \tilde{\psi}^-}{\partial \tilde{x}} = \frac{\partial \tilde{\psi}^+}{\partial \tilde{x}} = 0, \tag{4.49}$$

$$\frac{\partial^2 \tilde{\psi}^-}{\partial \tilde{y}^2} = \bar{\mu} \frac{\partial^2 \tilde{\psi}^+}{\partial \tilde{y}^2}, \tag{4.50}$$

at $\tilde{y} = \eta$.

Since the north and south walls are insulating and there is constant and uniform heat flux density through the east and west walls,

$$\frac{\partial \tilde{T}^-}{\partial \tilde{y}} = 0, \quad \text{at } \tilde{y} = 0, \tag{4.51}$$

$$\bar{\kappa} \frac{\partial \tilde{T}^+}{\partial \tilde{y}} = 0, \quad \text{at } \tilde{y} = 1, \tag{4.52}$$

$$\frac{\partial \tilde{T}^-}{\partial \tilde{x}} = -1, \quad \text{at } \tilde{x} = \pm A/2, \tag{4.53}$$

$$\bar{\kappa} \frac{\partial \tilde{T}^+}{\partial \tilde{x}} = -1, \quad \text{at } \tilde{x} = \pm A/2. \tag{4.54}$$

At the interface, we must have continuity of the temperature and the heat flux density. Thus,

$$\tilde{T}^- = \tilde{T}^+, \tag{4.55}$$

$$\frac{\partial \tilde{T}^-}{\partial \tilde{y}} = \bar{\kappa} \frac{\partial \tilde{T}^+}{\partial \tilde{y}}, \tag{4.56}$$

at $\tilde{y} = \eta$.

The Solution

In the parallel flow approximation [Wang et al., 1991], we assume that

$$\tilde{\psi}^- = \tilde{\psi}^- (\tilde{y}), \tag{4.57}$$

$$\tilde{\psi}^{+} = \tilde{\psi}^{+}(\tilde{y}), \tag{4.58}$$
$$\tilde{T}^{-} = C_0^{-} + C_1^{-}\tilde{x} + \Theta^{+}(\tilde{y}), \tag{4.59}$$
$$\tilde{T}^{+} = C_0^{+} + C_1^{+}\tilde{x} + \Theta^{-}(\tilde{y}). \tag{4.60}$$

From the form of these equations, we see that we must have $C_0^- = C_0^+ = C_0$ and $C_1^- = C_1^+ = C_1$ to have continuity of temperature at the interface.

In the parallel flow approximation, the PDEs for $\tilde{\psi}$ and $\tilde{T}$ reduce to the following ordinary differential equations (ODEs)

$$\frac{d^4\tilde{\psi}^{-}}{d\tilde{y}^4} = Ra^{-}C_1, \tag{4.61}$$
$$\frac{d^4\tilde{\psi}^{+}}{d\tilde{y}^4} = \bar{\alpha}\bar{\kappa}Ra^{+}C_1, \tag{4.62}$$
$$\frac{d^2\Theta^{-}}{d\tilde{y}^2} = \frac{d\tilde{\psi}^{-}}{dy}C_1, \tag{4.63}$$
$$\frac{d^2\Theta^{+}}{d\tilde{y}^2} = \frac{1}{\bar{\alpha}}\frac{d\tilde{\psi}^{+}}{dy}C_1. \tag{4.64}$$

By differentiation and insertion it can be confirmed that

$$\tilde{\psi}^{-} = \frac{Ra^{-}C_1}{24}\tilde{y}^2(\tilde{y}-\eta)\left(\tilde{y}+\eta+\Lambda^{-}\right), \tag{4.65}$$
$$\tilde{\psi}^{+} = \frac{Ra^{-}C_1}{24}\{1-\tilde{y}\}^2(\eta-\tilde{y})\left(\frac{\bar{\rho}\bar{\beta}}{\bar{\mu}}\{2-\tilde{y}-\eta\}+\Lambda^{+}\right), \tag{4.66}$$
$$\Theta^{-} = \frac{Ra^{-}(C_1)^2}{24}\left\{\left(\frac{\tilde{y}^5-\eta^5}{5}-\eta^2\frac{\tilde{y}^3-\eta^3}{3}\right) + \Lambda^{-}\left(\frac{\tilde{y}^4-\eta^4}{4}-\eta\frac{\tilde{y}^3-\eta^3}{3}\right)\right\}, \tag{4.67}$$
$$\Theta^{+} = -\frac{Ra^{-}(C_1)^2}{24\bar{\alpha}}\left\{\frac{\bar{\rho}\bar{\beta}}{\bar{\mu}}\left(\frac{\{1-\tilde{y}\}^5-\{1-\eta\}^5}{5} - (1-\eta)^2\frac{\{1-\tilde{y}\}^3-\{1-\eta\}^3}{3}\right) + \Lambda^{+}\left(\frac{\{1-\tilde{y}\}^4-\{1-\eta\}^4}{4} - (1-\eta)\frac{\{1-\tilde{y}\}^3-\{1-\eta\}^3}{3}\right)\right\}, \tag{4.68}$$

where

$$\Lambda^- = \frac{\bar{\rho}\bar{\beta}\left\{1-\eta\right\}^3 - \eta^2\left(4\bar{\mu}\eta + 5\left\{1-\eta\right\}\right)}{2\eta\left(\left\{1-\eta\right\} + \bar{\mu}\eta\right)}, \tag{4.69}$$

$$\Lambda^+ = \frac{\eta^3 - \frac{\bar{\rho}\bar{\beta}}{\bar{\mu}}\left\{1-\eta\right\}^2\left(4\left\{1-\eta\right\} + 5\bar{\mu}\eta\right)}{2\left\{1-\eta\right\}\left(\left\{1-\eta\right\} + \bar{\mu}\eta\right)}, \tag{4.70}$$

are solutions to these ODEs with the posed boundary conditions. Using the definition of ψ (4.20), we get expressions for the velocity

$$u^-(\tilde{y}) = \frac{Ra^- C_1}{24}\left\{4\tilde{y}^3 + 3\tilde{y}^2\Lambda^- - 2\tilde{y}\eta\left(\eta + \Lambda^-\right)\right\}, \tag{4.71}$$

$$\begin{aligned} u^+(\tilde{y}) = \frac{Ra^- C_1}{24}\Bigg\{ & \frac{\bar{\rho}\bar{\beta}}{\bar{\mu}}\left(1-\tilde{y}\right)^2\left(\tilde{y}-\eta\right) \\ & + \left(-3\tilde{y}^2 + 2\tilde{y}\left\{2+\eta\right\} - 1 - 2\eta\right) \\ & \left(\frac{\bar{\rho}\bar{\beta}}{\bar{\mu}}\left\{2-\tilde{y}-\eta\right\} + \Lambda^+\right)\Bigg\}. \end{aligned} \tag{4.72}$$

In order to determine the coefficient C_1, we must demand that the sum of the advective and conductive heat transfer across any constant-x line is equal to the heat transfer through the side walls. This yields the integral equation

$$\int_0^\eta \tilde{u}^-\Theta^- d\tilde{y} + \frac{\bar{\kappa}}{\bar{\alpha}}\int_\eta^1 \tilde{u}^+\Theta^+ d\tilde{y} - C_1\left(\eta + \bar{\kappa}\left\{1-\eta\right\}\right) = 1. \tag{4.73}$$

This equation was solved numerically for C_1 using Simpson's method [Süli and Mayers, 2006, Section 7.5] to evaluate the integrals and the bisection method [Süli and Mayers, 2006, Section 1.6] to find the appropriate root of the resulting polynomial. The coefficient C_0, equal to the dimensionless temperature at the interface, is undetermined by the posed boundary conditions.

4.3.2 Numerical Results

The analytical solution for this case is, strictly speaking, valid only when $A \to \infty$. However, Oueslati et al. [2012] claimed that using $A = 8$, as we do, was sufficient to accurately simulate the situation considered in the

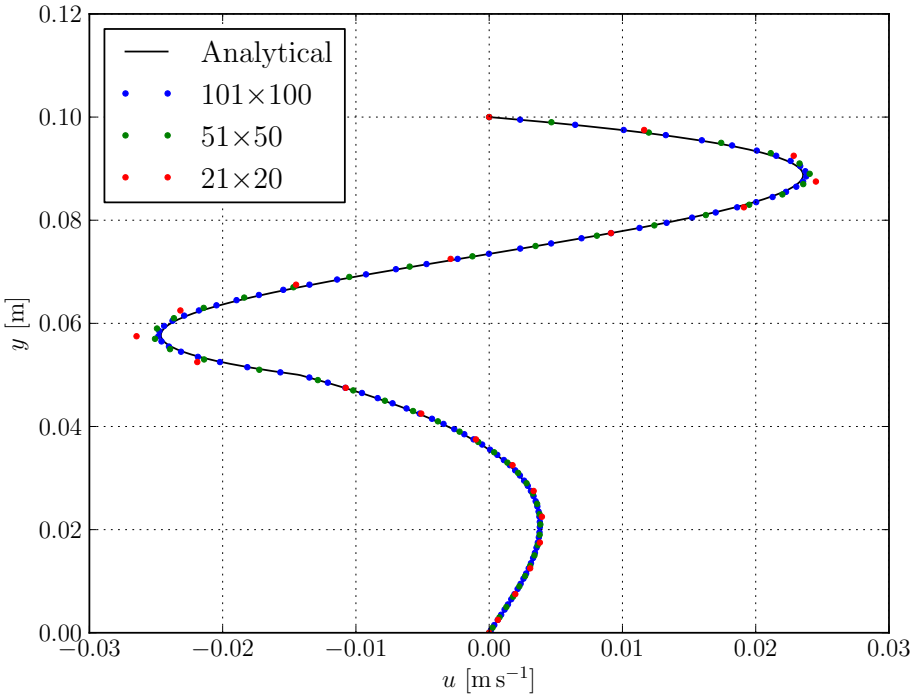


Figure 4.7: Comparison of selected numerically obtained velocity profiles along the vertical center line in Wang et al.'s two-layer convection case with the analytical solution. It is evident that the numerical solutions approach the analytical solution upon grid refinement.

steady-state analytical solution. It was verified that $E_{\mathrm{ss}}(u)$ did not exceed $10^{-7}\,\mathrm{m\,s^{-1}}$ in magnitude.

