Efficient numerical simulation of the liquid phase epitaxy

Dissertation zur Erlangung des Doktorgrades der
Mathematisch-Naturwissenschaflichen Fakultät der
Universität Augsburg

vorgelegt von Oliver Kriessl

Dezember 2005

Erster Gutachter: Prof. Dr. K.G. Siebert, Augsburg, Deutschland

Zweiter Gutachter: Prof. Dr. P. Morin, Santa Fe, Argentinien

Mündliche Prüfung: 01. Februar 2006

Danksagung

An dieser Stelle möchte ich mich bei allen bedanken, die zur Fertigstellung dieser Arbeit beigetragen haben. Es war ein langer Weg mit manchem Stolperstein bis zur Abgabe. Am Anfang stand einfach der Wunsch, es nochmals mit der Mathematik an der Uni Freiburg aufzunehmen. Ermöglicht wurde dies durch ein Projekt des Bundesministerium für Bildung und Forschung (BMBF), dem ich hiermit meinen herzlichen Dank für drei Jahre Förderung aussprechen möchte. Begleitet wurde ich von vielen netten Kollegen. Erwähnen möchte ich vor allem Oliver Müller, der als Zimmernachbar viel Anteil an alltäglichen Freuden und Frust hatte.

Dann kam der Wechsel an die Uni Augsburg. Zwei Personen haben ihn mir erheblich erleichtert: Daniel Köster, der mir dauerhaft bis heute und immer ohne jegliche Probleme Unterkunft gewährte, und Kunibert G. Siebert, der mich motivierte, unterstützte und auch in schwierigen Situationen immer wieder auf den richtigen Weg zurück brachte. Ohne diese beiden wäre die Arbeit nie möglich geworden. Danken möchte ich auch dem Augsburger Graduiertenkolleg, das mich am Ende seiner Laufzeit herzlich aufgenommen hat und finanzielle Sorgen im Hintergrund verschwinden ließ. Pedro Morin möchte ich für einen angenehmen Argentinienaufenthalt danken, dem manche Grundlage dieser Arbeit entsprungen ist.

In der heißen Phase des Aufschreibens bin ich meinen beiden Bürogenossen Christian Kreuzer und Daniel Köster zu großem Dank verpflichtet, die ausdauernd als Korrektoren hergehalten haben, wenn auch nicht immer in ganz nüchternem Zustand ... und Kunibert G. Siebert, der mich von allem frei und fern gehalten hat, was in irgendeiner Form die mir noch zur Verfügung stehende Zeit geschmälert hätte.

Der größte Rückhalt in der Zeit der Promotion aber war meine liebe Frau Tanja, die mich animiert hat, den Schritt nach Augsburg zu tun und alle guten und schlimmen Phasen tapfer mitgetragen hat. Besonders danke ich ihr für das schönste Geschenk, das sie mir überhaupt machen konnte — kurz vor der Abgabe!

Contents

Chapter 1

Introduction

In many industrial processes optimization plays a fundamental role since several important things may be improved such as energy consumption, material requirements, or quality. Most of these aspects can be enhanced by experiments but on the one hand this is time-consuming and on the other hand it is sometimes extremely costly. One way out may be performing numerical simulations of the process. This can lead to a deeper understanding and thus facilitate improvements.

The liquid phase epitaxial process which is the subject of this thesis is an important and critical step in the fabrication process of infrared detectors. A high quality of the emerging layer which has to be grown onto the substrates is decisive for their further usability. Performing numerical simulations of the liquid phase epitaxy requires the development of a physical model as well as specialized numerical methods. These steps are presented and the obtained numerical results are discussed.

The construction of an optimal control maintaining the process to receive an optimal result would even be more sophisticated. For the liquid phase epitaxy this is impossible until now since the complexity of the arising problem is so high that simulations of the process are already extremely time-consuming.

1.1 Infrared detectors

For the industrial production of infrared detectors high quality semiconductors are needed. The detectors we consider are optimized for wavelengths between 7 and 12 microns on the electromagnetic spectrum. In this region room temperature objects emit most of their light. The detectors use a semiconductor material based on mercury cadmium telluride (MCT) as most of the infrared detectors do. The material is very expensive and difficult to handle and a special manufacturing infrastructure is required that raises the cost of the chips. The fabrication process is split into several steps. It starts with the breeding of a single crystalline ingot (cf. Figure 1.2) by the vertical Bridgman method (see Section 1.1.1). This ingot is then cut into slices which are, after some further treatments like etching and polishing, the substrates (cf. Figure 1.3) for the epitaxial growth (see Section 1.1.2). At the end electronic devices are attached onto the layered substrates to complete the chip.

Figure 1.1: Infrared picture of a house. (Source: Internet)

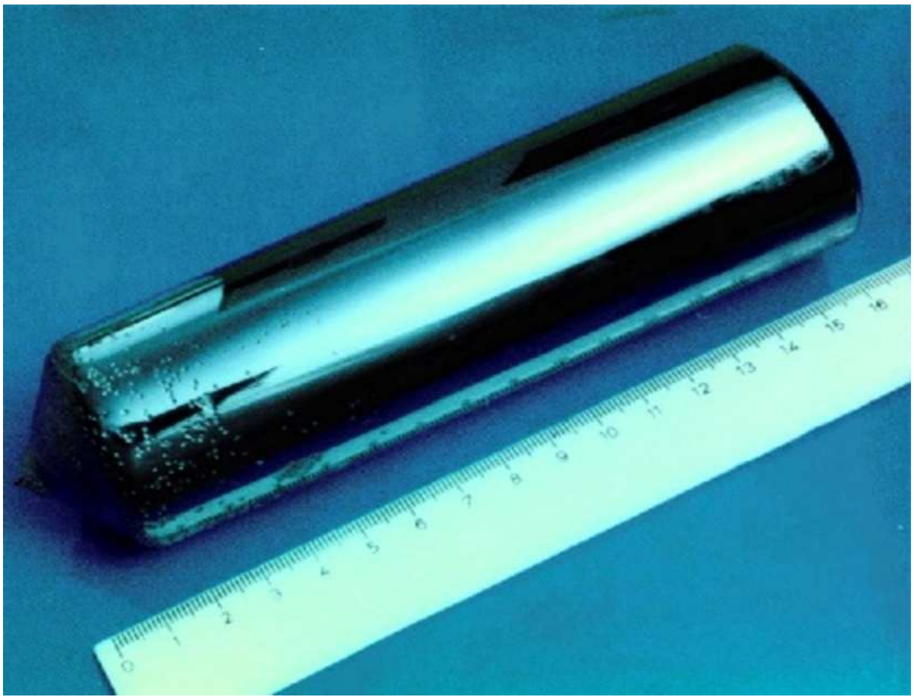


Figure 1.2: (Cd,Zn)Te crystal. (Source: AIM)

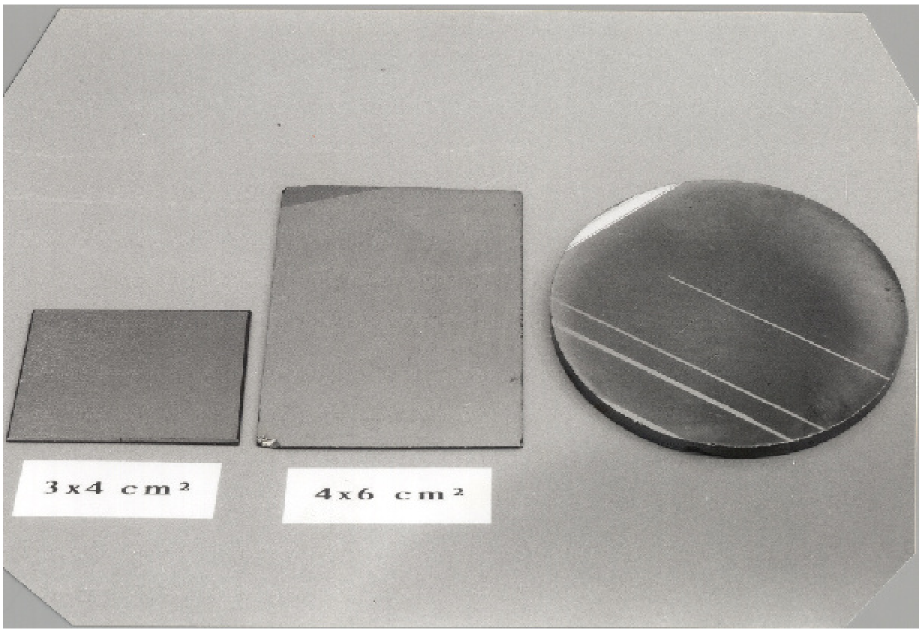


Figure 1.3: Slices of the substrate material. (Source: AIM)

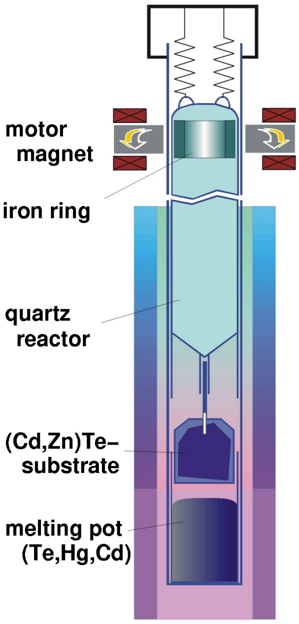
1.1.1 The vertical Bridgman method

For the breeding of the crystal the required components are molten inside a quartz ampoule in a furnace which is hotter on top and colder on bottom. Through lowering of the ampoule inside the furnace crystallization starts. A readily grown ingot of a (Cd,Zn)Te crystal as well as already etched and polished slices of the substrate material are shown in Figure 1.2 and 1.3.

The Kristallographisches Institut of the University of Freiburg, the Institut für Angewandte Mathematik of the University of Freiburg, and the industry partner AIM Infrarot-Module GmbH in Heilbronn analyzed the production of the (Cd,Zn)Te crystal in an interdisciplinary project. The obtained results led to a significant improvement of the process (cf. [6, 7, 8, 39]).

1.1.2 Liquid phase epitaxy

In the next step of the fabrication process a thin film, the epitaxial layer, with a thickness of about twenty microns has to be grown onto the substrate. There are different techniques available for this purpose. Here we deal with liquid phase epitaxy: In a complicated furnace (its sketch is shown on the left side) the compound materials are molten up. Then the substrate is dipped into the melt. By cooling down with an adequate rate a thin single crystalline film begins to grow with the surface of the substrate as nucleation area. The temperature during this process is around 465 degree Celsius. After about 40 minutes the substrate is pulled out of the melt.



This old fashioned technique has some important advantages: a high throughput is possible, the melt can be used for several processes, the shape of a substrate is not relevant, and multiple substrates can be treated simultaneously. Nevertheless the process is quite costly since the required components are expensive and must be very pure. So it is very important to improve the

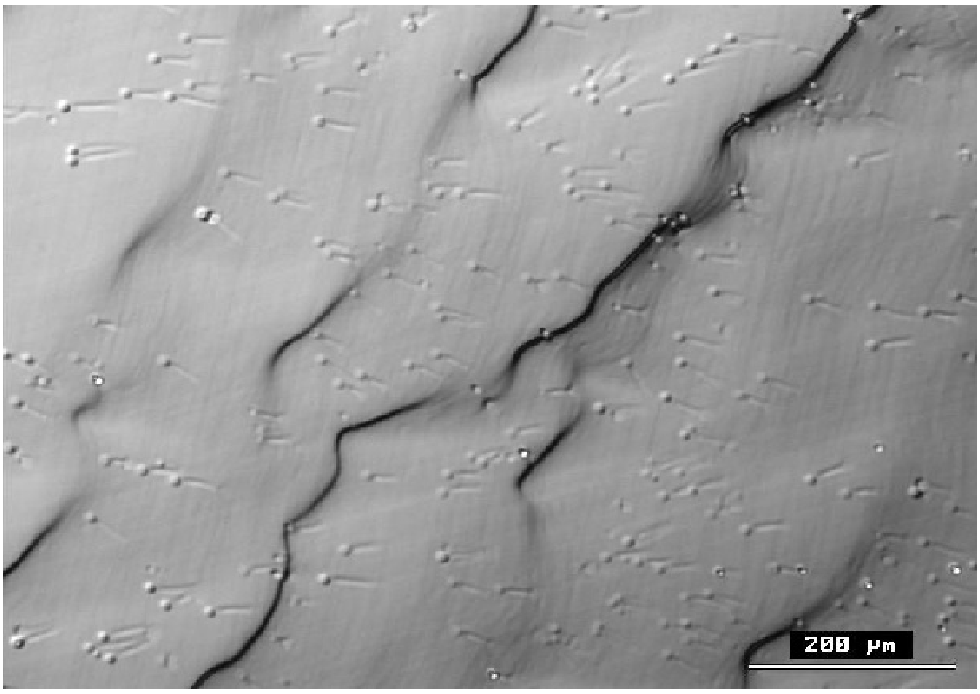


Figure 1.4: Epitaxial layer. (Source: AIM)

process rather by performing numerical simulations than by making experiments where most of the resulting semiconductors have to be thrown away because they do not comply with required quality standards.
The simulation of the liquid phase epitaxy was split into two separated parts. The first part deals only with the temperature field in the furnace (cf. next section). The second part, which is the subject of this thesis, deals with the processes inside the melting pot with given temperature data as boundary condition.

1.2 Global simulation

As already indicated in the last section the simulation of the liquid phase epitaxy has been split into two separated parts. The first part just deals with the temperature field in the furnace without crystal growth inside the melting pot. We call this part the global simulation. Since our part requires temperature data

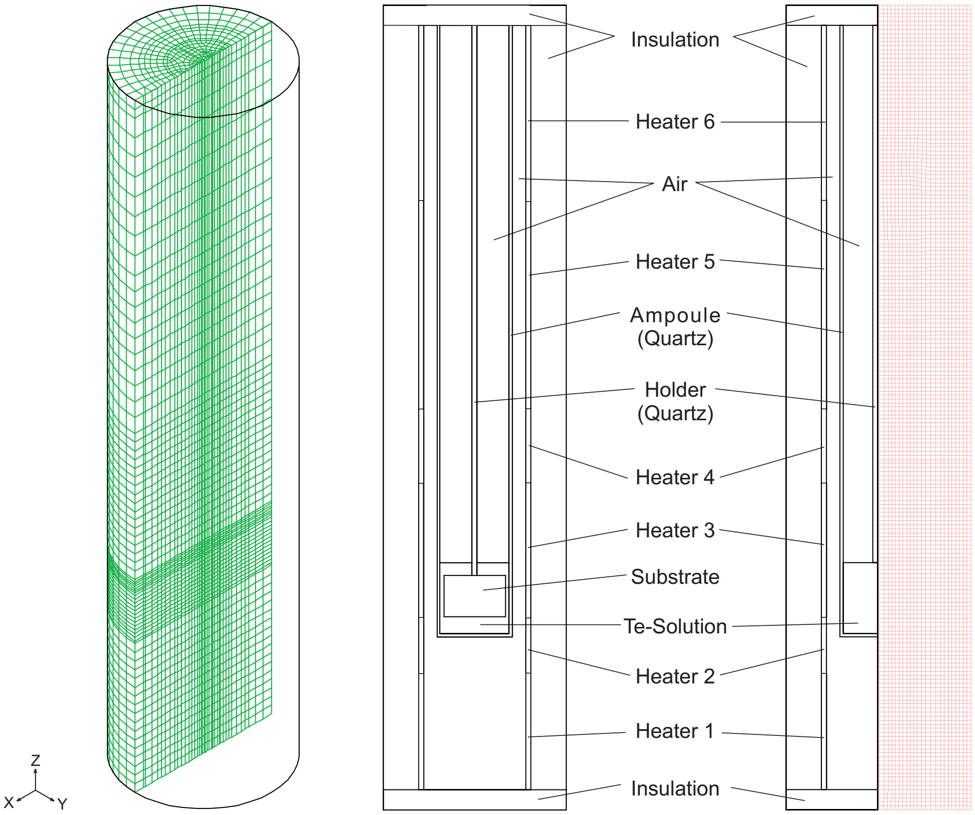


Figure 1.5: Grid for furnace simulations. (Source: K.-M. Lin)

of the global simulation as boundary condition for the local simulation inside the melting pot where the epitaxial growth occurs, we shortly present the main results of the investigation here. The global simulations were performed by Lin at the Kristallographisches Institut of the University of Freiburg.

For getting a cooling rate in the furnace the heater powers were continuously adapted, controlled by a connected computer. The idea of Lin was to perform several stationary simulations of the complete furnace for different heater powers, assuming thermal equilibrium for each state. Figure 1.5 shows the calculation grid for 2d and 3d simulations. By interpolation in time an instationary temperature field in the furnace is given.