The resulting horizontal velocity component along the vertical center line of the cavity is shown in Figure 4.7 for a selection of grid sizes. We observe a buoyancy-driven convection cell in the top layer (Phase 2), similar to that in the de Vahl Davis case, where the fluid moves clockwise, and we see a viscosity-driven cell in the bottom (Phase 1), where the fluid moves anti-clockwise. To plotting accuracy, there is good agreement between the analytical solution and the numerical results for the 101×100 grid. Furthermore, the numerical solutions appear to approach the analytical solution when the grid is refined.

Table 4.6: Convergence table based on the maximum horizontal velocity component $u_{\max}$ along the vertical center line for Wang et al.'s two-layer convection case.

N_y -	$u_{\max}$ $\mathrm{m\,s^{-1}}$	E_{abs} $\mathrm{m\,s^{-1}}$	n -
20	$2.487 \cdot 10^{-02}$	$1.31 \cdot 10^{-03}$	-
30	$2.442 \cdot 10^{-02}$	$8.66 \cdot 10^{-04}$	1.03
40	$2.421 \cdot 10^{-02}$	$6.49 \cdot 10^{-04}$	1.01
50	$2.405 \cdot 10^{-02}$	$4.96 \cdot 10^{-04}$	1.20
60	$2.395 \cdot 10^{-02}$	$3.97 \cdot 10^{-04}$	1.21
80	$2.386 \cdot 10^{-02}$	$3.02 \cdot 10^{-04}$	0.95
100	$2.380 \cdot 10^{-02}$	$2.43 \cdot 10^{-04}$	0.97

To do a quantitative comparison between the analytical solution and the numerical results, we consider the maximum of the horizontal velocity component $u_{\max}$ along the vertical center line. As in the de Vahl Davis-case, we locate the $u_{\max}$ from the numerical results by doing a fourth-order polynomial interpolation. In the analytical solution, $u_{\max}$ was found to be $2.3557 \cdot 10^{-2}\,\mathrm{m\,s^{-1}}$. A convergence table based on $u_{\max}$ for different grid sizes, with the analytical value as reference, is shown in Table 4.6. According to the estimated orders of convergence n from this table, the convergence to the analytical solution upon grid refinement is of first order. This is expected since the GFM is a first order method. The convergence to the analytical solution is also illustrated in Figure 4.8.

From this test case we can conclude that our implementation produced results in agreement with the analytical steady-state solution for a case involving temperature-driven two-phase flow. Also, the performed convergence study showed the expected first-order convergence. This gives reason to believe that the current implementation of the discretization of the advection-diffusion equation and the Boussinesq coupling to the Navier–Stokes equations is correct also for two-phase cases where the interface is treated with the GFM.

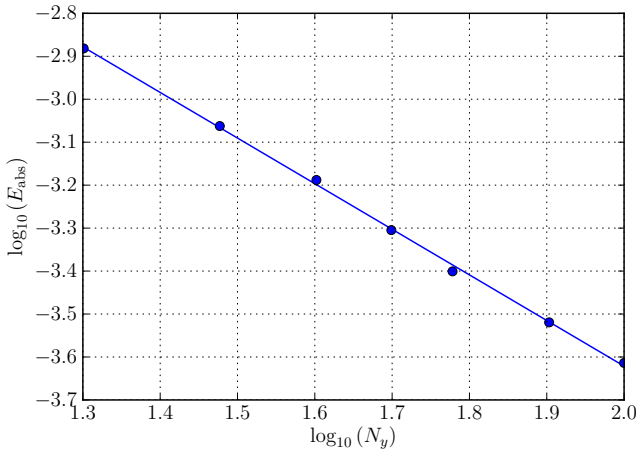


Figure 4.8: Log-log plot of the error E_{abs} in $u_{\max}$ against the linear grid size N_y in Wang et al.'s two-layer convection case. The solid line represents a first-order polynomial fit and has a slope of approximately -1.06.

4.4 Lava Lamp

In this case we consider a situation which is very similar to that found in commercial lava lamps. In these decorative contraptions, two immiscible fluids are contained in a closed vessel. At room temperature, one of the fluids (Phase 1) is denser than the other (Phase 2), but with a larger coefficient of thermal expansion. For visual effect, Phase 1 is usually opaque and dyed with some color while Phase 2 is more transparent. As the fluids are heated from below, Phase 1 eventually becomes less dense than Phase 2 and, under the right conditions, it forms a rising *blob*[‡]. The blob is subsequently cooled by the surrounding Phase 1 and the container walls. Eventually, it drops and the cycle starts over, creating the mesmerizing and almost hypnotic effect that we are used to seeing in lava lamps.

Recently, Gyüre and Jánosi [2009] conducted an experimental study on silicone oil in a saline solution to describe the mechanisms of lava lamp convection. Under specific conditions, they observed *blob exchange*

[‡]The author is uncertain as to whether the term "blob" has a well-defined scientific meaning or not. Still, it is used in the lava lamp literature [Gyüre and Jánosi, 2009] and therefore also in this work.

Table 4.7: Fluid parameters used in the lava lamp case.

	Unit	Phase 1	Phase 2
ρ	$\mathrm{kg\,m^{-3}}$	1080	1074
μ	$\mathrm{Pa\,s}$	0.05	$1.08 \cdot 10^{-3}$
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	1591	3993
κ	$\mathrm{W\,m^{-1}\,K^{-1}}$	0.146	0.596
β	$\mathrm{K^{-1}}$	$470 \cdot 10^{-6}$	$280 \cdot 10^{-6}$
T_∞	K	293	293

oscillations in which a blob of Phase 1 would rise from the bottom of the vessel, attach itself to the top, be cooled and sink down, with regular time periods. We will use the similar parameters to those of Gyüre and Jánosi [2009], as far as our model permits it, and try to replicate the oscillating behavior in our simulations.

The simulation was run on an axisymmetric 75×168 grid. With this grid, the domain is a cylinder. The eastern wall represents its entire outer face and the western wall its central axis of symmetry. The height of the cylinder was $L_y = 0.28\,\mathrm{m}$ and the radius was $L_x = 0.125\,\mathrm{m}$. The coefficient of surface tension was set to $\sigma = 0.04\,\mathrm{N\,m^{-1}}$, the gravitational acceleration to $\vec{g}_\infty = \left(-9.81\,\mathrm{m\,s^{-2}}\right)\hat{y}$ and the rest of the relevant fluid parameters are given in Table 4.7.

We imposed no-slip BCs for the velocity and Dirichlet for the temperature at all walls, except the western which, due to symmetry, needed to have free-slip and be thermally insulating. On the north wall we set the temperature to $T_{\mathrm{north}} = 312\,\mathrm{K}$, on the south wall we set $T_{\mathrm{south}} = 332\,\mathrm{K}$ and on the eastern we set $T_{\mathrm{east}} = 321.9\,\mathrm{K}$. The fluids were initially at rest with a uniform temperature of 321.9 K. The initial interface was a spherical shell of radius 0.0829 m, centered in the south east corner.

4.4.1 Numerical Results

The simulated interface location is shown in Figure 4.9 at different times. From this figure, we can observe the aforementioned blob exchange oscillations. According to the experimental measurements by Gyüre and

Jánosi [2009], Figure 9, we should get that the time period of the oscillations is between $9.2 \cdot 10^2$ s and $1.8 \cdot 10^3$ s. The period of the simulated oscillations was found to be $1.4 \cdot 10^3$ s, and thus in good agreement with the experimental results.

However, in spite of having similar time periods, we see, e.g. from comparing Figure 4.9 with Figure 4 from Gyüre and Jánosi [2009], that the simulated blob oscillations do not look the same as those in the experiments. The reasons for the deviations are probably many, as there are many aspects of lava lamp convection that are not accounted for in our model.

For instance, our model assumes uniform viscosities in both phases. However, according to Gyüre and Jánosi [2009], the viscosity of the oil (Phase 1) is almost exponentially decreasing with increasing temperature. This strongly temperature-dependent viscosity is not accounted for by our model.

Also, we choose constant temperature boundary conditions as the driving force for the lava lamp convection. As e.g. the temperature plot Figure 5 from Gyüre and Jánosi [2009] shows, this might not be realistic. The boundary conditions used here were chosen to guarantee fluid domain temperatures in the region around where Phase 1 and Phase 2 are equally dense.

The boundary conditions that we use for the level-set function give a contact angle between the interface and the boundary walls that is always 90°. As we see from Figure 4 in Gyüre and Jánosi [2009], the oil (Phase 1) wets the walls of the vessel slightly in the experiments and the 90° contact angle from the simulations is therefore not correct. The boundary conditions on the level-set function are chosen the way they are, and not with the aid of a force balance equation like (61.12) from Landau and Lifshitz [1987], because we have not included any material properties of domain walls in our model.