The conclusion of Lin's results is that instationary effects can not be neglected

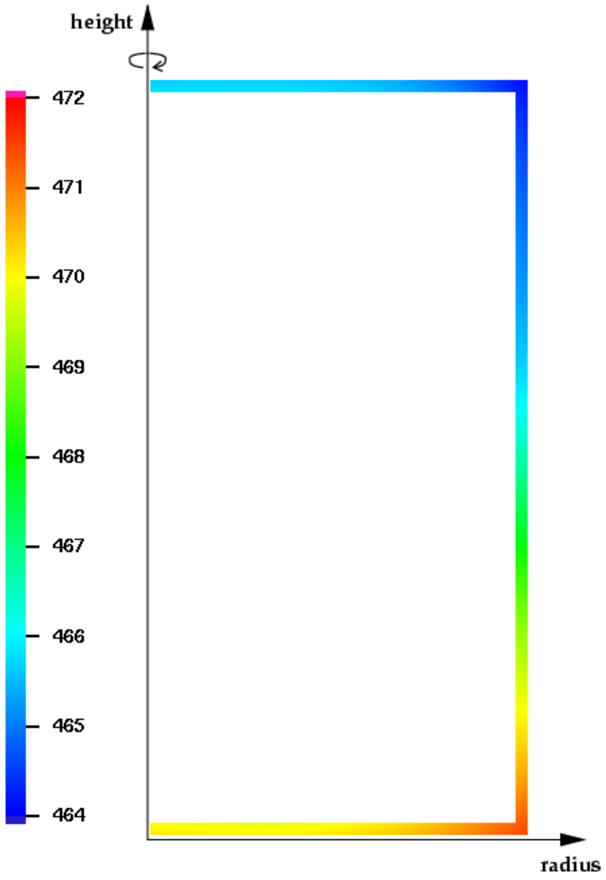


Figure 1.6: Rotationally symmetric spatial temperature field in ℃ on the boundary of the pot after 18 minutes in the process flow.

(see [19] for the complete results). Thus, an instationary simulation of the furnace would be necessary. He could overcome this problem by calibrating the simulation data with experimental data from AIM for their standard process flow. So Lin could provide us with a continuous time-dependent temperature field of the furnace with good compliance to measuring data from AIM. For the melting pot we obtained a rotationally symmetric spatial temperature field on its boundary (cf. Figure 1.6) that cools down with a constant cooling rate of approximately -0.15 degree Celsius per minute. Ongoing research about the topic of instationary simulations is done for instance at the Center of Advanced European Studies and Research (Caesar) in Bonn (cf. [54] or visit www.caesar.de).

1.3 Outline

In the next chapter we derive a physical model for the liquid phase epitaxy. The focus of the modeling lies on the calculation of the thickness of the grown layer and not on microscopic aspects. The model includes the arising convection in the melt and retains energy and mass conservation. The material constants and the complete system with initial values and boundary conditions in non-dimensional form is presented.

In the third chapter we discuss some mathematical aspects of a linearized subproblem. This already exhibits the difficulties occuring in the discretization of the problem. Two mathematical models are presented. The first approach takes all aspects of the physical model into account and therefore permits mass conservation also on the discrete level. The second one omits one fact – the geometrical movement of the phase boundary between liquid and solid part.

In the fourth chapter the basic finite element spaces are presented. Then the discretization of the Navier-Stokes equations is treated. Afterwards the fully discretized problems of both mathematical models are derived. Some aspects regarding stability and damping of numerical oscillations are considered.

The fifth chapter addresses the numerical solution of the resulting algebraic system by a Newton method with step size control. Furthermore we derive the linear system emerging in each Newton step and discuss appropriate iterative solvers. Since the linear system is quite large preconditioning is an important aspect and is addressed afterwards.

In the last chapter the discretization is tested. After a comparison of the two mathematical models the numerical results for different settings are presented. Finally we discuss them with respect to the experimental results and the modeling of the process.

Chapter 2

The Physical Model

The liquid phase epitaxy is a part of the production process of semiconductors: Inside a heated furnace a substrate consisting of cadmium, zinc, and tellurium (cf. Figure 1.3 on page 3) is dipped into the melting pot which is filled with a compound of the molten materials cadmium, mercury, and tellurium. By reducing the temperature of the heaters inside the furnace a thin single crystalline film begins to grow onto the surface of the substrate to form the epitaxial layer. The temperature data outside the pot is available from measurements and as simulation data (cf. Section 1.2) and acts as boundary condition for the inside temperature.

In this chapter we deal with the derivation of a physical model for the process taking place inside the melting pot. For this purpose we take a closer look at the conservation equations of thermodynamics valid inside our area of interest. This model was developed in collaboration with the Kristallographisches Institut of the University of Freiburg, based on a model that was already used and tested for crystal growth by the working group of Kimura (cf. [33, 34, 32, 46, 30]). It ought to allow an easy calculation of the thickness of the newly grown layer which is our main concern. Therefore it ignores microscopic effects like step bunching or micro drops as can be seen in Figure 1.4 (on page 5).

2.1 Prerequisites

The melting pot, denoted by Ω, consists of two essentially time-dependent parts: the substrate Ω^s and the melt Ω^ℓ. The illustration of the geometry is shown in Figure 2.1.

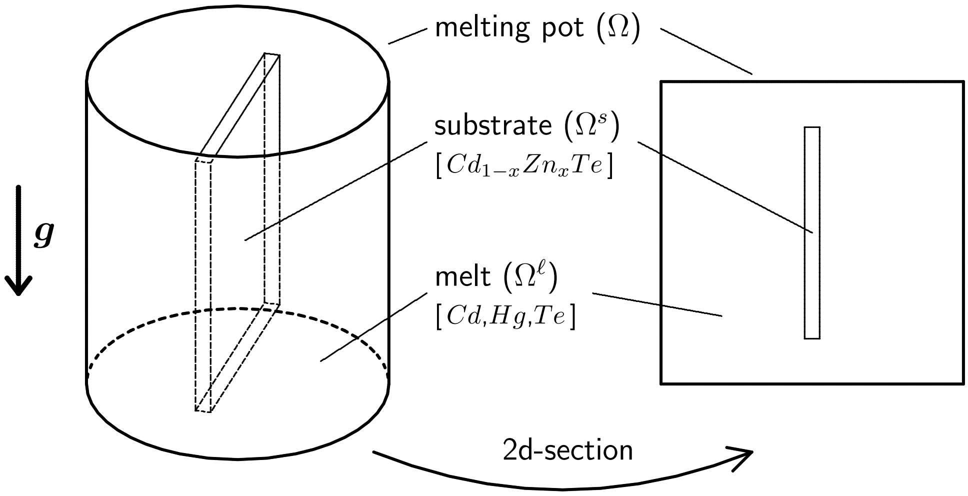


Figure 2.1: Scheme of the melting pot. On the left side the direction of gravity $\boldsymbol{g}$ is shown.

For the description of the flow inside the melt we need a parameterization of the flow over a fixed reference domain, namely the part filled with the melt before any crystal growth occurs, which we denote by Ω_0^ℓ.

Assumption 2.1. *We assume that for every $t \in [0,T]$ there exists a diffeomorphism*

$$\Phi(t,\,\cdot\,) : \overline{\Omega}_0^\ell \to \overline{\Omega}^\ell(t)$$

with $\Phi(0,\,\cdot\,) = \mathrm{id}$ and smooth in time, such that $x(t) = \Phi(t,\overline{x})$ is the trajectory of a particle starting at $\overline{x}$.

The velocity of a particle with reference $\overline{x}$ is therefore given by the time derivative of the diffeomorphism Φ:

$$\boldsymbol{u}(t,x) := \dot{x}(t) = \Phi_t(t,\overline{x}).$$

The velocity $\boldsymbol{u}$ can be understood as a mean velocity of all particles at the specified place (t,x) and thus implies a homogeneous distribution of matter, i.e. it is not the exact velocity of a single particle.

For the derivation Reynolds' transport theorem is crucial.

Theorem 2.2 (Reynolds).
Let Φ be defined as in assumption 2.1. Additionally let the Jacobian of Φ be regular, i.e. $\det(D_x\Phi(t,x)) > 0$. Let $V_0 \subset \Omega_0^\ell$ be an open subset with smooth boundary and $V(t) := \Phi(V_0, t)$. Furthermore let $\boldsymbol{\nu}(t,x)$ be the outer normal of $\partial V(t)$. Then for a function $f \in C^1(I\!R^+ \times I\!R^d)$ with $d=2,3$ the following equation holds:

$$\frac{d}{dt}\int\limits_{V(t)} f\,dx = \int\limits_{V(t)} f_t + \nabla\cdot(f\boldsymbol{u})\,dx \tag{2.1a}$$

$$= \int\limits_{V(t)} f_t + \int\limits_{\partial V(t)} f\boldsymbol{u}\cdot\boldsymbol{\nu}\,do_x. \tag{2.1b}$$

Proof. First we transform the integral with the parameterization Φ to the fixed reference volume $V_0 = \Phi^{-1}(t, V(t))$. Then integration and time derivative can be interchanged:

$$\begin{aligned}\frac{d}{dt}\int\limits_{V(t)} f(t,x)\,dx &= \frac{d}{dt}\int\limits_{V_0} f(t,\Phi(t,\overline{x}))\,\det(D_x\Phi(t,\overline{x}))\,d\overline{x}\\ &= \int\limits_{V_0} \frac{d}{dt} f(t,\Phi(t,\overline{x}))\,\det(D_x\Phi(t,\overline{x}))\,d\overline{x}.\end{aligned}$$

Now we use the fact, that the time derivative of the Jacobian can be written as (see e.g. [25] paragraph 10)

$$\frac{d}{dt}\det(D_x\Phi(t,\overline{x})) = \det(D_x\Phi(t,\overline{x}))\nabla\cdot\boldsymbol{u}(t,\Phi(t,\overline{x})).$$

Using the chain rule and transforming back to the time-dependent control volume $V(t)$ yields equation (2.1a). Equation (2.1b) follows by applying Gauß's theorem. ∎

2.2 The Navier-Stokes equations

The Navier-Stokes equations describe, under some simplifications, the flow of a liquid or a gas. They can be derived from two physical laws: conservation of mass and conservation of momentum.

2.2.1 Conservation of mass

Let us consider a control volume $V(t)$ with smooth boundary inside the melt Ω^ℓ, moving around with the flow. Due to mass conservation, the mass inside the control volume remains constant. Thus the time derivative of this portion of mass is equal to zero. If we denote the density of the melt by ρ, by applying Reynolds' transport theorem 2.2 we obtain:

$$0 = \frac{d}{dt} \int\limits_{V(t)} \rho \, dx = \int\limits_{V(t)} \rho_t + \nabla \cdot (\rho \boldsymbol{u}) \, dx.$$

Since the control volume was chosen arbitrarily we get the following relation which is also known as continuity equation:

$$\rho_t + \nabla \rho \cdot \boldsymbol{u} + \rho \nabla \cdot \boldsymbol{u} = 0. \tag{2.2}$$

Now we assume that the variation of the density in the melt is very small and thus can be neglected, i.e. the density $\rho_0 = \rho(t,x)$ is constant. Then we get the incompressibility constraint

$$\nabla \cdot \boldsymbol{u} = 0. \tag{2.3}$$

2.2.2 Conservation of momentum

The remaining equations are derived from Newton's laws of motion. For this purpose we take a look at the time derivative of the momentum of a control volume, which should be identical to the appearing forces.

$$\frac{d}{dt} \int\limits_{V(t)} \rho \boldsymbol{u} \, dx = \underbrace{\int\limits_{V(t)} \rho \boldsymbol{f} \, dx}_{\text{volume forces}} + \underbrace{\int\limits_{\partial V(t)} \boldsymbol{\sigma}(\boldsymbol{\nu}) \, do_x}_{\text{boundary forces}} \tag{2.4}$$

We distinguish between two different forces. On the one hand we have forces affecting the volume such as gravity. They are called volume forces and are represented by the variable $\boldsymbol{f}$. On the other hand there exist forces acting on the boundary of the control volume, e.g. compressive stress. They are called boundary forces. For this second term we already used a result of thermodynamics which allows writing it as $\int_{\partial V(t)} \boldsymbol{\sigma}(\boldsymbol{\nu})\, do_x$ with the stress tensor $\boldsymbol{\sigma}$ applied to the outer normal $\boldsymbol{\nu}$.

To get a representation formula for the stress tensor we split it into two parts: The first part consists only of compressive stresses which are independent of the flow velocity. The second part consists of stresses induced by the flow velocity. If we now use properties of flows, e.g. that homogeneous rotations are not subjected to internal stresses, and under some simplifying assumptions, e.g. that the gradient of the velocity remains small and that there are no anisotropic effects, the components of the stress tensor can be written as

$$\sigma_{ij} \;=\; -p\delta_{ij} + \eta D(\boldsymbol{u})_{ij} \quad (i,j=1,\dots,d). \tag{2.5}$$

Here p denotes the pressure in the system, η the dynamic viscosity, $d=2,3$ is the spatial dimension, and $D(\boldsymbol{u})_{ij}$ the symmetric gradient of $\boldsymbol{u}$, i.e.

$$D(\boldsymbol{u})_{ij} \;=\; \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right).$$

Now we consider one component of the left hand side of equation (2.4). Assuming again that the density ρ_0 is constant we obtain

$$\begin{aligned}\frac{d}{dt}\int\limits_{V(t)} \rho u_i\, dx \;&=\; \rho_0 \int\limits_{V(t)} u_{i,t} + \nabla\cdot(u_i\boldsymbol{u})\, dx\\ &=\; \rho_0 \int\limits_{V(t)} u_{i,t} + u_i\nabla\cdot\boldsymbol{u} + \nabla u_i\cdot\boldsymbol{u}\, dx.\end{aligned}$$

Therefore, using the incompressibility constraint (2.3) and Gauß's theorem gives

$$\rho_0\left(\boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u}\right) \;=\; \nabla\cdot\boldsymbol{\sigma} + \rho_0\boldsymbol{f}.$$

Using again the incompressibility constraint (2.3), the divergence of the stress tensor as defined in equation (2.5) can be written as

$$\nabla \cdot \boldsymbol{\sigma} \;=\; -\nabla p + \eta \Delta \boldsymbol{u}. \tag{2.6}$$

Combining the above results we can state the incompressible Navier-Stokes equations for the liquid domain Ω^ℓ

$$\begin{aligned} \rho_0 \left(\boldsymbol{u}_t + \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) - \eta \Delta \boldsymbol{u} + \nabla p \;&=\; \rho_0 \boldsymbol{f}, \\ \nabla \cdot \boldsymbol{u} \;&=\; 0, \end{aligned} \tag{2.7}$$

with initial condition $\boldsymbol{u} = \boldsymbol{u}_0$. On the boundary we assume no slip boundary conditions which are introduced in Section 2.5.

2.2.3 Boussinesq approximation

For the derivation of the incompressible Navier-Stokes equations we assumed that the variation of density in the melt can be neglected. In fact, if the variation of temperature and concentration fields is small and the flow velocity stays small this is as well a reasonable simplification. Nevertheless, these small variations in density have a big influence on the term of the volume forces represented by $\boldsymbol{f}$ (cf. equation (2.4)): Areas with different temperature or composition cause very small differences in density generating thermal and solutal lifting forces under the influence of gravity. In consideration of these effects we use the Navier-Stokes equations in Boussinesq approximation. In principle this approach supposes that the density is constant, except for exactly the volume force term.

Let ρ_0 be the density of the melt at a fixed temperature θ_0 with fixed concentrations $c_{i,0}$. If we assume that the density depends linearly on small variations of these fixed values we can write

$$\rho \;=\; \rho_0 \Big(1 - \beta_\theta (\theta - \theta_0) - \sum_{i=1,2} \beta_{c_i} (c_i - c_{i,0}) \Big),$$

where β_θ denotes the thermal expansion coefficient and β_{c_i} the solutal expansion coefficients in the melt. Furthermore, we only assume one volume force, the constant gravity $\boldsymbol{g}$.

Defining the kinematic viscosity as $\nu := \eta/\rho_0$ and introducing a modified pressure $\overline{p}$ as

$$\overline{p}(t,x) = p(t,x) - \rho_0 \, \boldsymbol{g} \cdot x,$$

we can state the Navier-Stokes equations in Boussinesq approximation

$$\begin{aligned} \boldsymbol{u}_t + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \frac{1}{\rho_0} \nabla \overline{p} &= -\Big(\beta_\theta (\theta - \theta_0) + \sum_{i=1,2} \beta_{c_i} (c_i - c_{i,0}) \Big) \boldsymbol{g}, \\ \nabla \cdot \boldsymbol{u} &= 0, \end{aligned} \tag{2.8}$$

with identical initial values and boundary conditions as in the last section.
In what follows, we only consider the modified pressure $\overline{p}$. For the sake of simplicity it is again denoted by p.

2.3 Heat equation

For the derivation of the heat equation we consider the energy of the system: In a control volume the total energy is given as the sum of the kinetic and the internal energy,

$$\int_{V(t)} \underbrace{\rho \frac{|u|^2}{2}}_{\text{kin. energy}} \, dx \; + \int_{V(t)} \underbrace{\rho \varepsilon}_{\text{int. energy}} \, dx.$$

In Section 2.2.2 we regarded the forces acting on a control volume. We distinguished between the boundary forces ($\int_{\partial V(t)} \boldsymbol{\sigma}(\boldsymbol{\nu}) \, do_x$) and the volume forces ($\int_{V(t)} \rho \boldsymbol{f} \, dx$). Thus, the related power consists of the two terms

$$\int_{\partial V(t)} \boldsymbol{\sigma}(\boldsymbol{\nu}) \cdot \boldsymbol{u} \, do_x + \int_{V(t)} \rho \, \boldsymbol{f} \cdot \boldsymbol{u} \, dx.$$

Due to differences of temperature inside and outside the control volume we get an additional heat flux over its boundary denoted by $\boldsymbol{q}$. Collecting all parts the law of energy conservation yields

$$\begin{aligned} \frac{d}{dt} \int_{V(t)} \rho \frac{|u|^2}{2} + \rho \varepsilon \, dx &= - \int_{\partial V(t)} \boldsymbol{q} \cdot \boldsymbol{\nu} \, do_x + \int_{\partial V(t)} \boldsymbol{\sigma}(\boldsymbol{\nu}) \cdot \boldsymbol{u} \, do_x \\ &\quad + \int_{V(t)} \rho \, \boldsymbol{f} \cdot \boldsymbol{u} \, dx. \end{aligned} \tag{2.9}$$

Taking a closer look at the right hand side of equation (2.9), we get by the symmetry of the stress tensor $\boldsymbol{\sigma}$ and with Gauß's theorem

$$\text{r.h.s.} \;=\; \int\limits_{V(t)} -\nabla\cdot\boldsymbol{q} + (\nabla\cdot\boldsymbol{\sigma})\cdot\boldsymbol{u} + \eta\left(D(\boldsymbol{u}):\nabla\boldsymbol{u}\right) - p\,\nabla\cdot\boldsymbol{u} + \rho\,\boldsymbol{f}\cdot\boldsymbol{u}\,dx.$$

For the left hand side we use Reynolds' transport theorem 2.2:

$$\begin{aligned}\text{l.h.s.} \;=\; &\int\limits_{V(t)} \big(\underbrace{\rho_t + \nabla\cdot(\rho\boldsymbol{u})}_{=0}\big)\Big(\frac{|\boldsymbol{u}|^2}{2} + \varepsilon\Big) + \big(\underbrace{\rho(\boldsymbol{u}_t + \boldsymbol{u}\cdot\nabla\boldsymbol{u})}_{=\nabla\cdot\boldsymbol{\sigma}+\rho\boldsymbol{f}}\big)\cdot\boldsymbol{u}\,dx\\ &+ \int\limits_{V(t)} \rho(\varepsilon_t + \boldsymbol{u}\cdot\nabla\varepsilon)\,dx,\end{aligned}$$

with respect to the continuity equation (2.2), equation (2.6), and the Navier-Stokes equations (2.7). If the temperature gradient in the melt is not too big we can apply Fourier's law. Thus, the heat flux can be written as

$$\boldsymbol{q} \;=\; -k\nabla\theta, \tag{2.10}$$

where k denotes the heat conductivity of the melt. Furthermore, the following thermodynamical relations between the internal energy ε, the entropy s and the thermodynamical pressure p^{th} hold (see e.g. [36]):

$$\varepsilon_t \;=\; \theta s_t + \frac{p^{th}}{\rho^2}\rho_t, \qquad \nabla\varepsilon \;=\; \theta\nabla s + \frac{p^{th}}{\rho^2}\nabla\rho.$$

Combining these two equations we obtain the relation

$$\varepsilon_t + \boldsymbol{u}\cdot\nabla\varepsilon \;=\; \theta(s_t + \boldsymbol{u}\cdot\nabla s) + \frac{p^{th}}{\rho^2}(\underbrace{\rho_t + \boldsymbol{u}\cdot\nabla\rho}_{=0}),$$

where the last term vanishes due to the continuity equation (2.2) and the incompressibility constraint (2.3). For the left hand side of equation (2.9) this leads to

$$\text{l.h.s.} \;=\; \int\limits_{V(t)} (\nabla\cdot\boldsymbol{\sigma})\cdot\boldsymbol{u} + \rho\boldsymbol{f}\cdot\boldsymbol{u} + \rho\theta(s_t + \boldsymbol{u}\cdot\nabla s)\,dx.$$

Altogether, we get the general heat equation

$$\rho\theta(s_t + \boldsymbol{u}\cdot\nabla s) \;=\; \eta\left(D(\boldsymbol{u}):\nabla\boldsymbol{u}\right) + \nabla\cdot(k\nabla\theta). \tag{2.11}$$

For obtaining an easier representation of the heat equation without occurrence of the entropy s we need two more assumptions. First, the flow velocity should be small compared to the acoustic velocity. Second, the maximal temperature difference in the melt should be small. In our setting this is fulfilled. Hence, the variation in density which arises from variations in pressure and temperature is small and can be neglected. Note, that the same assumption was already used in Section 2.2.1. Now we get the following relation between the entropy s and the temperature θ (cf. [36]) with constant specific heat c:

$$\theta s_t \;=\; c\theta_t, \qquad \theta\nabla s \;=\; c\nabla\theta.$$

Furthermore, we assume that the change of temperature due to friction can be neglected as well. This is reasonable since simulations show that we have small velocity gradients and also a small dynamic viscosity η. Hence, the viscous term $\eta\,(D(\boldsymbol{u})\!:\!\nabla\boldsymbol{u})$ in equation (2.11) vanishes and we get a simpler form of the heat equation in the melt which we will use in the sequel:

$$\theta_t + \boldsymbol{u}\cdot\nabla\theta - \frac{k_\ell}{\rho_0 c_0}\Delta\theta \;=\; 0 \qquad \text{in } \Omega^\ell. \tag{2.12}$$

Here $k = k_\ell$ denotes again the heat conductivity, $c = c_0$ the specific heat and $\rho = \rho_0$ the constant density.

The heat equation is also valid in the solid phase Ω^s. Due to the fact that there is no flow velocity in a crystalline material the convective term $\boldsymbol{u}\cdot\nabla\theta$ vanishes from equation (2.12). If we refer to the equation in the solid phase, we denote the heat conductivity k by k_s.

2.4 Conservation of mass

The melt in the pot is a compound of the three different materials cadmium, mercury, and tellurium. Denoting by c_i $(i = 0, 1, 2)$ the respective relative concentrations the following identity holds:

$$c_0 + c_1 + c_2 \;=\; 1.$$

Obviously, it is sufficient to calculate only two of the concentrations. Thus, the concentration c_0 (cadmium) will not be considered in the sequel.

If the concentrations inside and outside a control volume $V(t)$ differ we get a flux over the boundary due to diffusion. Considering the concentration of one component in $V(t)$ mass conservation yields

$$\frac{d}{dt}\int\limits_{V(t)} c_i\,dx \;=\; -\int\limits_{\partial V(t)} \boldsymbol{m}_i\cdot\boldsymbol{\nu}\,do_x, \tag{2.13}$$

where $\boldsymbol{m}_i$ denotes the mass flux over the boundary $\partial V(t)$. Fick's law gives the link between the flux and the gradient of the concentration

$$\boldsymbol{m}_i \;=\; -D_i\nabla c_i, \tag{2.14}$$

where D_i denotes the diffusion coefficient of the concentration c_i. Applying Reynolds' transport theorem 2.2 to the left hand side of equation (2.13) and Gauß's theorem and Fick's law to the right hand side we obtain

$$\int\limits_{V(t)} c_{i,t}+\boldsymbol{u}\cdot\nabla c_i\,dx \;=\; \int\limits_{V(t)} \nabla\cdot(D_i\nabla c_i)\,dx.$$

Since the control volume was arbitrary this results in the partial differential equation for the conservation of mass:

$$c_{i,t}+\boldsymbol{u}\cdot\nabla c_i - D_i^\ell\Delta c_i \;=\; 0 \qquad \text{in } \Omega^\ell \quad (\,i=1,2\,).$$

Again this equation holds also in the solid part Ω^s, without convective term and with diffusion coefficient $D_i = D_i^s$.

2.5 Boundary conditions

During the growth of the epitaxial layer onto the substrate we get a thin boundary layer in which the phase transition takes place. For our model we assume that the thickness of this boundary layer is infinitesimal, i.e. it collapses to an interface. Models of this kind are known as "sharp interface" models. In contrast, the so called "phase field" models allow a thin boundary layer ("mushy region") where the status of the material is assumed to be neither solid nor liquid.

In this section we focus on the energy and mass balance on this interface which we denote by Γ. We derive interface conditions such that energy and mass conservation are fulfilled and look at the physical properties of the materials at the phase boundary. Furthermore, the boundary conditions on the outer boundary for flow velocity, temperature, and concentrations are introduced.

2.5.1 Boundary conditions on the outer boundary

At the walls of the melting pot as well as on the surface of the substrate we have no slip boundary conditions for the flow in the melt, i.e. the flow velocity is equal to zero. On the surface at the top of the pot the boundary is not fixed and thus a tangential flow would be possible. The influence of this tangential flow on the flow in the interior of the melt and in particular on the flow near the interface is very small. Thus, we assume no slip boundary conditions here as well and get

$$\boldsymbol{u} = 0 \qquad \text{on } \partial\Omega^\ell. \tag{2.15}$$

For the temperature simulation and measuring data at the exterior of the pot are available. We denote it by θ_{ext}. The temperature difference between the inner and the outer side of the pot causes a heat flux. If we assume a linear dependency between this difference and the normal derivative of the temperature which represents the heat flux normal to the boundary, we end up with a Robin boundary condition. Denoting by R the heat exchange coefficient we get

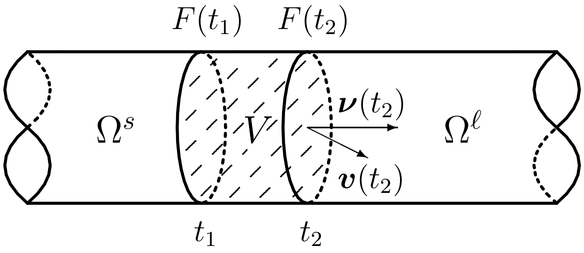
$$k\frac{\partial\theta}{\partial\boldsymbol{\nu}} = R\,(\theta_{ext} - \theta) \qquad \text{on } \partial\Omega. \tag{2.16}$$

For the components of the concentrations the boundary condition follows from mass conservation: No mass can vanish out of the pot during the whole process. Thus, we get homogenous Neumann boundary conditions

$$\frac{\partial c_i}{\partial\boldsymbol{\nu}} = 0 \qquad \text{on } \partial\Omega. \tag{2.17}$$

2.5.2 Stefan condition for temperature and concentrations

The relation between the values of temperature, and concentrations respectively, in the solid and the liquid phase is given by a Stefan condition. For deriving it we regard a small piece F of the interface Γ which is moving with velocity $\boldsymbol{v}\cdot\boldsymbol{\nu}$ in normal direction. The volume that we get if we integrate the surface area of F in the time interval (t_1,t_2) is denoted by V, the heat flux over F out of the liquid and into the solid phase by $\boldsymbol{q}_\ell$ ($\boldsymbol{q}_s$ analogous), and by L the latent heat which represents the gain in energy due to the phase transition from liquid to solid.



Now balance of energy implies that the difference of the heat fluxes in the time interval (t_1,t_2) must be equal to the gain in energy of the grown crystal with volume V:

$$(2.18)\qquad \int_{t_1}^{t_2}\int_{F(t)} \boldsymbol{q}_\ell\cdot\boldsymbol{\nu}-\boldsymbol{q}_s\cdot\boldsymbol{\nu}\,do_x\,dt \;=\; L\rho_0 V \;=\; L\rho_0\int_{t_1}^{t_2}\int_{F(t)}\boldsymbol{v}\cdot\boldsymbol{\nu}\,do_x\,dt.$$

Assuming local thermodynamical equilibrium yields that the temperature field is smooth inside the subdomains Ω^s,Ω^ℓ and overall continuous. Nevertheless the gradient of the temperature may jump on the interface Γ. Therefore, the normal derivatives of the temperature on the interface is defined as follows: For any sequences $(x_j)_{j\in\mathbb{N}}\subset\Omega^\ell$ and $(x_k)_{k\in\mathbb{N}}\subset\Omega^s$ converging to $x\in\Gamma$, let

$$(2.19)\qquad \frac{\partial\theta^\ell}{\partial\boldsymbol{\nu}}(t,x) \;=\; \lim_{j\to\infty}\frac{\partial\theta}{\partial\boldsymbol{\nu}}(t,x_j)\quad\text{and}\quad \frac{\partial\theta^s}{\partial\boldsymbol{\nu}}(t,x)\;=\;\lim_{k\to\infty}\frac{\partial\theta}{\partial\boldsymbol{\nu}}(t,x_k).$$

Since the temperature is smooth in both subdomains, the definitions are independent of the choice of the sequences. Now, the heat fluxes in equation (2.18) can be replaced by applying Fourier's law (2.10). Taking further the limit $t_2\to t_1$ yields the classical Stefan condition for the temperature:

$$(2.20)\qquad \left[k\frac{\partial\theta}{\partial\boldsymbol{\nu}}\right]_s^\ell := k_\ell\frac{\partial\theta^\ell}{\partial\boldsymbol{\nu}}-k_s\frac{\partial\theta^s}{\partial\boldsymbol{\nu}} \;=\; -L\rho_0\boldsymbol{v}\cdot\boldsymbol{\nu}.$$

For a concentration c_i argumentation is similar. The balance of mass implies that the difference of the mass fluxes $\boldsymbol{m}_\ell, \boldsymbol{m}_s$ over Γ must coincide with the change of concentration in the volume V:

$$\int_{t_1}^{t_2} \int_{F(t)} \boldsymbol{m}_\ell \cdot \boldsymbol{\nu} - \boldsymbol{m}_s \cdot \boldsymbol{\nu} \, do_x \, dt \;=\; \int_{t_1}^{t_2} \int_{F(t)} (c_i^\ell - c_i^s) \boldsymbol{v} \cdot \boldsymbol{\nu} \, do_x \, dt. \tag{2.21}$$

Here c_i^ℓ and c_i^s denote the limit of the concentration on the interface Γ seen from the liquid and the solid side. Hence, with sequences as in equation (2.19), they are defined as

$$c_i^\ell(t,x) := \lim_{j\to\infty} c_i^\ell(t,x_j) \quad \text{and} \quad c_i^s(t,x) := \lim_{k\to\infty} c_i^s(t,x_k), \tag{2.22}$$

assuming that the concentration is smooth in each subdomain. Note, that in general equation (2.21) requires the concentration c_i being discontinuous over the interface Γ. Applying Fick's law (2.14) in equation (2.21) yields the Stefan condition for the concentration c_i

$$\left[D_i \frac{\partial c_i}{\partial \boldsymbol{\nu}} \right]_s^\ell \;=\; -\,[c_i]_s^\ell \, \boldsymbol{v} \cdot \boldsymbol{\nu}, \tag{2.23}$$

where the normal derivatives are defined analogously to the normal derivatives of the temperature in equation (2.19).

2.5.3 The phase diagram

In the last section we dealt with the question which boundary conditions on the interface Γ are necessary for ensuring energy and mass conservation to hold. In this section we clarify how temperature behaves on the interface and how the composition of the grown crystal layer depends on the composition of the components in the melt.