From this test case, we can conclude that our two-phase flow model reproduced the experimentally observed blob exchange oscillations and their time periods. This gives reason to believe that the implementation was correct also on axisymmetric grids. Still, the simulated behavior can not be said to be in full agreement with the experimental results and neither can we expect it to be, since many of the physical aspects of lava lamp convection that were presented by Gyüre and Jánosi [2009] are not

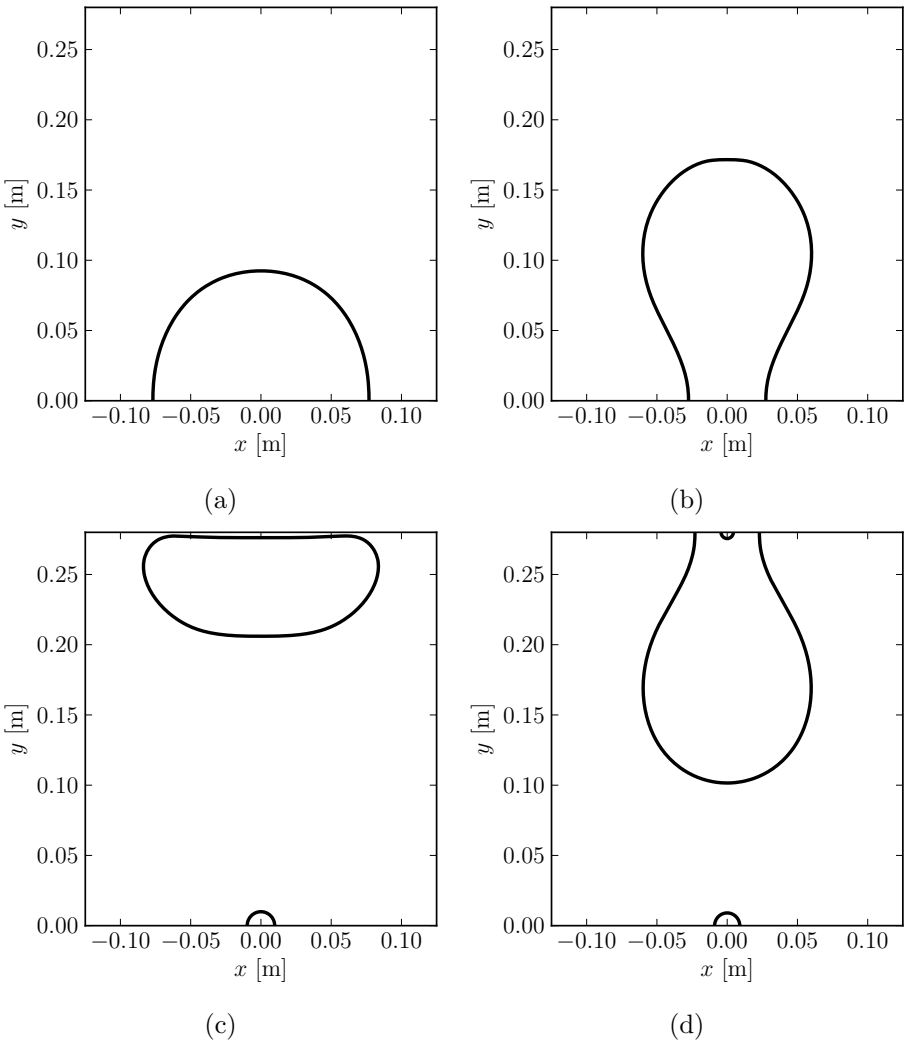


Figure 4.9: The interface location in the lava lamp case at (a) $t = 500\,\mathrm{s}$, where the increased temperature in Phase 1 causes its shape to deviate slightly from a the initial half-sphere, (b) at $t = 766\,\mathrm{s}$, where a rising blob has formed, (c) at $t = 775\,\mathrm{s}$, where the rising blob attaches itself to the top of the vessel, and (d) at $t = 1214\,\mathrm{s}$, where the blob has been cooled and sinks back down.

accounted for in our model.

4.5 Vaporization with Uniform Mass Flux

As a first test of the phase-transition model and its ability to preserve the discontinuities in $\vec{u}$ and p at the interface, a one-dimensional case with uniform mass flux density m was simulated. We emphasize that in this particular case m was set to a given value, initially $2.0\,\mathrm{kg\,m^{-2}\,s^{-1}}$, and not calculated from the temperature field.

We consider a fluid domain with length $L = 2.0\,\mathrm{m}$. In the initial configuration, we have a liquid film in the middle of the domain (Phase 1), with interfaces to the gaseous phase (Phase 2) at $x = 0.5\,\mathrm{m}$ and at $x = 1.5\,\mathrm{m}$. We initialize the liquid film to be at rest, we set $\vec{u}^- = 0\,\mathrm{m\,s^{-1}}$, and we initialize the gas velocities as $\vec{u}^+ = \left(1.0\,\mathrm{m\,s^{-1}}\right)\hat{n}$, so that they satisfy the jump condition (2.38) on page 14. The relevant fluid properties are given in Table 4.8.

On both the eastern and western boundaries, we impose outflow BCs on the velocities and Dirichlet BCs on the pressure. On the northern and the southern boundaries we pose free-slip BCs on the velocity and Neumann BCs on the pressure. Even though the simulations were performed on a two-dimensional 21×21 grid, the initial and boundary conditions make this an effective one-dimensional problem and the simulated results will be independent of y.

The simulation was run for two slightly different cases A and B. Case A was run from $t = 0\,\mathrm{s}$ to $t = 0.15\,\mathrm{s}$ with $m = 2.0\,\mathrm{kg\,m^{-2}\,s^{-1}}$ for the entire run and Case B from $t = 0\,\mathrm{s}$ to $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$ where m was increased to $2.1\,\mathrm{kg\,m^{-2}\,s^{-1}}$ after $t = 5.0 \cdot 10^{-5}\,\mathrm{s}$. To simplify the analysis of the time stepping, the forward Euler method with a constant time-step length of $\Delta t = 2 \cdot 10^{-5}\,\mathrm{s}$ was used. It was confirmed that this time-step length was consistent with the CFL criteria from Section 3.7.

4.5.1 Analytical Solution

Let us consider Case A first. Due to the initial configuration, the time derivatives of $\vec{u}^+$ and $\vec{u}^-$ are zero at all times and the velocities will remain $\vec{u}^- = 0\,\mathrm{m\,s^{-1}}$ and $\vec{u}^+ = \left(1.0\,\mathrm{m\,s^{-1}}\right)\hat{n}$. However, because there is mass transport from the liquid to the gaseous phase, the liquid film will shrink

Table 4.8: Fluid parameters used in the vaporization with uniform mass flux case.

	Unit	Phase 1	Phase 2
ρ	$\mathrm{kg\,m^{-3}}$	2.0	1.0
μ	$\mathrm{Pa\,s}$	$1.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$

and the interfaces will move from their initial positions with velocity $\vec{w} = \left(-1.0\,\mathrm{m\,s^{-1}}\right)\hat{n}$.

Since the velocity field is uniform on both sides of the interface and there is no interface curvature, the only contribution to the pressure jump (2.39) from page 14 is $-m^2\,[1/\rho]$ and thus $[p] = -2.0\,\mathrm{Pa}$. This means that the pressure in the liquid will be constant and equal to $2.0\,\mathrm{Pa}$ and the pressure in the gas will be constant and equal to $0.0\,\mathrm{Pa}$.

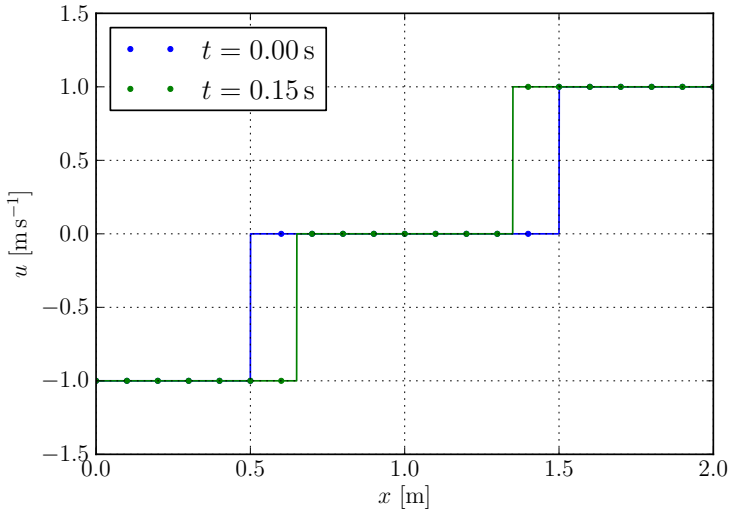
Case B will develop similarly to Case A until m is increased at $t = 5.0 \cdot 10^{-5}\,\mathrm{s}$. After this time, the gas velocities at the interface will have to be $\left(1.05\,\mathrm{m\,s^{-1}}\right)\hat{n}$ to satisfy the jump condition (2.38). Because the fluids are incompressible, the change in velocity at the interface should be instantly propagated through the entire domain. Therefore we should have $\vec{u}^{+} = \left(1.05\,\mathrm{m\,s^{-1}}\right)\hat{n}$ in the entire gas phase, not just at the interface, immediately after m has increased.

4.5.2 Numerical Results

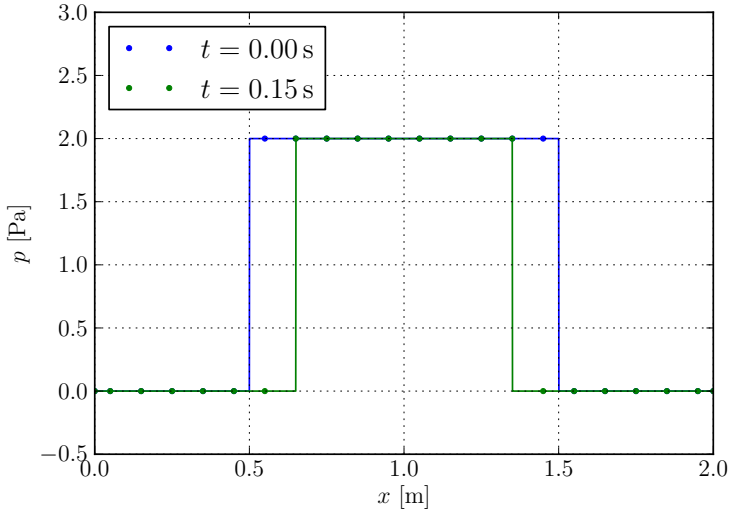
The simulated velocity and pressure fields from Case A at $t = 0.0\,\mathrm{s}$ and at $t = 0.15\,\mathrm{s}$ are shown, together with the analytical solution, in Figure 4.10.

We observe that the implementation shows correct behavior [Gibou et al., 2007] in that it has preserved the sharp jumps in $\vec{u}$ and p at the interface even as the grid nodes change character from liquid to gas.