Assuming local thermodynamical equilibrium on the interface yields that the temperature has to be continuous over Γ and thus is equal to the melting temperature θ_m of the growing crystal layer. For the melting temperature we consider the general approach writing it as the sum of three different components (cf. [53]):

$$\theta_m \;=\; \theta_{s.t.} + \theta_{n.e.} + \theta_{imp.}. \tag{2.24}$$

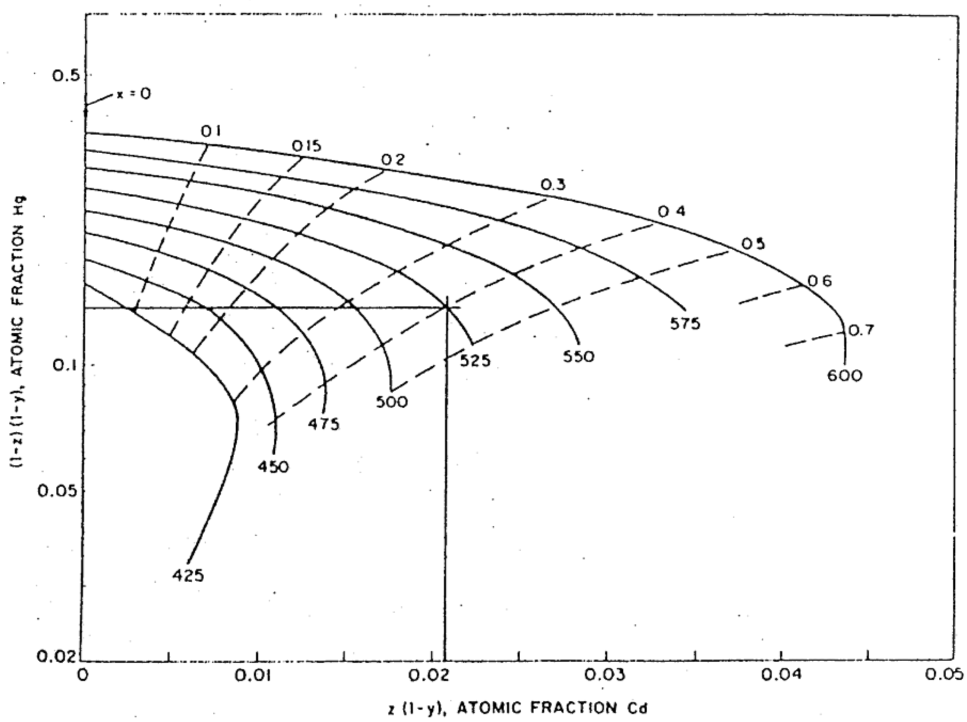


Figure 2.2: Phase diagram for a tellurium-rich melt with composition $(Hg_{1-z}Cd_z)_{1-y}Te_y$. The melting temperature is represented by the solid lines, the composition x of the crystal $Hg_{1-x}Cd_xTe$ by the dashed lines (from [27]).

The first term $\theta_{s.t.}$ represents the portion of temperature which is related to the surface tension of the melt at the phase boundary and thus depends on the curvature of the interface. Experiments show that the growth of the layer is nearly uniform. Thus, curvature effects can be neglected.

The second term $\theta_{n.e.}$ stands for the influence of non equilibrium states on the melting temperature. Because of the very small growth velocity of the layer we assume that temperature is always near the equilibrium state on the interface. Therefore, this term can be neglected as well.

The third term $\theta_{imp.}$ is related to the composition and the impurities of the melt. We will concentrate on this part and use the results of physical experiments done by the group of Harman (cf. [27]) to get appropriate values.

To obtain the melting temperature and the composition of the crystal for different compositions in the melt the group of Harman performed physical experiments (cf. [27]). Their results are summarized in the phase diagram (see Figure 2.2). In [14] equations for the phase diagram of Harman can be found giving a good approximation to melting temperature and composition in the crystal for our configuration. For the composition $(Hg_{1-z}Cd_z)_{1-y}Te_y$ in the melt the following equations for the melting temperature $\theta_m = T\,℃$ and the composition x of the crystal $Hg_{1-x}Cd_xTe$ hold:

$$\begin{aligned} T &= 1102 + 250z + 420yz - 785y, \\ x &= \frac{z}{0.22 + 0.78z}. \end{aligned} \tag{2.25}$$

Note, that the concentrations of the materials jump at the interface Γ and thus are discontinuous. Therefore, we already defined two different values for each concentration on Γ (cf. equation (2.22)) denoted by c_i^ℓ and c_i^s. With the above notation the concentration of mercury in the liquid phase is given by $c_1^\ell = (1-z)(1-y)$ and by $c_1^s = (1-x)/2$ in the solid phase. For the concentration of tellurium we get $c_2^\ell = y$ and $c_2^s = 1/2$. Rewriting the phase diagram equations (2.25) yields:

$$\theta_m = \left(1352 - 365c_2^\ell + 420c_1^\ell - 670\frac{c_1^\ell}{1-c_2^\ell}\right)℃ \qquad \text{on } \Gamma, \tag{2.26a}$$

$$c_1^s = \frac{0.11c_1^\ell}{1 - 0.78c_1^\ell - c_2^\ell} \qquad \text{on } \Gamma, \tag{2.26b}$$

$$c_2^s = \frac{1}{2} \qquad \text{on } \Gamma. \tag{2.26c}$$

This crystal has a structure consisting of precisely 50% tellurium atoms whereas the atoms of cadmium and mercury are interchangeable. This can be seen in equation (2.26b) and (2.26c).

2.6 Material constants

The equations derived in this chapter contain several material constants. Since the involved materials are not in the main focus of material science it is difficult to find the values for the temperature of the process. We had to collect them from many different sources (cf. [14, 38, 29, 31, 23, 37, 35]). A summary of all constants with utilized values is given in Table 2.1.

constant	value	unit	description
ν	$2 \cdot 10^{-6}$	$\frac{m^2}{s}$	kinematic viscosity (melt)
ρ_0	$6.1 \cdot 10^{3}$	$\frac{kg}{m^3}$	mean density
β_θ	$1.3 \cdot 10^{-4}$	$\frac{1}{℃}$	thermal expansion coefficient (melt)
β_{c_1}	$-3.0 \cdot 10^{-1}$	–	solutal expansion coefficient of Hg
β_{c_2}	$3.0 \cdot 10^{-2}$	–	solutal expansion coefficient of Te
$\boldsymbol{g}$	$(0, 0, -9.81)$	$\frac{m}{s^2}$	gravitation vector
R	$4.0 \cdot 10^{3}$	$\frac{W}{m^2\, ℃}$	heat exchange coefficient
k_ℓ	10.0	$\frac{kg\, m}{s^3\, ℃}$	heat conductivity (melt)
c_0	$2.0 \cdot 10^{2}$	$\frac{J}{kg\, ℃}$	mean specific heat
$a_\ell = \frac{k_\ell}{\rho_0\, c_0}$	$8.2 \cdot 10^{-6}$	$\frac{m^2}{s}$	heat conduction coefficient (melt)
$a_s = \frac{a_\ell}{2}$	$4.1 \cdot 10^{-6}$	$\frac{m^2}{s}$	heat conduction coefficient (substrate)
D_1^ℓ	$5.7 \cdot 10^{-8}$	$\frac{m^2}{s}$	diffusion coefficient of Hg (melt)
D_2^ℓ	$8.2 \cdot 10^{-9}$	$\frac{m^2}{s}$	diffusion coefficient of Te (melt)
D_1^s	$2.5 \cdot 10^{-15}$	$\frac{m^2}{s}$	diffusion coefficient of Hg (substrate)
D_2^s	$1.2 \cdot 10^{-15}$	$\frac{m^2}{s}$	diffusion coefficient of Te (substrate)
L	$1.3 \cdot 10^{5}$	$\frac{J}{kg}$	latent heat

Table 2.1: Material constants with units and description.

For the heat conductivity of the substrate material at a temperature of about $470\,℃$ we did not get reliable values. Thus, we chose the value to be half as big as the value in the melt. In fact it does not have a significant influence on the results of the simulations. For some of the other parameters we found a wide range of values depending on the used measurement method. We always tried to use the most reliable value or calculated a mean value of the available data.

2.7 The non-dimensional system

For the mathematical treatment of the equations it is useful to rescale them and dispose the physical units. For this purpose we define a characteristic length, time, and velocity:

characteristic length	X	$0.01\,m$	$(\widehat{=}\,1\,cm\,)$
characteristic time	T	$1\,s$	
characteristic velocity	V	$0.01\,\frac{m}{s}$	$(\widehat{=}\,1\,\frac{cm}{s}\,)$

Now we use these characteristic quantities to rescale all unknowns of the system collected in Table 2.2. Setting

$$\hat{t} \;=\; \frac{t}{T} \quad \text{and} \quad \hat{x} \;=\; \frac{x}{X} \tag{2.27}$$

we define the rescaled flow velocity and pressure in the Navier-Stokes equations as

$$\hat{\boldsymbol{u}}(\hat{t},\hat{x}) := \frac{\boldsymbol{u}(t,x)}{V} \quad \text{and} \quad \hat{p}(\hat{t},\hat{x}) := \frac{p(t,x)}{\rho_0\,V^2},$$

the rescaled temperature and the interface velocity as

$$\hat{\theta}(\hat{t},\hat{x}) := \frac{c_0}{L}\,(\theta(t,x) - \theta_0) \quad \text{and} \quad \hat{v}_\Gamma(\hat{t},\hat{x}) := \frac{v_\Gamma(t,x)}{V}, \tag{2.28}$$

and the rescaled concentrations as

$$\hat{c}_i(\hat{t},\hat{x}) := c_i(t,x) \qquad (\,i = 1,2\,).$$

The offset of the temperature θ_0 is chosen to be the melting temperature for the initial composition of cadmium ($c_{0,0}\!=\!7.623\text{‰}$), mercury ($c_{1,0}\!=\!14.6377\%$), and tellurium ($c_{2,0}\!=\!84.6\%$) which yields $\theta_0 = 467.85\,℃$. For the time derivative of the temperature, for instance, using equation (2.27) and (2.28) now yields

$$\theta_t(t,x) \;=\; \frac{d}{dt}\big(\tfrac{L}{c_0}\hat{\theta}(\hat{t},\hat{x}) - \theta_0\big) \;=\; \frac{d}{dt}\big(\tfrac{L}{c_0}\hat{\theta}(\tfrac{t}{T},\tfrac{x}{X})\big) \;=\; \tfrac{L}{c_0 T}\hat{\theta}_{\hat{t}}(\hat{t},\hat{x}),$$

variable	unit	description
$\boldsymbol{u}$	$\frac{m}{s}$	flow velocity of the melt
p	$\frac{kg}{ms^2}$	pressure
θ	℃	temperature
c_1	$-$	relative concentration of mercury
c_2	$-$	relative concentration of tellurium
v_Γ	$\frac{m}{s}$	velocity of the interface (in normal direction)

Table 2.2: Unknowns of the system with physical unit and description.

and for the temperature gradient we get

$$\nabla_x\theta_t(t,x) \;=\; \nabla_x\big(\tfrac{L}{c_0}\hat{\theta}(\hat{t},\hat{x}) - \theta_0\big) \;=\; \nabla_x\big(\tfrac{L}{c_0}\hat{\theta}(\tfrac{t}{T},\tfrac{x}{X})\big) \;=\; \tfrac{L}{c_0X}\nabla_{\hat{x}}\hat{\theta}(\hat{t},\hat{x}).$$

Substituting all derivatives of the temperature in equation (2.12) we obtain

$$\tfrac{L}{c_0T}\hat{\theta}_{\hat{t}} + \tfrac{VL}{c_0X}\hat{\boldsymbol{u}}\cdot\nabla_{\hat{x}}\theta - \tfrac{L}{c_0X^2}\nabla_{\hat{x}}\cdot\big(a\nabla_{\hat{x}}\hat{\theta}\big) \;=\; 0.$$

Thus, the non-dimensional form of the heat equation results from the last equation by multiplying it with the constant $\frac{c_0T}{L}$:

$$\hat{\theta}_{\hat{t}} + \underbrace{\tfrac{VT}{X}}_{=1}\,\hat{\boldsymbol{u}}\cdot\nabla_{\hat{x}}\theta - \nabla_{\hat{x}}\cdot\big(\tfrac{aT}{X^2}\nabla_{\hat{x}}\hat{\theta}\big) \;=\; 0,$$

with the non-dimensional heat conduction coefficient $\hat{a} = \frac{aT}{X^2}$. Proceeding similarly with all other equations as well as the boundary conditions we obtain the non-dimensional system. For the sake of simplicity we drop the hat "^" for all non-dimensional variables and use again their former notation whereas we keep it for all material constants. The complete description of the problem is now given as follows:

System of partial differential equations

In Ω^s the flow velocity $\boldsymbol{u} = 0$. In Ω^ℓ we have:

$$\begin{aligned} \boldsymbol{u}_t + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \frac{1}{Re} \Delta \boldsymbol{u} + \nabla p \; &= \; -\Big(\hat{\beta}_\theta \, \theta + \sum_{i=1,2} \hat{\beta}_{c_i} (c_i - c_{i,0})\Big) \hat{\boldsymbol{g}}, \\ \nabla \cdot \boldsymbol{u} \; &= \; 0. \end{aligned} \tag{2.29}$$

For the temperature and the concentrations in $\Omega^s \cup \Omega^\ell$ we have:

$$\begin{aligned} \theta_t + \boldsymbol{u} \cdot \nabla \theta - \nabla \cdot (\hat{a} \nabla \theta) \; &= \; 0, \\ c_{i,t} + \boldsymbol{u} \cdot \nabla c_i - \nabla \cdot (\hat{D}_i \nabla c_i) \; &= \; 0 \quad (\, i = 1, 2\,). \end{aligned} \tag{2.30}$$

Conditions on the interface

On Γ the Stefan conditions hold for the temperature and the concentrations:

$$\left[\hat{a} \frac{\partial \theta}{\partial \boldsymbol{\nu}} \right]_s^\ell \; = \; -v_\Gamma \quad \text{and} \quad \left[\hat{D}_i \frac{\partial c_i}{\partial \boldsymbol{\nu}} \right]_s^\ell \; = \; -\, [c_i]_s^\ell \, v_\Gamma \quad (\, i = 1, 2\,). \tag{2.31}$$

The remaining equations on Γ are given by the phase diagram:

$$\begin{aligned} \theta \; &= \; 1.36 - 0.56 c_2^\ell + 0.65 c_1^\ell - \frac{1.03 c_1^\ell}{1 - c_2^\ell}, \\ c_1^s \; &= \; \frac{0.11 c_1^\ell}{1 - 0.78 c_1^\ell - c_2^\ell} \quad \text{and} \quad c_2^s \; = \; \frac{1}{2}. \end{aligned} \tag{2.32}$$

Outer boundary conditions

$$\begin{aligned} \boldsymbol{u} \; &= \; 0 && \text{on } \partial\Omega^\ell, \\ \frac{\partial \theta}{\partial \boldsymbol{\nu}} \; &= \; \frac{\hat{R}}{\hat{a}_\ell} \big(\hat{\theta}_{ext} - \theta \big) && \text{on } \partial\Omega, \\ \frac{\partial c_i}{\partial \boldsymbol{\nu}} \; &= \; 0 \quad (\, i = 1, 2\,) && \text{on } \partial\Omega. \end{aligned} \tag{2.33}$$

Initial values

$$\begin{aligned} \boldsymbol{u}(\,\cdot\,, 0) \; &= \; 0 && \text{in } \Omega^\ell, \\ \theta(\,\cdot\,, 0) \; &= \; 0 && \text{in } \Omega^\ell \cup \Omega^s, \\ c_i(\,\cdot\,, 0) \; &= \; c_{i,0} \quad (\, i = 1, 2\,) && \text{in } \Omega^\ell \cup \Omega^s. \end{aligned} \tag{2.34}$$

Values of the non-dimensional parameters

parameter	non-dim. value
Re	50
$\hat{\beta}_\theta$	$8.45 \cdot 10^{-2}$
$\hat{\beta}_{c_1}$	$-3.0 \cdot 10^{-1}$
$\hat{\beta}_{c_2}$	$3.0 \cdot 10^{-2}$
$\hat{\boldsymbol{g}}$	$(0, 0, -981)$
$\hat{R}$	$3.3 \cdot 10^{-1}$

parameter	non-dim. value
$\hat{a}_\ell$	$8.2 \cdot 10^{-2}$
$\hat{a}_s$	$4.1 \cdot 10^{-2}$
$\hat{D}_1^\ell$	$5.7 \cdot 10^{-4}$
$\hat{D}_2^\ell$	$8.2 \cdot 10^{-5}$
$\hat{D}_1^s$	$2.5 \cdot 10^{-11}$
$\hat{D}_2^s$	$1.2 \cdot 10^{-11}$

2.8 Some remarks on the model

The derived model is developed to allow an efficient numerical simulation of the liquid phase epitaxy in two and three space dimensions. A model including all physical aspects of this difficult process would be far from practical benefit. Nevertheless, we want to discuss some aspects we did not consider.

Segregation. Due to gravity the heaviest component mercury will start sinking to the bottom of the melting pot. In contrast a high flow velocity prevents this effect from happening too fast. Segregation is one of the reasons why triggering a high flow velocity in the melt by a large temperature gradient along the side walls of the melting pot is crucial.

Missing gradient terms. Taking a closer look at the derivation of the heat equation we realize that the heat flux (cf. Fourier's law (2.10)) in a material with several components does not only depend on the temperature gradient but also on the chemical potential of the involved materials. Furthermore Fick's law (cf. (2.14)) only holds for a melt with constant temperature and thus an additional term with the temperature gradient should be taken into account. Since the mixing effect of the flow in the melt is dominating these terms can be neglected.

'sharp interface' versus 'phase field'. For our problem we have made the decision to use a sharp interface model. Since our main interest is the calculation of the thickness of the emerging film and thus the exact position of the interface, a sharp interface model is much more appropriate: The big disadvantage of phase field models is that they need a very high grid resolution near the interface to get the accurate position of the interface itself. If a grid resolution of about one micron around the interface is required this yields about four millions degrees of freedom for the nonlinear system in a 2-dimensional simulation. Since with our model calculations on much coarser grids are possible, only around thirty thousand unknowns in a 2-dimensional simulation and around one million unknowns in a 3-dimensional simulation are required to obtain satisfying results.

Topological changes and nucleation. The disadvantage of sharp interface models is that they do neither allow topological changes of the phase boundary nor nucleation. However, in our process this is no limitation. The industrial production of the wafers show that the growth of the layer is nearly uniform and thus no topological changes occur. Furthermore, experiments showed that the pure melt without the substrate as nucleation area can be undercooled more than fifty degrees below the melting temperature without occurrence of nucleation. In our model the phase boundary exists from the beginning of the simulation and is constituted to be the surface between the solid and the liquid part, exactly where the crystal growth starts in the real process.

Chapter 3

The Mathematical Model

In this chapter we present some basic notations and give a short introduction to Sobolev spaces. In order to understand the spaces required for a weak solution of the physical model as well as the arising difficulties we derive a linearized model problem and study its properties. We then present a variational formulation of the problem. We consider two different approaches. The first is the weak form of the full system on the halfed domain with respect to symmetry. The second omits the geometrical motion of the phase boundary. Both approaches are suited for discretization with finite elements which we will use for the simulations. Existence and uniqueness of the full system are not covered, since both are still open questions.

3.1 Sobolev spaces

For the derivation of the weak formulation we need Sobolev spaces for the calculation domain as well as for its boundary. In this section we present a short introduction to this field. The definitions and results are mainly taken from the books of Alt [2], Lions/Magenes [40], and Hackbusch [26]. Further results can be found in [1, 41, 55]. For extensions of the results to domains with curvilinear, polygonal, or polyhedral boundaries we refer to Grisvard [24].

Let Ω be a bounded domain of $\mathbb{R}^d$ with Lipschitz boundary $\Gamma = \partial\Omega$. For $p \geq 1$

let $L^p(\Omega)$ be the classical Banach space consisting of all Lebesgue measurable and p-integrable functions φ on Ω with the corresponding norm

$$\|\varphi\|^p_{L^p(\Omega)} \;:=\; \int_\Omega |u(x)|^p \, dx \;<\; \infty.$$

The Banach space $L^\infty(\Omega)$ consists of all Lebesgue measurable functions φ on Ω which are essentially bounded, equipped with the norm

$$\|\varphi\|_{L^\infty(\Omega)} \;:=\; \operatorname*{ess\,sup}_{x\in\Omega} |\varphi(x)|.$$

The functions with all weak derivatives up to k-th order ($k\in\mathbb{N}_0$) lying in $L^p(\Omega)$ span the Sobolev space $H^{k,p}(\Omega)$. This is again a Banach space when equipped with the norm

$$\|\varphi\|_{k,p,\Omega} \;=\; \Big(\sum_{|\alpha|\le k} \|D^\alpha\varphi\|^p_{L^p(\Omega)}\Big)^{\frac{1}{p}},$$

where

$$D^\alpha \;:=\; \frac{\partial^{|\alpha|}}{\prod_{i=1}^d \partial x_i^{\alpha_i}}, \quad \alpha=(\alpha_1,\dots,\alpha_d), \quad \alpha_i\in\mathbb{N}_0, \quad |\alpha|=\sum_{i=1}^d \alpha_i$$

denotes the weak derivative for the multi-index α. In the special case $p=2$ we use the common notation $H^k(\Omega)$ instead of $H^{k,2}(\Omega)$ and drop the index p for the norm, i.e.

$$\|\varphi\|_{k,\Omega} \;=\; \|\varphi\|_{k,2,\Omega}.$$

The spaces $H^k(\Omega)$ are Hilbert spaces with inner product

$$\big(\varphi,\psi\big)_{k,\Omega} \;=\; \sum_{|\alpha|\le k} \int_\Omega D^\alpha\varphi\, D^\alpha\psi\, dx.$$

For $k=0$ the index k for the inner product and the norm may be dropped.

Up to now, we have defined the Sobolev spaces for an integral order of derivatives. This can be extended to real numbers $s\ge 0$. For a non-integer number s define $k\in\mathbb{N}$ and $0<\lambda<1$ such that $s=k+\lambda$. Then the Banach space $H^s(\Omega)$ consists of all functions $\varphi\in H^k(\Omega)$ satisfying the Sobolev-Slobodeckij norm

$$\|\varphi\|^2_{s,\Omega} \;:=\; \|\varphi\|^2_{k,\Omega} + \sum_{|\alpha|=k} I_\lambda(D^\alpha\varphi, D^\alpha\varphi) < \infty,$$

where

$$I_\lambda(\varphi,\psi) \;:=\; \iint\limits_{\Omega\times\Omega} \frac{\big(\varphi(x)-\varphi(y)\big)\big(\psi(x)-\psi(y)\big)}{|x-y|^{d+2\lambda}}\,dx\,dy.$$

It is again a Hilbert space with inner product

$$\big(\varphi,\psi\big)_{s,\Omega} \;=\; \big(\varphi,\psi\big)_{k,\Omega} + \sum_{|\alpha|=k} I_\lambda(D^\alpha\varphi, D^\alpha\psi).$$

Next, we define Sobolev spaces for the boundary Γ. Since Ω is a Lipschitz domain there exist a positive integer $M\in\mathbb{N}$, open and bounded subsets $U_i\subset\mathrm{I\!R}^d$, and bijections $\phi_i : U_i \to K_1^d(0) = \big\{\xi\in\mathrm{I\!R}^d \,\big|\, |\xi|<1\big\}$, such that for $1\le i\le M$

$$\begin{aligned}
&\Gamma\subset\bigcup_{i=1}^{M} U_i, \quad \phi_i\in C^{0,1}(\overline{U}_i), \quad \phi_i^{-1}\in C^{0,1}(\overline{K_1^d(0)}),\\
&\phi_i(U_i\cap\Gamma) \;=\; \big\{\xi\in K_1^d(0)\,\big|\,\xi_d=0\big\},\\
&\phi_i(U_i\cap\Omega) \;=\; \big\{\xi\in K_1^d(0)\,\big|\,\xi_d>0\big\},\\
&\phi_i(U_i\cap(\mathrm{I\!R}^d\setminus\Omega)) \;=\; \big\{\xi\in K_1^d(0)\,\big|\,\xi_d<0\big\}.
\end{aligned}$$

Furthermore one can construct a partition of unity for Γ on $(U_i)_{i=1}^M$ (cf. for instance [55]), i.e. there exist

$$\begin{aligned}
&\sigma_i\in C^\infty(U_i), \quad \operatorname{supp}(\sigma_i)\subset U_i,\\
&\sum_{i=1}^{M}\sigma_i^2(x) \;=\; 1 \quad \forall x\in\Gamma.
\end{aligned}$$

Defining $\alpha_i : K_1^{d-1}(0)\to U_i\cap\Gamma$ as $\alpha_i(x)=\phi_i^{-1}(x,0)$, the space $H^s(\Gamma)$ consists of all measurable functions $\lambda:\Gamma\to\mathrm{I\!R}$ such that

$$(\sigma_i^2\lambda)\circ\alpha_i\in H^s(K_1^{d-1}(0)) \text{ for } 1\le i\le M.$$

It is a Hilbert space with inner product

$$(\lambda,\mu)_{s,\Gamma} \;=\; \sum_{i=1}^{M}\big((\sigma_i^2\lambda)\circ\alpha_i,(\sigma_i^2\mu)\circ\alpha_i\big)_{s,K_1^{d-1}(0)}.$$

The proof of independency of $H^s(\Gamma)$ regarding the choice of atlas $(U_i,\phi_i)_{i=1}^M$ and partition of unity $(\sigma_i)_{i=1}^M$ as well as the equivalence of the resulting norms can be found, e.g. in [55].

The connection between the Sobolev space $H^1(\Omega)$ and the boundary space $H^{\frac{1}{2}}(\Gamma)$ is established by the following trace theorem (cf. [40]).

Theorem 3.1 (Trace Theorem).
Let $\Omega \subset \mathbb{R}^d$ be a Lipschitz domain with boundary Γ. The trace operator

$$\gamma \,:\, H^1(\Omega) \to H^{\frac{1}{2}}(\Gamma) \quad \textit{with} \quad \varphi \mapsto \gamma\varphi$$

is linear, continuous, and onto. Furthermore it has a right continuous inverse.

Note, that for continuous functions $u \in C^0(\overline{\Omega})$ the trace γ is nothing else than the restriction of the function to the boundary, i.e. $\gamma u = u|_\Gamma$. Now, by means of the trace theorem, we obtain an equivalent norm on $H^{\frac{1}{2}}(\Gamma)$ better suited for later calculations. For a given function $\lambda \in H^{\frac{1}{2}}(\Gamma)$ we define

$$\|\lambda\|_{\frac{1}{2},\Gamma} \;:=\; \inf_{\substack{\varphi\in H^1(\Omega)\\ \gamma\varphi=\lambda}} \|\varphi\|_{1,\Omega}. \tag{3.1}$$

The dual space of $H^s(\Gamma)$, defined as the set of all bounded linear mappings from $H^s(\Gamma)$ to $\mathbb{R}$, is denoted by $H^{-s}(\Gamma)$, equipped with the usual operator norm

$$\|\mu\|_{-s,\Gamma} \;:=\; \sup_{\substack{\lambda\in H^s(\Gamma)\\ \lambda\neq 0}} \frac{|\mu(\lambda)|}{\|\lambda\|_{s,\Gamma}}. \tag{3.2}$$

For $\lambda \in H^s(\Gamma)$ and $\mu \in H^{-s}(\Gamma)$ we define the duality pairing as

$$\langle \mu, \lambda \rangle_{s,\Gamma} \;:=\; \mu(\lambda). \tag{3.3}$$

3.2 Existence and uniqueness of a subproblem

In this section we derive a linear subproblem of the full problem presented in Chapter 2. The spaces required for existence and uniqueness of a weak solution are used to derive a variational formulation of the complete system in Section 3.3.

The domains $\Omega, \Omega^s, \Omega^\ell \subset \mathbb{R}^d$ $(d=2,3)$ are defined as in Figure 2.1 on page 10. These domains are Lipschitz, the subdomains Ω^s and Ω^ℓ are disjoint, separated by the interface Γ. The interface between the subdomains is connected and possesses nonzero finite $(d-1)$–dimensional Hausdorff measure. Furthermore, we assume that Γ is smooth. Note that both subdomains as well as the interface are time dependent. The entire domain Ω, however, remains fixed.

3.2.1 Derivation of the subproblem

We require a subproblem which is as simple as possible while still yielding information about the appropriate space containing the interface velocity. This is realized by the following simplifications to the system as described in Chapter 2 on page 27:

- The Navier-Stokes equations (2.29) are not considered. The flow velocity $\boldsymbol{u}$ in the temperature and concentration equations (2.30) is set to zero.
- To decouple the remaining system we assume that the concentrations in equation (2.30) are known a priori.
- To obtain a stationary problem from (2.30) we consider the semi-discretization in time for any $t \in [0,T]$ with $\theta_t(t,x) \approx \big(\theta(t,x) - \theta(t-\tau,x)\big)/\tau$ and time step size τ. We assume that $\theta(t-\tau,x)$ is already known.
- The motion of the interface is neglected. As a consequence Ω^s, Ω^ℓ, and Γ are fixed, thus eliminating the Stefan condition for temperature (cf. equation (2.31)).
- The Robin boundary condition in (2.33) is replaced by a homogeneous Neumann boundary condition.

We obtain the following Helmholtz equation for temperature at time t in the solid and liquid domain

$$\begin{aligned} \theta - \tau\nabla\cdot(\hat{a}\nabla\theta) &= f && \text{in } \Omega^\ell,\ \Omega^s, \\ \theta &= g(c_1^\ell, c_2^\ell) && \text{on } \Gamma, \\ \frac{\partial\theta}{\partial\boldsymbol{\nu}} &= 0 && \text{on } \partial\Omega, \end{aligned} \tag{3.4}$$

with right hand side $f = \tau\theta(t-\tau,x)$ and Dirichlet boundary condition $g(c_1^\ell,c_2^\ell) = 1.36 - 0.56c_2^\ell + 0.65c_1^\ell - 1.03c_1^\ell/(1-c_2^\ell)$ on the inner boundary Γ (cf. equation

(2.32) on page 27). Assuming $g \in H^{\frac{1}{2}}(\Gamma)$, the variational formulation of equation (3.4) is to find a solution $\theta \in H_g = \{\varphi \in H^1(\Omega) \mid \gamma\varphi = g \text{ on } \Gamma\}$ of

$$(\theta, \varphi)_\Omega + \tau(\hat{a}\nabla\theta, \nabla\varphi)_\Omega = (f, \varphi)_\Omega \qquad \forall \varphi \in H_0. \tag{3.5}$$

Existence and uniqueness of the weak solution θ follows from standard elliptic theory (cf. for instance [21]). In order to calculate the related interface velocity v_Γ we must apply the Stefan condition for the temperature (cf. equation (2.31) on page 27)

$$\left[\hat{a}\frac{\partial\theta}{\partial\boldsymbol{\nu}}\right]_s^\ell = -v_\Gamma. \tag{3.6}$$

As the Trace Theorem 3.1 does not ensure the existence of the normal derivatives on Γ for $\theta \in H^1(\Omega)$ we require the following theorem stated in [3].

Theorem 3.2. *Let $h \in L^2(\Omega)$, $c > 0$, and u be any weak solution of*

$$u - c\Delta u = h \quad \textit{in } \Omega.$$

Then we have $\partial u/\partial\boldsymbol{\nu} \in H^{-\frac{1}{2}}(\Omega)$ and

$$\left\|\frac{\partial u}{\partial\boldsymbol{\nu}}\right\|_{-\frac{1}{2},\Gamma} \leq C(c, d, \Omega)\left(\|u\|_{1,\Omega} + \|f\|_{0,\Omega}\right).$$

Since $\theta|_{\Omega^s}, \theta|_{\Omega^\ell}$ fulfill the requirements of Theorem 3.2 in the subdomains Ω^s, Ω^ℓ we may infer the existence of normal derivatives on Γ. Equation (3.6) can now be applied to calculate the interface velocity $v_\Gamma \in H^{-\frac{1}{2}}(\Gamma)$.

3.2.2 Saddle point problem

The complete coupled system as given on page 27 must be solved in both variables θ, v_Γ simultaneously. We will state an alternative formulation of problem (3.5).

In this section the domain Ω, the right hand side f, and the boundary condition g are the same as in (3.4). Define a bilinear form a on $H^1(\Omega) \times H^1(\Omega)$ as

$$a(\varphi, \psi) = (\varphi, \psi)_\Omega + \tau(\hat{a}\nabla\varphi, \nabla\psi)_\Omega, \tag{3.7}$$

and a bilinear form b on $H^1(\Omega)\times H^{-\frac{1}{2}}(\Gamma)$ as

$$b(\varphi,\mu) \;=\; \langle \mu, \gamma\,\varphi\rangle_{\frac{1}{2},\Gamma}. \tag{3.8}$$

Since it is clear that the duality pairing $\langle\,\cdot\,,\,\cdot\,\rangle_{\frac{1}{2},\Gamma}$ always acts on Γ we omit the trace operator γ in the sequel.