Further, we observe that there is agreement between the results from the simulation and the exact analytical results, to plotting accuracy. More quantitatively, the error E_1 in the pressure field was $2.33 \cdot 10^{-10}\,\mathrm{Pa}$ and E_1 in the velocity field $5.32 \cdot 10^{-14}\,\mathrm{m\,s^{-1}}$ with respect to the analytical solution at $t = 0.15\,\mathrm{s}$. The implementation was therefore able, even on this coarse 21×21 grid, to produce results whose accuracy was limited by machine precision and the tolerance in the Poisson solver.



(a)



(b)

Figure 4.10: Plot of (a) the velocity profile and (b) the pressure field in the initial configuration $t = 0.0\,\mathrm{s}$ and at $t = 0.15\,\mathrm{s}$ in the vaporization with uniform mass flux Case A. The analytical solution is shown as solid lines.

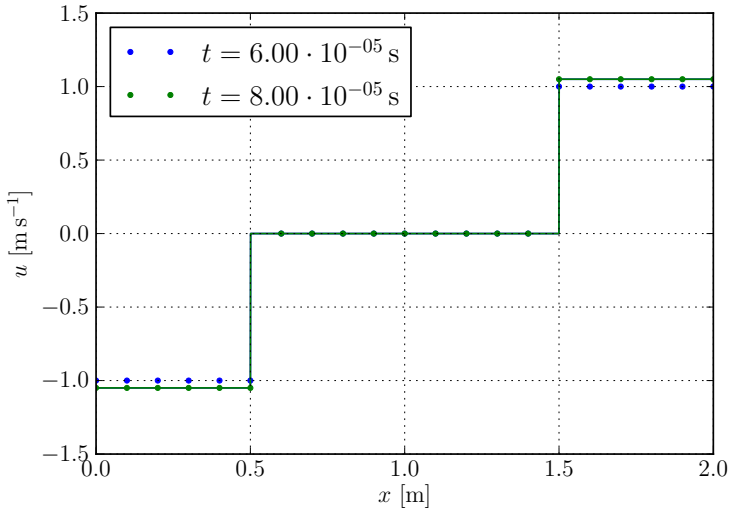
The velocity fields from Case B at $t = 6.0 \cdot 10^{-5}\,\mathrm{s}$, at $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$ and the analytical solution are shown in Figure 4.11a. From this figure, we immediately see that the simulated velocity field at $t = 6.0 \cdot 10^{-5}\,\mathrm{s}$ does not agree with the analytical solution. This must, however, be expected due to the explicit method. The velocity profile at $t = 6.0 \cdot 10^{-5}\,\mathrm{s}$ was calculated with $m = 2.0\,\mathrm{kg\,m^{-2}\,s^{-1}}$ from the previous time step, at $t = 4.0 \cdot 10^{-5}\,\mathrm{s}$. The new value $m = 2.1\,\mathrm{kg\,m^{-2}\,s^{-1}}$ was not used until $t = 6.0 \cdot 10^{-5}\,\mathrm{s}$ to calculate the velocity at $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$. Therefore, the velocity at $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$ is the first to be consistent with the new m. The error E_1 in the velocity field at $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$ was $4.38 \cdot 10^{-6}\,\mathrm{m\,s^{-1}}$.

The pressure field at $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$ is shown in Figure 4.11b. Its large numerical value with respect to the pressure fields in Figure 4.11a alone is enough to make it seem strange and rouse suspicion. Still, it may be explained as follows. When m changes from one time step to the next and we set the ghost values according to (3.104) and (3.105) from page 43, then the gas ghost velocity $\vec{u}_{\mathrm{g}}^{-}$ in the liquid has increased with respect to the gas velocity on the gas side and thus we no longer have $\vec{u}$ divergence-free. The pressure field must then adapt to restore the divergence-freeness, as we demand in the projection step (3.1), and the result is what we see in Figure 4.11b.

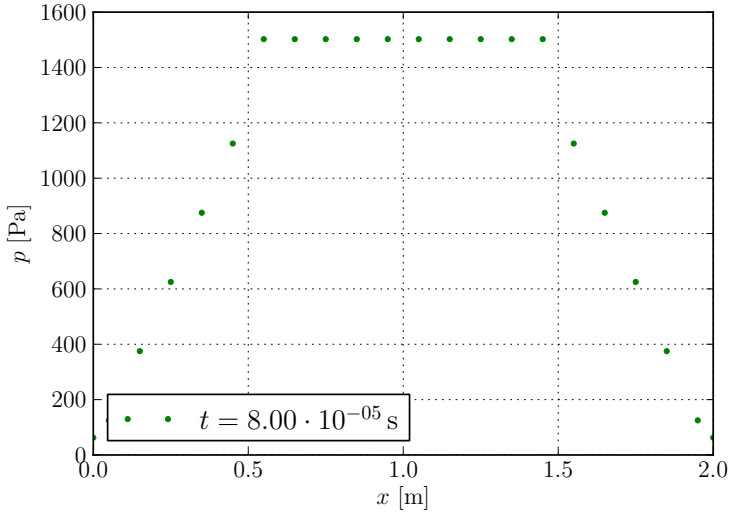
This speculation can be supported by evaluating $\partial_x p/\rho$ anywhere on the interior of the gas phase west of the film. Doing so, we get approximately $2500\,\mathrm{m\,s^{-2}}$. This is consistent with the $\partial_t u$ needed here to get a uniform velocity field and thus enforce the divergence-freeness of $\vec{u}$, namely $\left(0.05\,\mathrm{m\,s^{-1}}\right) / \left(2.0 \cdot 10^{-5}\,\mathrm{s}\right) = 2500\,\mathrm{m\,s^{-2}}$.

The strange pressure field at $t = 8.0 \cdot 10^{-5}\,\mathrm{s}$ is therefore probably an artifact of the incompressible model. Its strange shape and large numerical values are necessary to mediate the information about the new velocity jump at the interface, caused by the new mass flux value, to the entire domain so that the velocity field keeps its divergence-freeness.

We should also mention that this Case B, with a large, instantaneous jump in m is not physically realistic. If it were, m would be determined from the spatial derivatives of the temperature field. These should be continuous in time and space and should therefore not produce such large jumps. Still, we must expect m to fluctuate and information about these fluctuations must be instantaneously propagated to the entire domain to keep the velocity field divergence free. This somewhat exaggerated exam-



(a)



(b)

Figure 4.11: Plot of (a) the velocity field at $t = 6.0 \cdot 10^{-5}$ s and at $t = 8.0 \cdot 10^{-5}$ s and (b) the pressure field at $t = 8.0 \cdot 10^{-5}$ s in the vaporization with uniform mass flux Case B.

Table 4.9: Fluid parameters used in the numerical simulation of vaporization with physical mass flux case.

	Unit	Phase 1	Phase 2
ρ	$\mathrm{kg\,m^{-3}}$	1.0	0.2
μ	$\mathrm{Pa\,s}$	$1.0 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	1.0	5.0
κ	$\mathrm{W\,K^{-1}\,m^{-1}}$	1.0	1.0

ple indicates that the implementation is robust enough to handle this.

From this test case we can conclude that the current implementation succeeded in preserving the discontinuities in $\vec{u}$ and p, even when grid nodes changed character from liquid to gas, in the case of uniform mass flux density m. It also succeeded in propagating an instantaneous change in the velocity field at the interface to the entire domain instantaneously as is necessary in the incompressible model. However, this reminds us of the fact that the pressure in the incompressible model is not a thermodynamic state variable, but merely a field that we choose to be whatever is needed to keep the velocity field divergence-free.

4.6 Vaporization with Physical Mass Flux

To test the phase-transition model with a physical mass flux density m, calculated from the temperature profile, another simple one-dimensional case was simulated. This case was chosen partly because of its simplicity, but mostly because it has an analytical solution for the temperature and velocity profiles and for the interface location to which the results from the simulations could be compared.

We consider a one-dimensional fluid domain of length $L = 1.0\,\mathrm{m}$, occupied by a liquid (Phase 1) and a gaseous phase (Phase 2) of the same substance. The two phases are separated by an interface located at $x = \eta(t)$, as shown in Figure 4.12. The specific enthalpy difference associated with the phase transition is $\Delta h = -1.0 \cdot 10^{-3}\,\mathrm{J\,kg^{-1}}$. The other relevant fluid parameters are given in Table 4.9.

The northern, southern and western walls have free-slip BCs and the

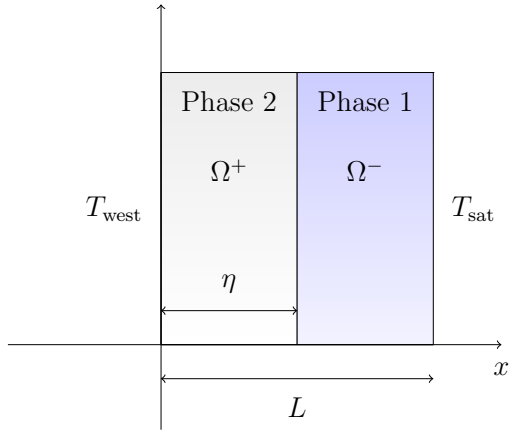


Figure 4.12: Schematic drawing of the initial fluid configuration in the vaporization with physical mass flux case. Phase 1 is the liquid and Phase 2 is the gaseous phase. The east wall is held at the saturation temperature T_{sat} and the west wall is held at $T_{\text{west}} > T_{\text{sat}}$.

eastern wall has an outflow condition. When solving the Poisson equation for the pressure, we impose Neumann BCs on the pressure at all walls except the eastern, where we demand $p = 0\,\text{Pa}$. As boundary conditions for temperature, we hold the temperature on the western wall constant, $T(0,t) = T_{\text{west}} = 0.1185\,\text{K}$, and the temperature on the eastern wall constant and equal to the saturation temperature, $T(L_x,t) = T_{\text{sat}} = 0\,\text{K}$[§].