Now, the alternative formulation of problem (3.5) is to find a solution $(\theta,\lambda)\in H^1(\Omega)\times H^{-\frac{1}{2}}(\Gamma)$ of the equations

$$\begin{aligned} a(\theta,\varphi)+\tau\,b(\varphi,\lambda) \;&=\; \big(f,\varphi\big)_\Omega \qquad &&\forall\varphi\in H^1(\Omega),\\ b(\theta,\mu) \;&=\; \langle\mu,g\rangle_{\frac{1}{2},\Gamma} \qquad &&\forall\mu\in H^{-\frac{1}{2}}(\Gamma). \end{aligned} \tag{3.9}$$

This is an example of a saddle point problem as defined for instance in [11]): Considering the function

$$\mathcal{L}(\varphi,\mu) \;:=\; \tfrac{1}{2}a(\varphi,\varphi)-\big(f,\varphi\big)_\Omega \;+\;\tau\,\big[\,b(\varphi,\mu)-\langle\mu,g\rangle_{\frac{1}{2},\Gamma}\big],$$

we observe that the solution (θ,λ) of equation (3.9) is a saddle point of $\mathcal{L}$, i.e.

$$\mathcal{L}(\theta,\mu)\;\le\;\mathcal{L}(\theta,\lambda)\;\le\;\mathcal{L}(\varphi,\lambda) \qquad \forall(\varphi,\mu)\in H^1(\Omega)\times H^{-\frac{1}{2}}(\Gamma).$$

This new formulation has the advantage of solving simultaneously for the temperature θ and the constraint λ. The latter turns out to be related to the interface velocity v_Γ:

Lemma 3.3. *The problems* (3.5) *and* (3.9) *are equivalent with* $\lambda\!=\!-v_\Gamma$.

Proof. For problem (3.5) we have, as already mentioned in the last section, existence and uniqueness of a weak solution $\theta\in H^1(\Omega)$. By application of equation (3.6) we obtain the related interface velocity $v_\Gamma\in H^{-\frac{1}{2}}(\Gamma)$. We must now demonstrate that $(\theta,-v_\Gamma)\!\in\! H^1(\Omega)\!\times\! H^{-\frac{1}{2}}(\Gamma)$ is a solution of equation (3.9) and that this solution is unique.

First, we note that $\theta^s=\theta|_{\Omega^s}$ is the (unique) weak solution of the Neumann problem

$$\begin{aligned} u-\tau\nabla\cdot(\hat{a}\nabla u) \;&=\; f \qquad &&\text{in } \Omega^s,\\ \frac{\partial u}{\partial\boldsymbol{\nu}^s} \;&=\; \frac{\partial\theta^s}{\partial\boldsymbol{\nu}^s} \qquad &&\text{on } \Gamma. \end{aligned}$$

Since $C_0^\infty(\mathbb{R}^d)$ is dense in $H^1(\Omega^s)$ (cf. for instance [24]) we get, using an approximation argument, that for any $\varphi \in H^1(\Omega^s)$

$$(3.10)\qquad \big(\theta^s,\varphi\big)_{\Omega^s} + \tau\big(\hat a\nabla\theta^s,\nabla\varphi\big)_{\Omega^s} \;=\; \big(f,\varphi\big)_{\Omega^s} + \tau\big\langle \hat a\frac{\partial\theta^s}{\partial\boldsymbol{\nu}^s},\varphi\big\rangle_{\frac12,\Gamma}.$$

Analogously, on the liquid domain we obtain for any $\varphi \in H^1(\Omega^\ell)$

$$(3.11)\qquad \big(\theta^\ell,\varphi\big)_{\Omega^\ell} + \tau\big(\hat a\nabla\theta^\ell,\nabla\varphi\big)_{\Omega^\ell} \;=\; \big(f,\varphi\big)_{\Omega^\ell} + \tau\big\langle \hat a\frac{\partial\theta^\ell}{\partial\boldsymbol{\nu}^\ell},\varphi\big\rangle_{\frac12,\Gamma}.$$

Combining (3.10) and (3.11) yields for any $\varphi \in H^1(\Omega)$

$$\begin{aligned}
\big(\theta,\varphi\big)_{\Omega} + \tau\big(\hat a\nabla\theta,\nabla\varphi\big)_{\Omega} \;&=\; \big(f,\varphi\big)_{\Omega} - \tau\Big\langle \Big[\hat a\frac{\partial\theta}{\partial\boldsymbol{\nu}^s}\Big]_s^\ell,\varphi\Big\rangle_{\frac12,\Gamma}\\
&=\; \big(f,\varphi\big)_{\Omega} + \tau\big\langle v_\Gamma,\varphi\big\rangle_{\frac12,\Gamma}\\
&=\; \big(f,\varphi\big)_{\Omega} - \tau\big\langle \lambda,\varphi\big\rangle_{\frac12,\Gamma},
\end{aligned}$$

and thus the first equation of problem (3.9). Since $\gamma\,\theta = g$, the second equation of problem (3.9) is also satisfied.

To show uniqueness of the solution of problem (3.9) we consider the difference (w,ϑ) of two solutions and observe that it solves the homogeneous problem, i.e.

$$(3.12)\qquad a(w,\varphi) + \tau\, b(\varphi,\vartheta) \;=\; 0 \qquad \forall\varphi \in H^1(\Omega),$$

$$(3.13)\qquad b(w,\mu) \;=\; 0 \qquad \forall\mu \in H^{-\frac12}(\Gamma).$$

Testing equation (3.12) with $\varphi = w$ and using equation (3.13) yields

$$a(w,w) \;=\; -\tau\, b(w,\vartheta) \;=\; 0$$

and thus $w = 0$. Looking again at equation (3.12) and using $a(w,\varphi) = 0$ yields $\vartheta = 0$. ∎

By interpreting problem (3.9) as a saddle point problem, existence and uniqueness may be shown by means of the abstract saddle point theory developed since Brezzi's famous paper [10]. The starting point of the investigation in this field was a work of Babuška [3]. It dealt with the usage of Lagrange multipliers for the weak formulation of Dirichlet boundary conditions for the Poisson problem. A good summary of the results may be found in the book of Brezzi and Fortin [11]. Applied to problem (3.9) the saddle point theory yields (cf. for instance [11]):

Theorem 3.4. *Let the bilinear form a as defined in equation (3.7) be invertible on the kernel of the trace operator γ, or equivalently, that a constant $\alpha_0 > 0$ exists such that*

$$\inf_{\substack{\varphi\in\ker\gamma\\ \varphi\neq 0}} \sup_{\substack{\psi\in\ker\gamma\\ \psi\neq 0}} \frac{a(\varphi,\psi)}{\|\varphi\|_{1,\Omega}\,\|\psi\|_{1,\Omega}} \geq \alpha_0. \tag{3.14}$$

Additionally, the bilinear form b as defined in equation (3.8) fulfills the LBB-condition, meaning that there exists a constant $k_0 > 0$ such that

$$\sup_{\substack{\varphi\in H^1(\Omega)\\ \varphi\neq 0}} \frac{b(\varphi,\mu)}{\|\varphi\|_{1,\Omega}} \geq k_0\,\|\mu\|_{-\frac{1}{2},\Gamma}. \tag{3.15}$$

Then, there exists a unique solution $(\theta,\lambda)\in H^1(\Omega)\times H^{-\frac{1}{2}}(\Gamma)$ of problem (3.9) for any given $f\in L^2(\Omega)$ and $g\in H^{\frac{1}{2}}(\Gamma)$.

Remark 3.5. *The LBB-condition (3.15) in Theorem 3.4, named after Ladyzhenskaya, Babuška, and Brezzi, plays an important role in the choice of a finite element spaces suited for the discretization of a saddle point problem. It must be uniformly fulfilled for the discrete spaces approximating $H^1(\Omega)$ and $H^{-\frac{1}{2}}(\Gamma)$ to obtain a stabile discretization with respect to mesh refinement (cf. Section 4.5.1 in the next chapter).*

The bilinear form $a(\,\cdot\,,\,\cdot\,)$ as defined in equation (3.7) is bounded and positive definite. Thus, equation (3.14) holds. The problem of proving the prerequisites of Theorem 3.4 is reduced to fulfilling the LBB-condition (3.15).

Lemma 3.6. *The bilinear form b as defined in equation (3.8) fulfills the LBB-condition.*

Before proving this lemma let us state another lemma which simplifies the proof decisively.

Lemma 3.7. *Let the norm on $H^{\frac{1}{2}}(\Gamma)$ be defined as in equation (3.1). Then for all $\lambda\in H^{-\frac{1}{2}}(\Gamma)$ we have:*

$$\sup_{\substack{\psi\in H^{\frac{1}{2}}(\Gamma)\\ \psi\neq 0}} \frac{\langle\lambda,\psi\rangle_{\frac{1}{2},\Gamma}}{\|\psi\|_{\frac{1}{2},\Gamma}} = \sup_{\substack{\varphi\in H^1(\Omega)\\ \varphi\neq 0}} \frac{\langle\lambda,\varphi\rangle_{\frac{1}{2},\Gamma}}{\|\varphi\|_{1,\Omega}}. \tag{3.16}$$

Furthermore, the solution $w \in H^1(\Omega)$ of the homogeneous Neumann problem

$$(w,\varphi)_{1,\Omega} \;=\; \langle\lambda,\varphi\rangle_{\frac{1}{2},\Gamma} \quad \forall\varphi \in H^1(\Omega), \tag{3.17}$$

fulfills the following equation:

$$\|w\|_{1,\Omega} \;=\; \|\lambda\|_{-\frac{1}{2},\Gamma}. \tag{3.18}$$

Proof. First we wish to show equation (3.16):
"$\geq$": For arbitrary $\psi \in H^{\frac{1}{2}}(\Gamma)$ let $\varphi \in H^1(\Omega)$ with $\gamma\,\varphi = \psi$. We get by definition of the norm on Γ that

$$\|\psi\|_{\frac{1}{2},\Gamma} \;=\; \inf_{\substack{\phi\in H^1(\Omega)\\ \gamma\,\phi=\psi}} \|\phi\|_{1,\Omega} \;\leq\; \|\varphi\|_{1,\Omega}.$$

"$\leq$": Let any $\psi \in H^{\frac{1}{2}}(\Gamma)$. Consider the convex functional $B(v) = \|v\|^2_{1,\Omega}$. If we minimize $B(v)$ under the constraint $\gamma\,v = \psi$ there exists a $\varphi \in H^1(\Omega)$ such that

$$\|\varphi\|^2_{1,\Omega} \;=\; B(\varphi) \;=\; \inf_{\substack{v\in H^1(\Omega)\\ \gamma\,v=\psi}} B(v),$$

which is the solution of problem (3.4) with $\hat{a} = \tau = 1$, $f = 0$ and $g = \psi$. Using the definition of the norm and the fact that the square root function is monotone gives that

$$\|\psi\|_{\frac{1}{2},\Gamma} \;=\; \inf_{\substack{v\in H^1(\Omega)\\ \gamma\,v=\psi}} \|v\|_{1,\Omega} \;=\; \sqrt{B(\varphi)} = \|\varphi\|_{1,\Omega}.$$

Combining the two cases yields that equation (3.16) is fulfilled.
Equation (3.18) follows by application of equation (3.16) and (3.17):

$$\|\lambda\|_{-\frac{1}{2},\Gamma} \;=\; \sup_{\substack{\varphi\in H^1(\Omega)\\ \varphi\neq 0}} \frac{\langle\lambda,\varphi\rangle_{\frac{1}{2},\Gamma}}{\|\varphi\|_{1,\Omega}} \;=\; \sup_{\substack{\varphi\in H^1(\Omega)\\ \varphi\neq 0}} \frac{(w,\varphi)_{1,\Omega}}{\|\varphi\|_{1,\Omega}} \;=\; \|w\|_{1,\Omega}.$$

∎

The proof of Lemma 3.6 is now a simple application of Lemma 3.7.

Proof of Lemma 3.6.
Let any $\lambda \in H^{-\frac{1}{2}}(\Gamma)$. Choosing w as the solution of equation (3.17) in Lemma 3.7 we obtain

$$\sup_{\substack{\varphi \in H^1(\Omega) \\ \varphi \neq 0}} \frac{\langle \lambda, \varphi \rangle_{\frac{1}{2},\Gamma}}{\|\lambda\|_{-\frac{1}{2},\Gamma} \|\varphi\|_{1,\Omega}} \geq \frac{\langle \lambda, w \rangle_{\frac{1}{2},\Gamma}}{\|\lambda\|_{-\frac{1}{2},\Gamma} \|w\|_{1,\Omega}} \overset{(3.17)}{=} \frac{\|w\|^2_{1,\Omega}}{\|\lambda\|_{-\frac{1}{2},\Gamma} \|w\|_{1,\Omega}} \overset{(3.18)}{=} 1.$$

∎

3.3 Variational formulation of the system

The main focus of our work lies on the treatment of the growth process of the crystal layer on top of the substrate described by the Stefan conditions and the equations of the phase diagram. Thus, we assume for the derivation of the variational formulation of the system that the flow velocity $\boldsymbol{u}$ is already given. Hence, the Navier-Stokes equations (2.7) are not required for the solution of the remaining problem and are not treated in this section.

For the mathematical model we use the fact that the melting pot as shown in Figure 2.1 on page 10 possesses a plane of symmetry. Because of this it is enough to consider only one half of it. A sketch of the new geometry with corresponding boundaries is shown in Figure 3.1. By the bisection of the melting pot we obtain a new boundary segment, the symmetry boundary Γ_i. To complete the system we have to choose boundary conditions on Γ_i for the temperature θ and the concentrations $c_{1,2}$. Assuming that they are smooth near Γ_i and show the same symmetry as the domain, their normal derivatives on Γ^i must vanish. The boundary conditions on Γ^i are therefore homogeneous Neumann boundary conditions, i.e.

$$\frac{\partial \theta}{\partial \boldsymbol{\nu}} = \frac{\partial c_1}{\partial \boldsymbol{\nu}} = \frac{\partial c_2}{\partial \boldsymbol{\nu}} = 0 \quad \text{on } \Gamma^i. \tag{3.19}$$

For the flow velocity no slip boundary conditions are assumed. This is reasonable since the substrate holder, which is not included in our geometry, separates the melting pot.

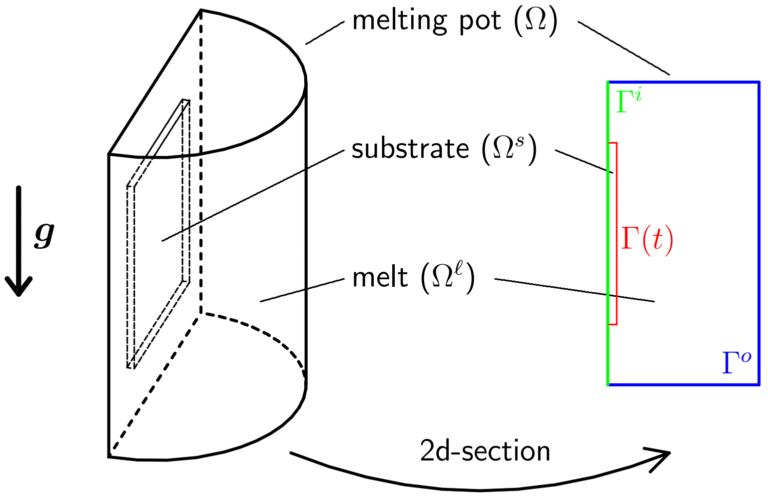


Figure 3.1: 3- and 2-dimensional geometry for the simulations with description of the different boundary types.

3.3.1 Functional framework

From the previous chapter we already know that the concentrations may jump over the interface Γ. This will need to be incorporated in an appropriate solution space. For the sake of simplicity let us first introduce the following notation for the union of the two open subdomains Ω^s and Ω^ℓ. For fixed $t > 0$ let

$$\Omega^\bullet(t) := \Omega^s(t) \cup \Omega^\ell(t) = \Omega \setminus \Gamma(t). \tag{3.20}$$

Dropping the dependency on t the space of concentrations can be defined as

$$\mathcal{H}^{1,p}(\Omega^\bullet) := \left\{ \varphi \in L^p(\Omega) \mid \varphi|_{\Omega^s} \in H^{1,p}(\Omega^s) \wedge \varphi|_{\Omega^\ell} \in H^{1,p}(\Omega^\ell) \right\}. \tag{3.21}$$

Together with the $H^{1,p}$-broken norm

$$\|\varphi\|_{1,p,\Omega^\bullet} = \left(\|\varphi^s\|^2_{1,p,\Omega^s} + \|\varphi^\ell\|^2_{1,p,\Omega^\ell}\right)^{\frac{1}{2}},$$

where $\varphi^s = \varphi|_{\Omega^s}$ and $\varphi^\ell = \varphi|_{\Omega^\ell}$, it is a Banach space, because $H^{1,p}(\Omega^s)$ and $H^{1,p}(\Omega^\ell)$ are Banach spaces themselves and Γ has a vanishing Lebesgue measure. Again as for the definition of the usual Sobolev norm we drop the index p in the special case of $p=2$. Keep in mind that while Γ, $\Omega^\bullet$, and $\mathcal{H}^{k,p}(\Omega^\bullet)$ are time-dependent, the domain Ω is always fixed.