Contrary to the other cases, we do not start the simulation at 0 s, but rather at $t = 0.25\,\text{s}$. The initial interface location is thus $\eta(0.25\,\text{s}) = 0.5\,\text{m}$. Both the temperature and the velocity profiles were initialized according to the analytical solution. The simulations were run from $t = 0.25\,\text{s}$ to $t = 0.6\,\text{s}$.

As in the vaporization case with uniform mass flux, the initial and boundary conditions made this an effective one-dimensional problem and the numerical results should be independent of y even though the simula-

[§]In spite of having performed most of the work on this thesis behind a computer rather than in the laboratory, the author is aware that a saturation temperature of 0 K is rare in real physical systems. This temperature was chosen to have results that would be comparable to those from Gibou et al. [2007]. Note also that the velocity and pressure fields do not depend on the zero-level of the temperature, so this could easily have been chosen differently.

tions were performed on a two-dimensional grid.

4.6.1 Analytical Solution

The analytical solution to this case was given, in part, by Gibou et al. [2007]. Here it is presented in some more detail.

As before, we give our solution different names in the two phases,

$$T(x,t) = \begin{cases} T^+(x,t) & \text{if } x \in \Omega^+ \\ T^-(x,t) & \text{if } x \in \Omega^- \end{cases}, \tag{4.74}$$

$$u(x,t) = \begin{cases} u^+(x,t) & \text{if } x \in \Omega^+ \\ u^-(x,t) & \text{if } x \in \Omega^- \end{cases}. \tag{4.75}$$

Due to the boundary and interface conditions, we can set the temperature in the entire liquid phase equal to T_{sat} at all times,

$$T^-(x,t) = T_{\text{sat}}, \tag{4.76}$$

and the velocity in the gaseous phase to zero,

$$u^+(x,t) = 0. \tag{4.77}$$

Thus the advection-diffusion equation for temperature (2.11) from page 8 in the gaseous phase reduces to

$$\partial_t T^+ = \alpha^+ \nabla^2 T^+. \tag{4.78}$$

It can be confirmed by straightforward differentiation and insertion that (4.78) has a general solution

$$T^+(x,t) = C_1 + C_2 \operatorname{erf}\left(\frac{x}{2\sqrt{\alpha^+ t}}\right), \tag{4.79}$$

where erf is the Gauss error function, defined in terms of an integral of the Gauss function,

$$\operatorname{erf}(\varphi) = \frac{2}{\sqrt{\pi}} \int_0^{\varphi} e^{-\tau^2} d\tau. \tag{4.80}$$

To determine the constants C_1 and C_2, we consider the boundary conditions. The boundary condition at the western wall, $T(0,t) = T_{\text{west}}$, gives

$$C_1 = T_{\text{west}}. \tag{4.81}$$

Next, we choose our C_2 in terms of a new constant C_3, related by

$$C_2 = \frac{T_{\text{sat}} - T_{\text{west}}}{\operatorname{erf}(C_3)}. \tag{4.82}$$

Then, the interface condition $T(\eta(t),t) = T_{\text{sat}}$ is satisfied when

$$\eta(t) = 2C_3\sqrt{\alpha^+ t}. \tag{4.83}$$

To determine C_3, we demand that the velocity of the interface $w = u^+ - m/\rho^+$ evaluated at the interface is equal to $\partial_t \eta(t)$. Using that $u^+ = 0$ and (2.42) on page 14, this results in the transcendental equation

$$C_3\ e^{(C_3)^2} \operatorname{erf}(C_3) = \frac{c_p^+ (T_{\text{sat}} - T_{\text{west}})}{\Delta h \sqrt{\pi}}. \tag{4.84}$$

This equation was numerically solved for C_3 using the bisection method [Süli and Mayers, 2006, Section 1.6]. To find $u^-(x,t)$ we solve the jump condition (2.38) from page 14, that is

$$u^+ - u^- = m\left[\frac{1}{\rho}\right], \tag{4.85}$$

for u^-. Again using that $u^+ = 0$ and (2.42), in addition to (4.83), we get

$$u^-(x,t) = \frac{\kappa^+}{\Delta h \sqrt{\pi \alpha^+ t}} \left[\frac{1}{\rho}\right] C_2\ e^{-(C_3)^2}. \tag{4.86}$$

4.6.2 Numerical Results

This case was run on five different grid sizes 10×10, 20×10, 40×10, 60×10 and 80×10. As Gibou et al. [2007], we use the GFM method for all fluid properties, except for the viscous term where we resort to CSF.

The obtained temperature profiles on the 40×10 gird is shown in Figure 4.13 at the initial time $t = 0.25\,\text{s}$, the final time $t = 0.6\,\text{s}$ and at $t = 0.425\,\text{s}$.

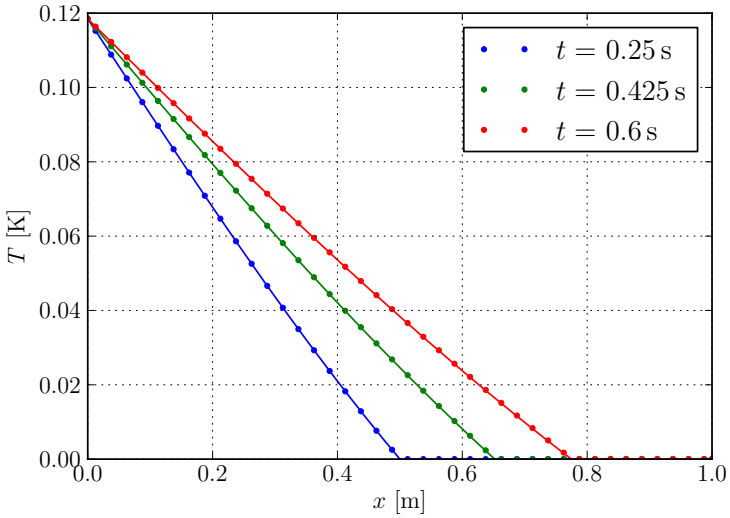


Figure 4.13: The temperature profiles obtained on the 40×10 grid at different times in the vaporization with physical mass flux case. The analytical solution at each time is drawn as a solid line.

At each time the analytical solution is also shown. From inspecting Figure 4.13, we see that the jump in $\partial_x T$ at the interface appears to decrease in time, so we expect the jump in velocity to decrease too, in accordance with (2.42) and (2.38) from page 14. We can also conclude that there is agreement between the simulation results and the analytical solution to plotting accuracy.

The velocity profiles obtained on the same 40×10 grid at the same points in time are shown in Figure 4.14. Again the analytical solution at each point in time is shown for comparison. From inspecting this figure, we see that the velocity jump at the interface does indeed decrease with time, as expected from the temperature profiles in Figure 4.13. Also, the jump in velocity appears a little overestimated with respect to the analytical solution, especially at $t = 0.25\,\mathrm{s}$. This is probably due to overestimation of $-\kappa^+ \partial_x T^+$ and may thus be attributed to the smearing of $\kappa\ \partial_x T$ around the interface that we commented on in Section 3.4.2. Overall, we conclude that there is good agreement between the simulated and analytical velocity profiles to plotting accuracy.

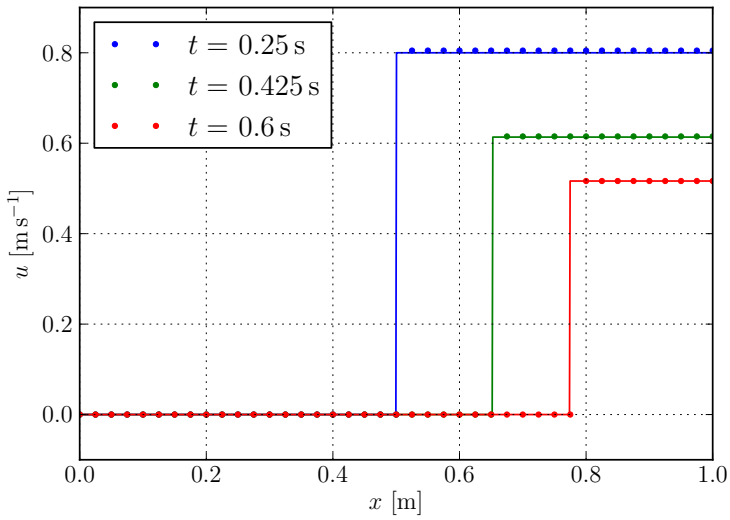


Figure 4.14: The velocity profiles obtained on the 40×10 grid at different times in the vaporization with physical mass flux case. The analytical solution at each time is drawn as a solid line.

The interface location η obtained from simulations on a selection of grid sizes is plotted against time t in Figure 4.15. The corresponding analytical solution is also shown for comparison. It is clear that $\eta(t)$ is somewhat overestimated, especially on the coarse 10×10 and 20×10 grids. This is consistent with overestimation of $-\kappa^+ \partial_x T^+$. In any event, it is evident that the simulation results approached the analytical solution upon grid refinement and that there is agreement, to plotting accuracy, between the simulated results on the 80×10 grid and the analytical solution.

To quantify the convergence order, a convergence study based on E_{abs} in $\eta(0.6\,\mathrm{s})$ with respect to the analytical solution $0.7746\,\mathrm{m}$ was performed. The results are given in Table 4.10 and in Figure 4.16. These are consistent with the first-order convergence rate we would expect from the first-order GFM treatment of the interface conditions.

From this case, we can conclude that there is good agreement between the numerical results and the analytical solution to the temperature and velocity profiles and the interface location. Also, we demonstrated the expected first-order convergence when refining the spatial grid. This indi-

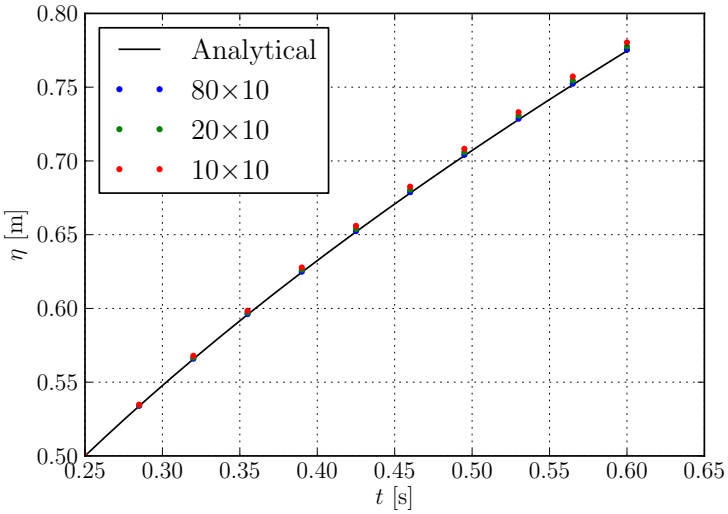


Figure 4.15: A plot of the interface location η against time t obtained on a selection of grid sizes in the vaporization with physical mass flux case. The analytical solution is shown as a solid line.