Since temperature, concentrations, and interface velocity are time dependent we introduce the space of L^p-functions with values in a Banach space (cf. for instance [55]). Let X be any Banach space with the norm $\|\cdot\|_X$ and let $1 \leq p \leq \infty$. Then $L^p(0,T;X)$ denotes the space of all functions $\varphi(t,x)$ satisfying $\varphi(t) \in X$ for all $t \in [0,T]$ and $\|\varphi(t)\|_X \in L^p(0,T)$. It is again a Banach space equipped with the norm

$$\|\varphi\|_{L^p(0,T;X)} \;=\; \left(\int_0^T \|\varphi(t)\|_X^p \, dt\right)^{1/p} \tag{3.22}$$

if $p < \infty$, and

$$\|\varphi\|_{L^\infty(0,T;X)} \;=\; \operatorname*{ess\,sup}_{t\in[0,T]} \|\varphi(t)\|_X \tag{3.23}$$

if $p = \infty$. Concerning the derivatives of functions with values in Banach spaces we cite a lemma which for instance can be found in [51].

Lemma 3.8. *Let X be a given Banach space with dual X' and let u and g be two functions belonging to $L^1(0,T;X)$. Then the following two conditions are equivalent:*

(i) *u is a.e. equal to a primitive function of g, i.e.*

$$u(t) \;=\; \zeta + \int_0^t g(s)\, ds, \quad \zeta \in X, \text{ a.e. } t \in [0,T].$$

(ii) *For each test function $\phi \in C_0^\infty((0,T))$,*

$$\int_0^T u(t)\frac{d\phi}{dt}\, dt \;=\; -\int_0^T g(t)\phi(t)\, dt.$$

Moreover, if (i) or (ii) are satisfied, u is almost everywhere equal to a continuous function from $[0,T]$ into X.

In principle this lemma is the equivalent to Sobolev's embedding theorem for real valued functions in one space dimension. It legitimates the specification of pointwise initial conditions in time for the temperature and the concentrations in the fixed interface approach (cf. Problem 3.11).

3.3.2 Moving interface

In this section we derive a variational formulation of the physical problem as defined in Section 2.7 on page 27, as already mentioned without the Navier-Stokes equations (2.7). Therefore we require space-time-dependent test functions which vanish for time T and define

$$X^1(T,\Omega) \;=\; \big\{\varphi \in H^1((0,T)\times\Omega) \,\big|\, \gamma\,\varphi = 0 \text{ on } \{T\}\times\Omega\big\}. \tag{3.24}$$

To obtain the variational formulation we use the standard technique and multiply for fixed $t \in [0,T]$ the partial differential equations (2.30) with a space-time-dependent test function $\varphi \in X^1(T,\Omega)$. Then, we integrate over the subdomains Ω^s, Ω^ℓ. For the temperature equation this yields

$$\begin{aligned}
\big(\theta_t,\varphi\big)_{\Omega^s(t)} + \big(\boldsymbol{u}\cdot\nabla\theta,\varphi\big)_{\Omega^s(t)} - \big(\nabla\cdot(\hat{a}\nabla\theta),\varphi\big)_{\Omega^s(t)} &= 0,\\
\big(\theta_t,\varphi\big)_{\Omega^\ell(t)} + \big(\boldsymbol{u}\cdot\nabla\theta,\varphi\big)_{\Omega^\ell(t)} - \big(\nabla\cdot(\hat{a}\nabla\theta),\varphi\big)_{\Omega^\ell(t)} &= 0.
\end{aligned}$$

The two equations in the two subdomains are then summed:

$$\big(\theta_t,\varphi\big)_{\Omega^\bullet(t)} + \big(\boldsymbol{u}\cdot\nabla\theta,\varphi\big)_{\Omega^\bullet(t)} - \big(\nabla\cdot(\hat{a}\nabla\theta),\varphi\big)_{\Omega^\bullet(t)} \;=\; 0. \tag{3.25}$$

Assuming that the temperature is smooth in the subdomains Ω^s, Ω^ℓ and overall continuous, the second order term can be integrated by parts. Thereby we get boundary integrals and use the Stefan conditions (2.31) on Γ, the outer boundary conditions (2.33) on Γ^o, and the boundary conditions (3.19) for the symmetry boundary Γ^i to substitute them:

$$\begin{aligned}
-\big(\nabla\cdot(\hat{a}\nabla\theta),\varphi\big)_{\Omega^\bullet(t)} &= \big(\hat{a}\nabla\theta,\nabla\varphi\big)_{\Omega^\bullet(t)} - \big(\hat{a}_s\tfrac{\partial\theta}{\partial\boldsymbol{\nu}_s},\varphi\big)_{\partial\Omega^s(t)} - \big(\hat{a}_\ell\tfrac{\partial\theta}{\partial\boldsymbol{\nu}_\ell},\varphi\big)_{\partial\Omega^\ell(t)}\\
&= \big(\hat{a}\nabla\theta,\nabla\varphi\big)_{\Omega^\bullet(t)} - \big(\hat{a}_\ell\tfrac{\partial\theta}{\partial\boldsymbol{\nu}_\ell},\varphi\big)_{\Gamma^o(t)} + \big(\big[\hat{a}\tfrac{\partial\theta}{\partial\boldsymbol{\nu}_s}\big]_s^\ell,\varphi\big)_{\Gamma(t)} \qquad (3.26)\\
&= \big(\hat{a}\nabla\theta,\nabla\varphi\big)_{\Omega^\bullet(t)} - \big(\hat{R}(\hat{\theta}_{ext}-\theta),\varphi\big)_{\Gamma^o(t)} - \big(v_\Gamma,\varphi\big)_{\Gamma(t)},
\end{aligned}$$

where $\boldsymbol{\nu}_s, \boldsymbol{\nu}_\ell$ are the outer unit normals of Ω^s, Ω^ℓ. Recall that for a weak solution $\theta(t,\cdot) \in H^1(\Omega)$ the interface velocity v_Γ is only in $H^{-\frac{1}{2}}(\Gamma)$ (cf. Section 3.2.2). Thus we have to replace the boundary integral $(v_\Gamma,\varphi)_\Gamma$ by the duality pairing $\langle v_\Gamma,\varphi\rangle_{\frac{1}{2},\Gamma}$.

Next, we integrate equation (3.25) in time over the time interval $[0, T]$. We wish to integrate the first term with time derivative by parts. Note that

$$\begin{aligned}\frac{d}{dt}\big(\theta,\varphi\big)_{\Omega^s(t)} &= \big(\theta_t,\varphi\big)_{\Omega^s(t)} + \big(\theta,\varphi_t\big)_{\Omega^s(t)} + \big(\theta\varphi, v_\Gamma\big)_{\Gamma(t)},\\ \frac{d}{dt}\big(\theta,\varphi\big)_{\Omega^\ell(t)} &= \big(\theta_t,\varphi\big)_{\Omega^\ell(t)} + \big(\theta,\varphi_t\big)_{\Omega^\ell(t)} - \big(\theta\varphi, v_\Gamma\big)_{\Gamma(t)},\end{aligned}$$

where v_Γ denotes the normal velocity of Γ with direction towards $\Omega^\ell(t)$. Since the temperature θ is continuous on Γ, we obtain that

$$\int_0^T \big(\theta_t,\varphi\big)_{\Omega^\bullet(t)}\,dt \;=\; -\int_0^T \big(\theta,\varphi_t\big)_{\Omega^\bullet(t)}\,dt - \big(\theta(0),\varphi(0)\big)_{\Omega^\bullet(0)}. \tag{3.27}$$

In the last term of equation (3.27), $\theta(0)$ is replaced by the initial value $\theta_0 = 0$. Collecting all parts gives the variational formulation of the temperature equation

$$\begin{aligned}-\int_0^T \big(\theta,\varphi_t\big)_\Omega\,dt + \int_0^T \big(\boldsymbol{u}\cdot\nabla\theta,\varphi\big)_\Omega\,dt + \int_0^T \big(\hat{a}\nabla\theta,\nabla\varphi\big)_\Omega\,dt\\ + \int_0^T \hat{R}\big((\theta-\theta_{ext}),\varphi\big)_{\Gamma^o}\,dt - \int_0^T \langle v_\Gamma,\varphi\rangle_{\frac{1}{2},\Gamma}\,dt \;=\; 0,\end{aligned}$$

where the time argument for the spatial integration domains is dropped and $\Omega^\bullet$ is replaced by Ω, since this does not affect the value of the integrals.

For the concentration equations we proceed in the same manner. Here we only present the two parts which differ:
First, integration by parts of the second order term, the use of the Stefan condition (2.31), and the use of the boundary conditions (2.33) on Γ^o and (3.19) on Γ^i yields

$$-\big(\nabla\cdot(\hat{D}_i\nabla c_i),\varphi\big)_{\Omega^\bullet(t)} \;=\; \big(\hat{D}_i\nabla c_i,\nabla\varphi\big)_{\Omega^\bullet(t)} - \big([c_i]_s^\ell v_\Gamma,\varphi\big)_{\Gamma(t)}. \tag{3.28}$$

Second, the concentrations are discontinuous on Γ. Hence integration by parts of the term with time derivative yields

$$\begin{aligned}\int_0^T \big(c_t,\varphi\big)_{\Omega^\bullet(t)}\,dt \;=\; -\int_0^T \big(c,\varphi_t\big)_{\Omega^\bullet(t)}\,dt + \int_0^T \big([c]_s^l\,\varphi, v_\Gamma\big)_{\Gamma(t)}\,dt\\ - \big(c(0),\varphi(0)\big)_{\Omega^\bullet(0)}.\end{aligned} \tag{3.29}$$

In contrast to equation (3.27) an additional nonlinear term occurs which is identical with the one arising from the partial integration of the second order term in equation (3.28) – but with a different sign. Thus the nonlinear term in the concentration equation cancels out.

The equations of the phase diagram (2.32) which are valid on the interface Γ are also reformulated in weak sense. Since the third equation containing the temperature is necessary for the determination of the interface velocity v_Γ we have to take $\mu \in H^{-\frac{1}{2}}(\Gamma)$ as test function. Hence, we have to replace the boundary integral by the duality pairing. On the other hand the equations for the liquid-solid coupling (the first and second equation in (2.32)) are used for the determination of the boundary values of the concentrations and thus are multiplied by trace functions $\psi \in H^{\frac{1}{2}}(\Gamma)$ of $H^1(\Omega)$.

Problem 3.9 (Moving interface approach).
For a given velocity field $\boldsymbol{u} \in L^\infty((L^\infty(\Omega))^d)$ find the interface $\Gamma(t)$ and almost everywhere in $(0,T)$ a temperature field $\theta(t) \in H^1(\Omega)$, concentration fields $c_i(t) \in \mathcal{H}^1(\Omega^\bullet(t))$ $(i=1,2)$, and the interface velocity $v_\Gamma \in H^{-\frac{1}{2}}(\Gamma(t))$ satisfying

$$
\begin{aligned}
-\int_0^T \big(\theta, \varphi_t\big)_\Omega\, dt + \int_0^T \big(\boldsymbol{u}\cdot\nabla\theta, \varphi\big)_\Omega\, dt + \int_0^T \big(\hat{a}\nabla\theta, \nabla\varphi\big)_\Omega\, dt \\
+ \int_0^T \hat{R}\big(\theta, \varphi\big)_{\Gamma^o}\, dt - \int_0^T \big\langle v_\Gamma, \varphi\big\rangle_{\frac{1}{2},\Gamma}\, dt \\
= \int_0^T \hat{R}\big(\hat{\theta}_{ext}, \varphi\big)_{\Gamma^o}\, dt \qquad & \forall \varphi \in X^1(T,\Omega), \\
-\int_0^T \big(c_i, \varphi_t\big)_\Omega\, dt + \int_0^T \big(\boldsymbol{u}\cdot\nabla c_i, \varphi\big)_{\Omega^\bullet}\, dt + \int_0^T \big(\hat{D}_i\nabla c_i, \nabla\varphi\big)_{\Omega^\bullet}\, dt \\
= \big(c_{i,0}, \varphi(0)\big)_\Omega \quad (i=1,2) \qquad & \forall \varphi \in X^1(T,\Omega),
\end{aligned}
$$

and

$$
\begin{aligned}
\big(c_1^s, \psi\big)_\Gamma &= \Big(\tfrac{0.11c_1^\ell}{1-0.78c_1^\ell - c_2^\ell}, \psi\Big)_\Gamma && \forall \psi \in H^{\frac{1}{2}}(\Gamma), \\
\big(c_2^s, \psi\big)_\Gamma &= \big(0.5, \psi\big)_\Gamma && \forall \psi \in H^{\frac{1}{2}}(\Gamma), \\
\big\langle \mu, \theta\big\rangle_{\frac{1}{2},\Gamma} &= \Big\langle \mu, 1.36 - 0.56c_2^\ell + 0.65c_1^\ell - \tfrac{1.03c_1^\ell}{1-c_2^\ell}\Big\rangle_{\frac{1}{2},\Gamma} && \forall \mu \in H^{-\frac{1}{2}}(\Gamma),
\end{aligned}
$$

a.e. in $(0,T)$.

Remark 3.10. *Assuming that the velocity field* $\boldsymbol{u}$ *is divergence free and has no slip boundary conditions on* Ω^ℓ *the concentration equation in Problem 3.9 yields mass conservations for the solution* c_i *(under appropriate regularity assumptions), i.e.*

$$\int\limits_\Omega c_i(T,x)\,dx \;=\; \int\limits_\Omega c_{i,0}(x)\,dx.$$

3.3.3 Fixed interface

The derivation of the variational formulation with fixed interface Γ is similar to the derivation in the last section. The interface is fixed at its initial position, i.e. $\Gamma = \Gamma(0)$. Hence, the domains $\Omega^s = \Omega^s(0)$ and $\Omega^\ell = \Omega^\ell(0)$ are fixed as well. Performing the same steps as in the last section we obtain equation (3.25) and (3.26) on the fixed domains. In contrast to the last section we do not integrate in time. Thus, the variational formulation of our system of partial differential equations as given in Section 2.7 on page 27 with fixed interface reads as follows:

Problem 3.11 (Fixed interface approach).
Let $\Gamma = \Gamma(0)$ *be fixed. For a given velocity field* $\boldsymbol{u} \in L^\infty(0,T;(L^\infty(\Omega))^d)$ *find a temperature field* $\theta \in L^2(0,T;H^1(\Omega))$*, concentration fields* $c_1, c_2 \in L^2(0,T;\mathcal{H}^{1,\infty}(\Omega^\bullet))$*, and the interface velocity* $v_\Gamma \in L^2(0,T;H^{-\frac{1}{2}}(\Gamma))$ *satisfying*

$$\begin{aligned}
&\big(\theta_t,\varphi\big)_\Omega + \big(\boldsymbol{u}\cdot\nabla\theta,\varphi\big)_\Omega + \big(\hat{a}\nabla\theta,\nabla\varphi\big)_\Omega \\
&\qquad + \hat{R}\big(\theta,\varphi\big)_{\Gamma^o} - \big\langle v_\Gamma,\varphi\big\rangle_{\frac{1}{2},\Gamma} = \hat{R}\big(\hat{\theta}_{ext},\varphi\big)_{\Gamma^o} \qquad \forall\varphi\in H^1(\Omega),\\[1ex]
&\big(c_{i,t},\varphi\big)_{\Omega^\bullet} + \big(\boldsymbol{u}\cdot\nabla c_i,\varphi\big)_{\Omega^\bullet} + \big(\hat{D}_i\nabla c_i,\nabla\varphi\big)_{\Omega^\bullet} \\
&\qquad -\big\langle v_\Gamma\,[c_i]_s^\ell,\varphi\big\rangle_{\frac{1}{2},\Gamma} = 0 \quad (i=1,2) \qquad \forall\varphi\in H^1(\Omega),
\end{aligned}$$

and

$$\begin{aligned}
\big(c_1^s,\psi\big)_\Gamma &= \big(\tfrac{0.11c_1^\ell}{1-0.78c_1^\ell-c_2^\ell},\psi\big)_\Gamma && \forall\psi\in H^{\frac{1}{2}}(\Gamma),\\
\big(c_2^s,\psi\big)_\Gamma &= \big(0.5,\psi\big)_\Gamma && \forall\psi\in H^{\frac{1}{2}}(\Gamma),\\
\big\langle\mu,\theta\big\rangle_{\frac{1}{2},\Gamma} &= \big\langle\mu, 1.36-0.56c_2^\ell+0.65c_1^\ell-\tfrac{1.03c_1^\ell}{1-c_2^\ell}\big\rangle_{\frac{1}{2},\Gamma} && \forall\mu\in H^{-\frac{1}{2}}(\Gamma),
\end{aligned}$$

a.e. in $(0,T)$*, with initial values*

$$\begin{aligned} \theta(0,\,\cdot\,) &= 0 && \textit{a.e. in } \Omega, \\ c_i(0,\,\cdot\,) &= c_{i,0}(\,\cdot\,) \quad (i=1,2) && \textit{a.e. in } \Omega. \end{aligned}$$

The thickness of the interface can then be calculated as

$$\int_0^T v_\Gamma(t)\,dt.$$

Remark 3.12. *Assuming that the velocity field* $\boldsymbol{u}$ *is divergence free and has no slip boundary conditions on* Ω^ℓ *the concentration equation in Problem 3.11 generally yields that*

$$0 \neq \int\limits_{\Omega^\bullet} c_{i,t}\,dx = \frac{d}{dt}\int\limits_{\Omega^\bullet} c_i\,dx.$$

Thus, mass conservation is violated.