Table 4.10: Convergence table based on the error E_{abs}, with respect to the analytical solution, in the interface location η at $t = 0.6\,\text{s}$ in the vaporization case with physical mass flux.

N_x	$\eta(0.6\,\text{s})$	E_{abs}	n
-	m	m	-
10	$7.803 \cdot 10^{-01}$	$5.70 \cdot 10^{-03}$	-
20	$7.776 \cdot 10^{-01}$	$2.99 \cdot 10^{-03}$	0.93
40	$7.761 \cdot 10^{-01}$	$1.49 \cdot 10^{-03}$	1.01
60	$7.756 \cdot 10^{-01}$	$1.00 \cdot 10^{-03}$	0.97
80	$7.753 \cdot 10^{-01}$	$7.49 \cdot 10^{-04}$	1.02

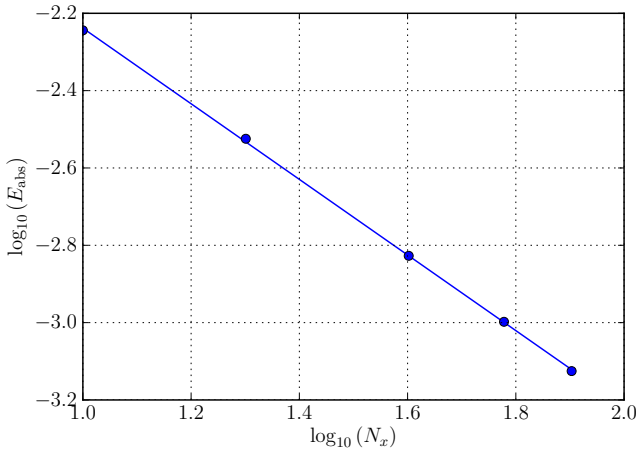


Figure 4.16: Log-log plot of the error E_{abs} in the interface location $\eta(0.6\,\text{s})$ in the vaporization case with physical mass flux. The solid line represents a first order polynomial fit and has a slope of approximately -0.98.

cates that the implementation of the phase transition model is correct in one dimension.

4.7 Vaporization of a Drop

To test the phase-transition model and its ability to preserve the discontinuities in $\vec{u}$ and p at the interface also in two dimensions, a case with a vaporizing drop was simulated. In contrast to the earlier cases with phase transition, we now have the possibility of a non-zero interface curvature and thus surface-tension forces come into play.

In this case we consider a two-dimensional fluid domain with $L_x = L_y = 0.08\,\text{m}$. Initially, we have a circular liquid drop (Phase 1) of radius $0.02\,\text{m}$ located in the center of the domain, surrounded by a gaseous phase (Phase 2). The fluids are initialized to be at rest and temperature on the entire domain is initially set equal to the saturation temperature $T_{\text{sat}} = 0\,\text{K}$.

The surface tension coefficient between the two phases is $\sigma = 0.1\,\text{N}\,\text{m}^{-1}$ and there is no gravity. The specific enthalpy difference associated with the phase change is $\Delta h = -1.0 \cdot 10^3\,\text{J}\,\text{kg}^{-1}$. Table 4.11 contains the rest of the relevant fluid parameters.

Table 4.11: Fluid parameters used in the vaporizing drop case

	Unit	Phase 1	Phase 2
ρ	$\mathrm{kg\,m^{-3}}$	200.0	5.0
μ	$\mathrm{Pa\,s}$	0.1	0.005
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	400.0	200.0
κ	$\mathrm{W\,K^{-1}\,m^{-1}}$	40.0	1.0

To have a supply of thermal energy to drive the vaporization, we set Dirichlet BCs, $T = 10\,\mathrm{K}$, for the temperature on all boundaries. We wish to allow the gas from the vaporizing drop to escape freely, so all boundaries carry outflow conditions for the velocity. Dirichlet boundary conditions were imposed for the pressure at all boundaries.

For this case, we expect that heat will flow from the boundaries into the fluid domain and increase its temperature. Eventually, the gas temperatures around the drop will be larger than the saturation temperature, vaporization will be initiated and the drop will gradually shrink. For symmetry reasons and due to surface tension, the drop should keep its circular shape and position in the center of the domain as it shrinks.

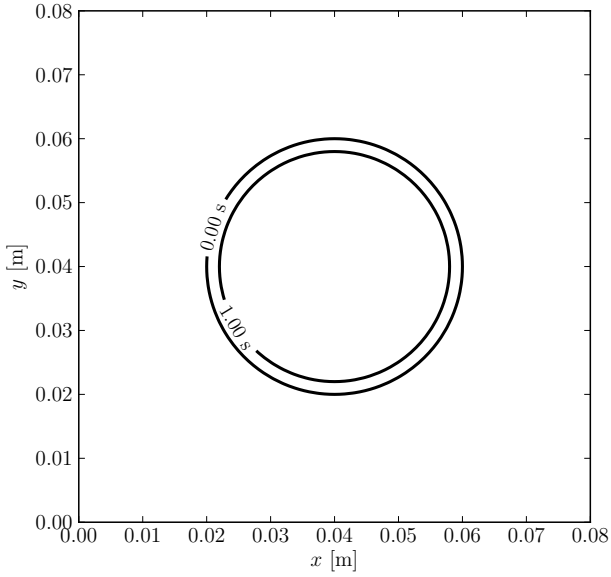
The simulation was run on a 101×101 grid to $t = 1.0\,\mathrm{s}$.

4.7.1 Numerical Results

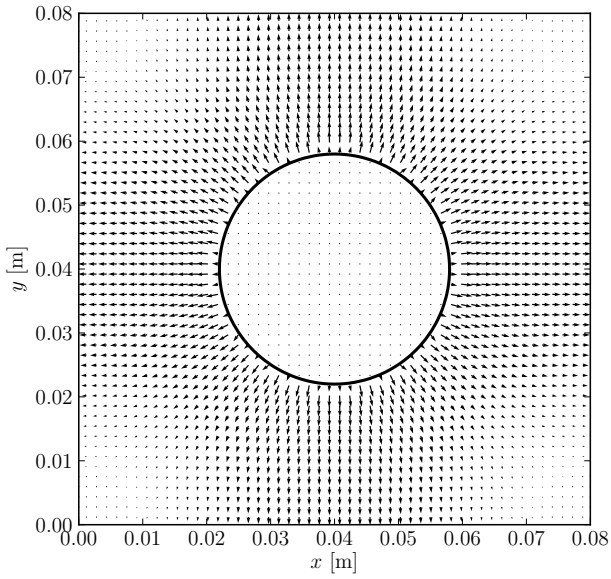
The interface location at $t = 0\,\mathrm{s}$ and at $t = 1.0\,\mathrm{s}$ is shown in Figure 4.17a. As expected, the drop has shrunk and it has kept its circular shape and its position in the center of the domain.

The velocity field at $t = 1.0\,\mathrm{s}$ is illustrated in Figure 4.17b. As expected, the velocities points normally away from the interface at the interface and they decrease gradually in magnitude as we move away the drop.

We observe also, and perhaps more unexpectedly, that the velocity field seems to bend towards the constant-x and the constant-y lines that go through the center of the drop. We might have expected a cylindrically symmetric velocity field with the magnitude of the velocity depending on the distance from the interface only. Still, the obtained velocity field is consistent with the Dirichlet BCs that were set of the pressure, which

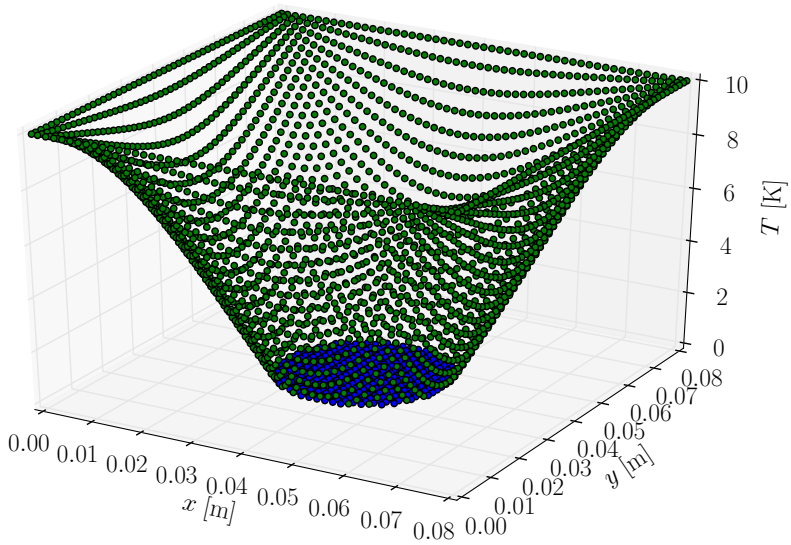


(a)

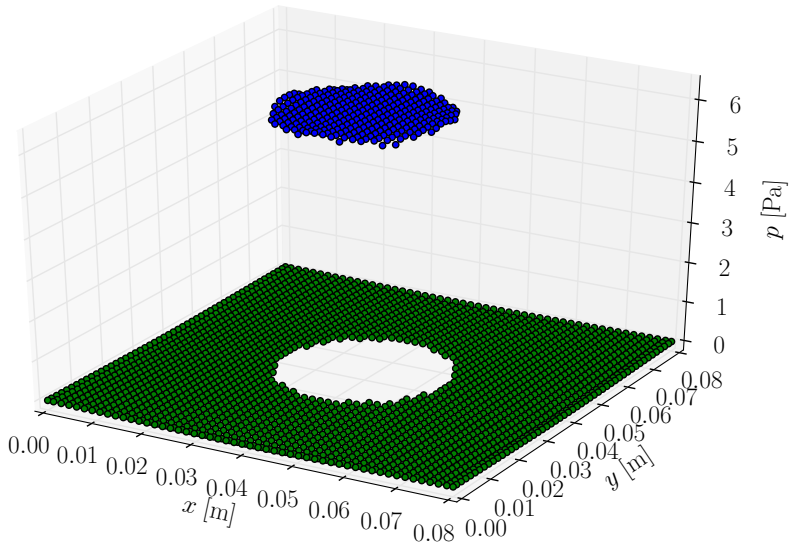


(b)

Figure 4.17: Plots of (a) the initial $t = 0\,\mathrm{s}$ and final $t = 1.0\,\mathrm{s}$ interface locations and (b) the velocity field at $t = 1.0\,\mathrm{s}$ in the vaporizing drop case.



(a)



(b)

Figure 4.18: Plots of (a) the temperature and (b) the pressure fields at $t = 1.0\,\mathrm{s}$ in the vaporizing drop case. The gaseous field values are indicated by green dots and the liquid values by blue dots.

do not have cylindrical symmetry. Because of the these BCs, $-\nabla p$ will tend to point along the direction normal to closest boundary rather than radially away from the drop and this will make the fluid tend to flow in this direction.

The temperature field at $t = 1.0\,\mathrm{s}$ is plotted in Figure 4.18a. This plot shows what we would qualitatively expect, a drop with uniform temperature T_{sat} and gas temperatures that decrease gradually when moving from the boundary to the interface.

Figure 4.18b shows the pressure at $t = 1.0\,\mathrm{s}$. As expected, it exhibits an interfacial jump that, for this case, can be mostly attributed to the interfacial tension. We observe that the interfacial pressure jumps at the points directly north, south, east and west of the drop center are somewhat higher compared to those located north-east, north-west, south-east and south-west. The reason for this is probably that the magnitude of temperature gradient in the gas phase is larger around these points resulting in a bigger mass flux. This is natural since the distance to the hot boundary is smaller for these points.

From this test case we can conclude that the our implementation of the phase-transition model is able to cope with interfacial jumps also in two dimensions. We have also seen that we must take care when imposing Dirichlet BCs on the pressure as they may not always be consistent with the physical situation we want to model.

4.8 Boiling Film

Finally, we turn our attention to a case with thin film boiling in two dimensions. In this situation, a thin layer of vapor (Phase 2) covers the south boundary and the rest of the fluid domain is occupied by a liquid (Phase 1). The south wall is heated to a temperature above the saturation temperature of the liquid and this causes the liquid to evaporate at the interface and increase the volume of the film. At the appropriate wall heating, rising vapor bubbles should form due to gravity and the density differences between the liquid and the vapor.

We consider a fluid domain with equal width and height, $L_x = L_y = 0.08\,\mathrm{m}$. The fluids are initially at rest and have a uniform temperature of

Table 4.12: Fluid parameters used in the boiling film case.

	Unit	Phase 1	Phase 2
ρ	$\mathrm{kg\,m^{-3}}$	200.0	5.0
μ	$\mathrm{Pa\,s}$	0.1	0.005
c_p	$\mathrm{J\,kg^{-1}\,K^{-1}}$	400.0	200.0
κ	$\mathrm{W\,K^{-1}\,m^{-1}}$	40.0	1.0

0 K. The interface location is initialized according to the parametrization

$$y(x) = \frac{L_x}{128}\left\{4 + \cos\left(\frac{2\pi(x - L_x/2)}{L_x}\right)\right\}. \tag{4.87}$$

The specific enthalpy difference associated with the phase change in this case is $\Delta h = -1.0 \cdot 10^3\,\mathrm{J\,kg^{-1}}$, the gravitational acceleration is $\vec{g} = \left(-9.81\,\mathrm{m\,s^{-1}}\right)\hat{y}$ and the coefficient of surface tension is $\sigma = 0.1\,\mathrm{N\,m^{-1}}$. The rest of the relevant fluid parameters are given in Table 4.12.

We pose no-slip BCs on the south wall, free-slip on the east and west walls and outflow BCs on the north wall. The north wall has a constant temperature equal to the saturation temperature of the liquid, $T_{\mathrm{sat}} = 0\,\mathrm{K}$, and the south wall a constant temperature of 0.1 K. The east and west walls are thermally insulating. We pose Neumann BCs on the pressure on all walls except the north where we use the Dirichlet BC.

The simulation was run on a 71×71 grid.

4.8.1 Numerical Results

The temporal development of the interface is shown in Figure 4.19. We see that the film increases in volume from $t = 0.3\,\mathrm{s}$ to $t = 0.4\,\mathrm{s}$ as the liquid at the interface evaporates. The vapor then gathers in the middle of the domain, forming the beginnings of a bubble. At $t = 0.5\,\mathrm{s}$ the bubble has begun to rise. In the final frame, at $t = 0.6\,\mathrm{s}$, the bubble has obtained a mushroom shape as its side lobes are pulled down by the vortices on either side of the stem. These vortices are illustrated in Figure 4.20.

According to Gibou et al. [2007], the qualitatively correct behavior for film boiling in two dimensions is, perhaps somewhat counter-intuitively,

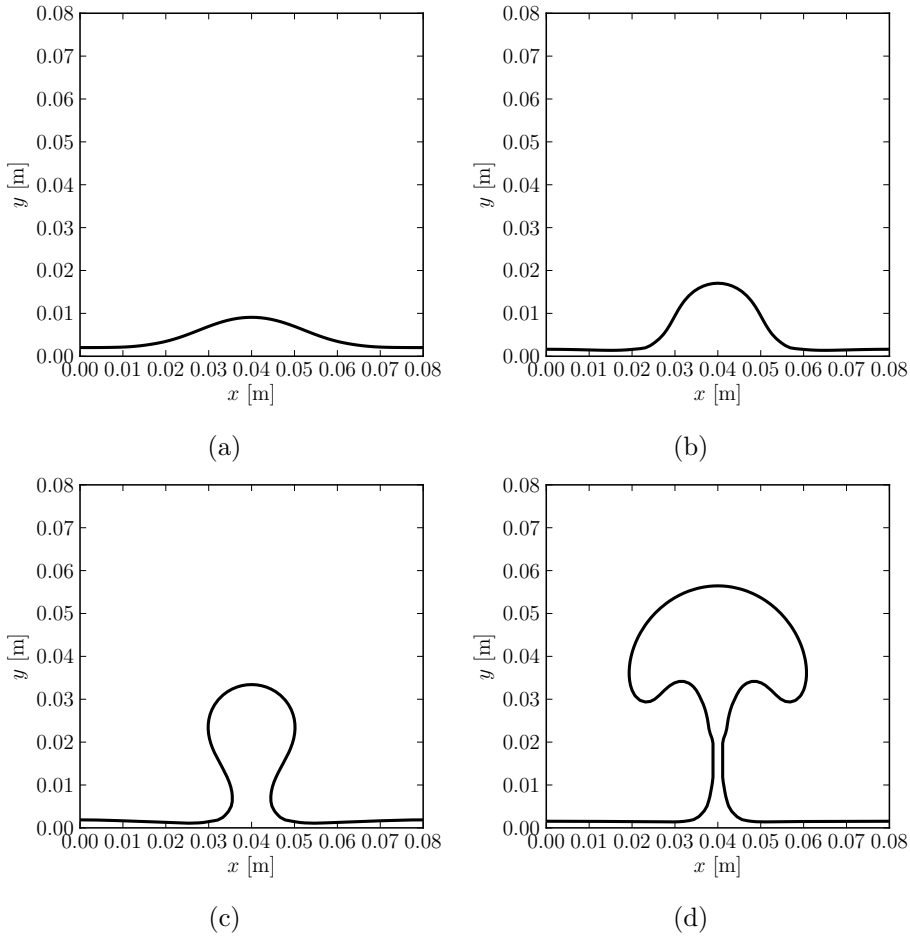


Figure 4.19: The interface location in the boiling film case at (a) $t = 0.3\,\mathrm{s}$, (b) $t = 0.4\,\mathrm{s}$, (c) $t = 0.5\,\mathrm{s}$ and (d) $t = 0.6\,\mathrm{s}$.

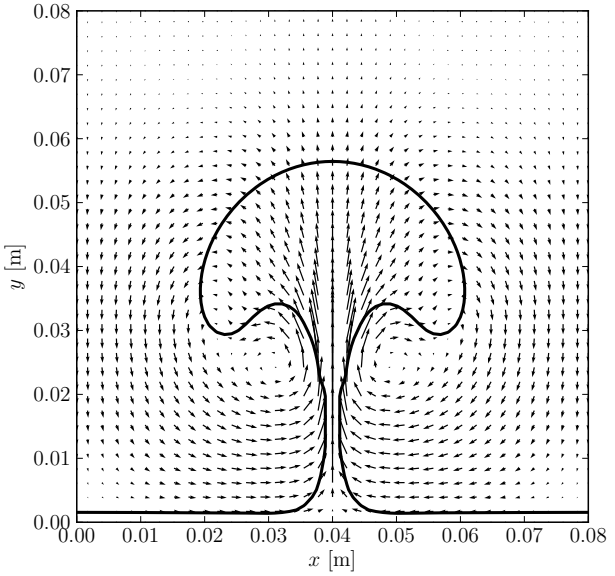


Figure 4.20: Illustration of the velocity field at $t = 0.6\,\mathrm{s}$ in the the boiling film case.. Vortices can be observed on either side of the stem.

that the stem of the rising bubble should not pinch off. In two dimensions, the curvature of the stem is negligible and therefore so are the surface tension effects and thus the stem persists and continues to fill the rising bubble with vapor. Gibou et al. [2007] observed this behavior in their boiling film simulations on fine grids. It was only on coarser grids, that could not resolve the stem of the rising bubbles, that pinch-off was observed. This explains why bubble pinch-off was not observed in the simulations run here.

4.9 Summary

In this chapter, we have presented eight test cases and discussed results from their simulations. Here, we briefly summarize the conclusions from these discussions.

De Vahl Davis' Benchmark Case

From the results of the simulations of the de Vahl Davis benchmark case we concluded that the implementation was able to reproduce the benchmark results on the 301×301 grid. Also, a convergence study indicated that the order of convergence was consistent with the expected value of 2. This gave reason to believe that the current implementation of the discretization of the advection-diffusion equation for temperature and the Boussinesq coupling to the Navier–Stokes equations for single-phase flow cases was correct.

Two-Phase Heat Conduction

In this case we saw that the CSF discretization of the diffusive term in the advection-diffusion equation for temperature failed to reproduce the analytical solution both with Dirichlet and Neumann boundary conditions. The GFM discretization, however, produced solutions in agreement with the analytical for both Dirichlet and Neumann boundary conditions. This test case thus showed the advantage of treating the interface sharply with the GFM instead of with the CSF method when solving the heat equation.

Wang et al.'s Two-Layer Convection Case

From this test case we concluded that our implementation produced results in agreement with the analytical steady-state solution for a case involving temperature-driven two-phase flow. Also, the performed convergence study showed the expected first-order convergence. This gave reason to believe that the current implementation of the discretization of the advection-diffusion equation and the Boussinesq coupling to the Navier–Stokes equations was correct also for two-phase flow cases where the interface is treated with the GFM.

Lava Lamp

Here we saw that our two-phase model reproduced the experimentally observed blob exchange oscillations and their time periods. This gave reason to believe that the implementation was correct also on axisymmetric grids. Still, the simulated behavior could not be said to be in full agreement with

the experimental results and neither could we expect it to be, since many of the physical aspects of lava lamp convection are not accounted for in our model.

Vaporization with Uniform Mass Flux

From this test case we could conclude that the current implementation succeeded in preserving the discontinuities in the velocity and pressure fields, even when grid nodes changed character from liquid to gas, in one dimension and in the case of uniform mass flux density. It also succeeded in propagating an instantaneous change in the velocity field at the interface to the entire domain instantaneously as is necessary in the incompressible model.

Vaporization with Physical Mass Flux

Here we observed that there was good agreement between the numerical results and the analytical solution to the temperature and velocity profiles and the interface location. Also, we demonstrated the expected first-order convergence when refining the spatial grid.

Vaporization of a Drop

From this test case we could conclude that the our implementation of the phase-transition model was able to cope with interfacial jumps also in two dimensions.

Boiling Film

In the boiling film test case we saw a boiling film that produced a rising bubble. The results were in qualitative agreement with those from Gibou et al. [2007].

5 Conclusions and Suggestions for Further Work

5.1 Conclusions

In this thesis, we have presented a two-phase flow model, based on the incompressible Navier–Stokes equations. We have also presented the numerical methods used in solving this model, with emphasis on non-standard methods such as the level-set method for tracking the interface, the ghost-fluid method for treating the interface sharply and the weighted essentially non-oscillatory scheme for discretizing advective terms in the governing equations.

The main contributions of this work have been to extend the implementation of the flow model through the discretization and implementation of the advection-diffusion equation for temperature and a Boussinesq coupling between the temperature and velocity fields. In two-phase flow, both the continuum surface force method and the ghost-fluid method were employed for handling jumps at the interface. Results from simulated cases indicated that implementation for both single- and two-phase flow with ghost-fluid method was correct, with second- and first-order convergence, respectively. Also, a model for phase transition has been presented, discussed and implemented. This implementation allowed for vaporization and condensation mass transport between the phases. Results from simulated one-dimensional cases with phase transition indicated that the implementation of this model was correct, with first-order convergence. The results from one dimension and the qualitatively correct results from two dimensions gave reason to believe that the implementation was correct also in two dimensions.

Through the introduction of heat-transport physics in the implementation of the two-phase flow model, it has been developed in direction of performing more detailed simulations that are relevant for two-phase heat

exchangers and hence for the natural gas liquefaction processes.

5.2 Suggestions for Further Work

As the results from the two-phase heat conduction case suggested, the continuum surface force (CSF) method did not perform very well at discretizing the diffusion term in the advection-diffusion equation for temperature. Still, we rely on this method when treating the viscous term in the Navier–Stokes equations when simulating cases with phase transition. Therefore, an effort could be made to assess the effect of the CSF treatment of viscosity, or if one can implement a ghost-fluid-method (GFM) treatment, for this type of simulations.

The chosen discretization of the diffusive term in the advection-diffusion equation with phase transition led to a strict time-step restriction. An implicit treatment of the advection-diffusion equation could be attempted to avoid this restriction and run stable simulations with a smaller cell division threshold θ_c and thus longer time steps. Another benefit of reducing θ_c is that we might get better-quality approximations to the temperature gradient at the interface.

One could also look at a way of posing boundary conditions on the level-set function, e.g. using equation (61.12) from Landau and Lifshitz [1987], to get a realistic contact angle between the interface and the domain walls. This might result in more accurate simulations in situations like the lava lamp case.

In this thesis, we have put more emphasis on verifying the implementation of the model (testing if the numerical method and its implementation solves the model equations) and less on validating the correctness of the model with respect to the real world (testing if the model accurately predicts experimental results). To this end, more detailed comparisons of simulated results with experiments should be performed.

Bibliography

T. D. Aslam. A partial differential equation approach to multidimensional extrapolation. *Journal of Computational Physics*, 193:349–355, 2003.

S. Balay, J. Brown, K. Buschelman, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith, and H. Zhang. PETSc web page, June 2013. URL `http://www.mcs.anl.gov/petsc`.

G. de Vahl Davis. Natural convection of air in a square cavity: A bench mark numerical simulation. *International Journal for Numerical Methods in Fluids*, 3(3):249–264, 1983.

V. K. Dhir. Numerical simulations of pool-boiling heat transfer. *AIChE Journal*, 47(4):813–834, 2001.

F. Gibou, R. Fedkiw, L. Cheng, and M. Kang. A second-order-accurate symmetric discretization of the Poisson equation on irregular domains. *Journal of Computational Physics*, 176:205–227, 2002.

F. Gibou, L. Chen, D. Nguyen, and S. Banerjee. A level set based sharp interface method for the multiphase incompressible Navier–Stokes equations with phase change. *Journal of Computational Physics*, 222:536–555, 2007.

M. Gisvold. Eventyret på Melkøya. Gemini Forskingsnytt fra NTNU og SINTEF, June 2004.

M. Griebel, T. Dornseifer, and T. Neunhoeffer. *Numerical Simulation in Fluid Dynamics*. SIAM, 1998.

B. Gyüre and I. M. Jánosi. Basics of lava lamp convection. *Physical Review E*, 80(4):046307, 2009.

E. B. Hansen. *Numerical Simulation of Droplet Dynamics in the Presence of an Electric Field*. PhD thesis, Norwegian University of Science and Technology, November 2005.

G. S. Jiang and C. W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126(1):202–228, 1996.

M. Kang, R. Fedkiw, and X. Liu. A boundary condition capturing method for multiphase incompressible flow. *Journal of Scientific Computing*, 15: 323–360, 2000.

D. I. Ketcheson and A. C. Robinson. On the practical importance of the SSP property for Runge–Kutta time integrators for some common Godunov-type schemes. *International Journal for Numerical Methods in Fluids*, 48(3):271–303, 2005.

E. Kreyszig. *Advanced Engineering Mathematics*. Wiley, 9th edition, 2006.

T. Kunugi. Brief review of latest direct numerical simulation on pool and film boiling. *Nuclear Engineering and Technology*, 44(8):847–854, 2012.

L. D. Landau and E. M. Lifshitz. *Fluid Mechanics*, volume 6 of *A Course on Theoretical Physics*. Elsevier, 2nd edition, 1987.

K. Y. Lervåg. Simulation of two-phase flow with varying surface tension. Master's thesis, The Norwegian University of Science and Technology, June 2008.

S. W. Løvseth. Enabling low-emission LNG systems web page, June 2013. URL `http://www.sintef.no/Projectweb/Low-Emission-LNG-Systems`.

Norwegian Petroleum Directorate. Factpages, May 2013. URL `http://factpages.ndp.no`.

S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2003.

S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

F. S. Oueslati, R. Bennacer, H. Sammouda, and M. El Ganaoui. Analytical and numerical solutions for natural convection in a shallow cavity filled with two immiscible fluids: Shear stress action. *Numerical Heat Transfer; Part A: Applications*, 62(8):605–623, 2012.

K. Rottmann. *Matematisk Formelsamling*. Spektrum forlag, 2003.

R. Scardovelli and S. Zaleski. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, 31:567–603, 1999.

M. Sussmann, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:149–159, 1994.

E. Süli and D. Mayers. *An Introduction to Numerical Analysis*. Cambridge University Press, 2006.

I. Tamura, T. Tanaka, T. Kagajo, S. Kuwabara, T. Yoshioka, T. Nagata, K. Kurahashi, and H. Ishitani. Life cycle CO_2 analysis of LNG and city gas. *Applied Energy*, 68(3):301–319, 2001.

S. Tanguy, T. Ménard, and A. Berlemont. A level set method for vaporizing two-phase flows. *Journal of Computational Physics*, 221(2):837–853, 2007.

G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, and Y.-J. Jan. A front-tracking method for the computations of multiphase flow. *Journal of Computational Physics*, 169(2):708–759, 2001.

C. H. Wang, M. Sen, and P. Vasseur. Analytical investigation of Bénard–Marangoni convection heat transfer in a shallow cavity filled with two immiscible fluids. *Applied Scientific Research*, 48(1):35–53, 1991.

F. M. White. *Fluid Mechanics*. McGraw–Hill, sixth edition, 2008.

H. K. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *Journal of Computational Physics*, 127 (1):179–195, 1996.