Remark 3.13. *Using the prerequisites and the partial differential equations in Problem 3.11 we can see that for the solution* θ *and* c_1, c_2 *Lemma 3.8 can be applied. Hence there exist continuous representatives in time and the initial conditions as given in Problem 3.11 make sense.*

Chapter 4

Discretization

In this chapter the discretization of the systems in Problem 3.9 and 3.11 is presented. Since the basis for all spatial discretizations are Lagrange finite elements, we first of all supply the definitions and some extensions required for the discretization of our problem.

The Navier-Stokes equations (2.7) in both approaches (cf. Algorithm 4.4 and 4.6) are decoupled in time, i.e. we take the flow velocity of the last time step for the calculation of the convective part in the temperature and the concentration equations. Since their treatment is not in our main focus we only give a short introduction with Taylor-Hood elements in space and a fractional theta scheme in time and refer to the literature for further details.

For the approach in Problem 3.9 which comprises the motion of Γ we derive a space-time finite element method. The approach in Problem 3.11 neglects the geometrical movement of the free boundary Γ. For it we derive a finite element discretization in space and use a backward Euler scheme in time. Afterwards some aspects regarding stability and numerical oscillations of the discretizations are addressed.

4.1 Finite element spaces

Let $a_0, a_1, \dots, a_d \in \mathbb{R}^d$ $(d = 2, 3)$ such, that the vectors $v_i = a_i - a_0$ for $i = 1, \dots, d$ are linearly independent. Then the convex hull

$$T \;=\; \mathrm{co}(a_0, \dots, a_d) \;=\; \left\{ x = \sum_{i=0}^{d} \lambda_i a_i \mid 0 \le \lambda_i \le 1,\; \sum_{i=0}^{d} \lambda_i = 1 \right\}$$

is called a non-degenerate d-simplex. The convex hull of every subset $S \subsetneq \{a_0, \dots, a_d\} \neq \emptyset$ is a sub-simplex of T. If $e_1, \dots, e_d$ are the canonical basis vectors of $\mathbb{R}^d$, $\hat{T} = \mathrm{co}(0, e_1, \dots, e_d)$ is called the unit simplex. 2-simplices are triangles and 3-simplices are tetrahedra. The diameter of a simplex is defined as

$$h_T \;=\; \mathrm{diam}(T) \;=\; \max\{|x - y| \mid x, y \in T\},$$

the maximum radius of all balls lying inside the simplex as

$$\rho_T \;=\; \sup\{r \in \mathbb{R} \mid B_r(x) \subset T,\; x \in T\}.$$

A triangulation $\mathcal{T}_h = \{T_i \text{ non degenerated simplex } \mid i = 1, \dots, I\}$ of a polygonal or polyhedral domain $\Omega_h \subset \mathbb{R}^d$ with $\overline{\Omega}_h = \bigcup_{T \in \mathcal{T}_h} T$ is called conforming if the intersection of two different elements T_1, T_2 is either disjoint or a sub-simplex of both elements. The index h is called mesh size and is defined as

$$h = \max_{T \in \mathcal{T}_h} h_T.$$

A sequence $(\mathcal{T}_{h_j})_{j \in \mathbb{N}}$ of triangulations is regular if there exists a positive constant c such that for all $T \in \mathcal{T}_{h_j}$

$$\rho_T \ge c h_T.$$

Denoting by $P_k(T)$ the space of all polynomials up to degree k on T the space of the Lagrange finite elements of k-th order is defined as

$$X_h^k(\Omega_h) \;=\; \left\{ \varphi_h \in C^0(\Omega_h) \,\middle|\, \varphi_h|_T \in P_k(T) \quad \forall T \in \mathcal{T}_h \right\}. \tag{4.1}$$

Note, that since the functions of this finite dimensional space are locally smooth and overall continuous, it is a subset of $H^1(\Omega_h)$.

For the pressure p in the Navier-Stokes equations (2.7) a similar space is required. Since p is only unique up to a constant we demand that the mean value of function in the space vanishes, i.e.

$$P_h^k(\Omega_h) \;=\; \Big\{ \varphi_h \in X_h^k(\Omega_h) \;\Big|\; \int_{\Omega_h} \varphi_h \, dx \;=\; 0 \Big\}. \tag{4.2}$$

For the discretization of the flow velocity $\boldsymbol{u}$ in the Navier-Stokes equations (2.7) the vector variant of $X_h^k(\Omega_h)$ is required, i.e.

$$V_h^k(\Omega_h) \;=\; \Big\{ \varphi_h \in \big(C^0(\Omega_h)\big)^d \;\Big|\; \varphi_h|_T \in \big(P^k(T)\big)^d \quad \forall T \in \mathcal{T}_h \Big\}, \tag{4.3}$$

$$\mathring{V}_h^k(\Omega_h) \;=\; \Big\{ \varphi_h \in V_h^k(\Omega_h) \;\Big|\; \varphi_h|_{\partial\Omega_h} = 0 \Big\}. \tag{4.4}$$

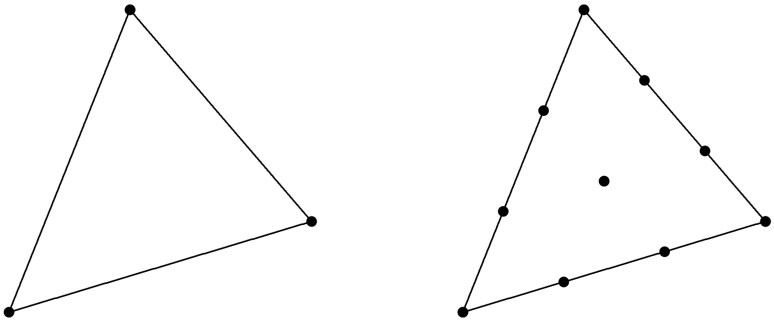


Figure 4.1: Local Lagrange grid points $L_1(T)$ (left) and $L_3(T)$ (right) for an arbitrary triangle T.

Our next concern is to construct a basis for $X_h^k(\Omega_h)$. For this we define the set of end points of a partition of $[0,1]$ into k equal parts:

$$S_k = \left\{ 0, \frac{1}{k}, \dots, \frac{k-1}{k}, 1 \right\}.$$

If we consider for an element T having vertices $a_0, \dots, a_d$ the local Lagrange grid points (cf. Figure 4.1)

$$L_k(T) \;=\; \Big\{ x \in T \;\Big|\; x = \sum_{i=0}^{d} \lambda_i a_i,\; \lambda_i \in S_k,\; \sum_{i=0}^{d} \lambda_i = 1 \Big\},$$

we observe that a function $\varphi_h \in X_h^k(\Omega_h)$ is uniquely determined on T by its values in all $x \in L_k(T)$ and on the whole domain by its values in the union of all local Lagrange grid points

$$L_k(\mathcal{T}_h) \;:=\; \bigcup_{T \in \mathcal{T}_h} L_k(T).$$

We order the Lagrange grid points and denote them by x_i, where $i = 1, \ldots, N_h^k$ and $N_h^k = |L_k(\mathcal{T}_h)|$. Now the functions $\varphi_i \in X_h^k(\Omega_h)$ defined as

$$\varphi_i(x_j) \;:=\; \delta_{ij}, \quad i, j = 1, \ldots, N_h^k.$$

form a basis of $X_h^k(\Omega_h)$. Therefore, the dimension of $X_h^k(\Omega_h)$ is N_h^k.

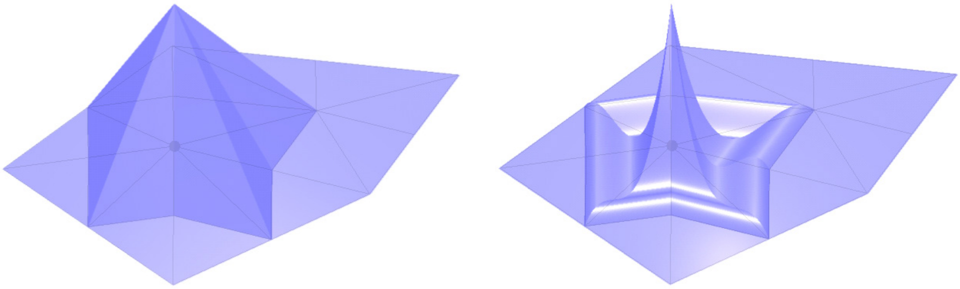


Figure 4.2: Basis function for elements of first (left) and third (right) order with underlying grid for the marked node in two dimensions

For continuous functions $v \in C^0(\overline{\Omega}_h)$ we require an interpolation operator $I_h : C^0(\overline{\Omega}_h) \to X_h^k(\Omega_h)$. Therefore we define $I_h v$ as the element of $X_h^k(\Omega_h)$ that coincides with v in all Lagrange grid points and thus can be written as

$$I_h v(x) \;=\; \sum_{i=1}^{N_h^k} v(x_i)\varphi_i(x).$$

As we have seen in Section 2.5.2 the concentrations c_i may have a jump on the interface Γ. To define an appropriate solution space we now extend $X_h^k(\Omega_h)$ by additional basis functions. For this we require a regular triangulation of the whole domain Ω_h such that the discrete interface Γ_h is represented by faces in

3d or edges in 2d of the elements. An example of such a triangulation of the domain Ω_h is shown in Figure 4.3. Note, that these sub-simplices in turn form a regular triangulation of Γ_h with dimension $(d-1)$. In the following we always assume such a triangulation $\mathcal{T}_h$ of the domain Ω_h.

Without loss of generality we can renumber the Lagrange grid points and denote by $x_1, \ldots, x_{j_h^k}$ $(j_h^k < N_h^k)$ the points lying on the interface Γ_h. Following the idea of [17] we consider the basis functions φ_i for $i = 1, \ldots, j_h^k$ and multiply them by the Heaviside function

$$H(x) \;=\; \begin{cases} 1, & \text{if } x \in \overline{\Omega}_h^s \\ 0, & \text{if } x \in \overline{\Omega}_h^\ell \setminus \Gamma_h. \end{cases}$$

The additional basis functions are denoted by $\varphi_{N_h^k+i} = H\varphi_i$ $(i = 1, \ldots, j_h^k)$. Figure 4.4 shows an example of this approach in one space dimension. Note, that in this situation the interface Γ_h consists only of a single point.

Enriching the finite element space $X_h^k(\Omega_h)$ with these discontinuous basis functions we obtain the discrete solution space for the concentrations as

$$\mathcal{X}_h^k(\Omega_h^\bullet) \;=\; X_h^k(\Omega_h) \cup \operatorname{span}\{\, H\varphi_i \;\mid\; i = 1, \ldots, j_h^k \,\}, \tag{4.5}$$

which is of dimension $\dim(\mathcal{X}_h^k(\Omega_h^\bullet)) = N_h^k + j_h^k$. For this finite element space the definition of an interpolant operator is more complicated. If $v : \Omega_h \to \mathbb{R}$ is a bounded function which is continuous on the subdomains Ω_h^s and Ω_h^ℓ we denote by $\tilde{v}^s$ and $\tilde{v}^\ell$ its continuous extension to $\overline{\Omega}_h^s$ and $\overline{\Omega}_h^\ell$. The interpolant can now be defined as

$$\tilde{I}_h v(x) \;=\; \begin{cases} I_h \tilde{v}^s(x), & x \in \overline{\Omega}_h^s \setminus \Gamma_h \\ I_h \tilde{v}^\ell(x), & x \in \overline{\Omega}_h^\ell. \end{cases}$$

Consider a function $\psi \in \mathcal{X}_h^k(\Omega_h^\bullet)$. We denote by $\psi^\ell : \Gamma_h \to \mathbb{R}$ its limit on Γ_h from the liquid domain Ω_h^ℓ and by $\psi^s : \Gamma_h \to \mathbb{R}$ its limit on Γ_h from the solid domain Ω_h^s. If ψ is given as

$$\psi \;=\; \sum_{i=1}^{N_h^k + j_h^k} k_i \varphi,$$

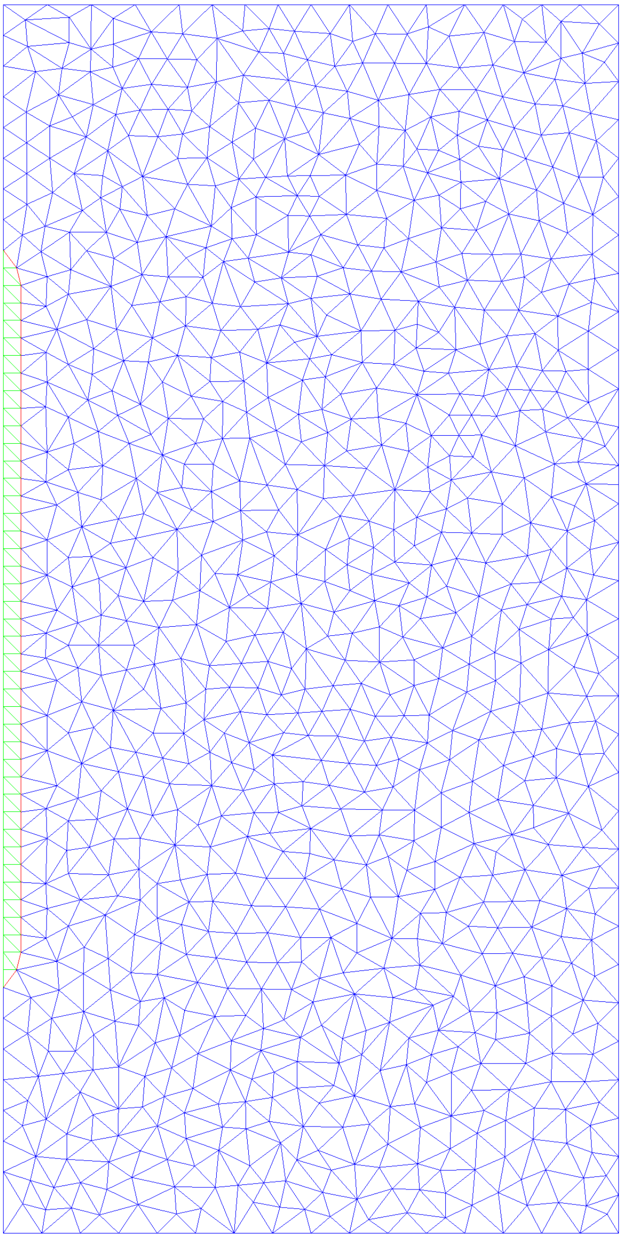


Figure 4.3: Example triangulation (2d) of the domain with solid part (green), liquid part (blue), and interface (red)

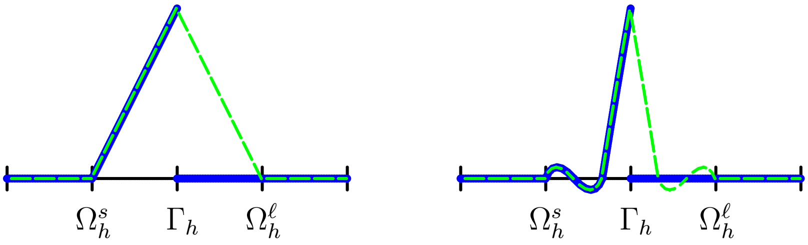


Figure 4.4: Normal basis function (dashed green) and enriched basis function (blue) for elements of first (left) and third (right) order in one space dimension

ψ^ℓ and ψ^s can be written as follows:

$$\psi^\ell \;=\; \sum_{i=1}^{j_h^k} k_i \varphi_i|_{\Gamma_h}, \tag{4.6}$$

$$\psi^s \;=\; \sum_{i=1}^{j_h^k} k_i \varphi_i|_{\Gamma_h} + \sum_{l=N_h^k+1}^{N_h^k+j_h^k} k_l \varphi_l|_{\Gamma_h}. \tag{4.7}$$

The normal interface velocity v_Γ as used in equation (2.31) is only defined on Γ. As a consequence we require a finite element space representing functions on Γ_h. As already mentioned above the triangulation $\mathcal{T}_h$ of Ω_h induces a triangulation $\mathcal{S}_h$ of Γ_h with dimension $(d-1)$. Similar to equation (4.1) we define the Lagrange finite elements of k-th order on this discrete interface as

$$M_h^k(\Gamma_h) \;=\; \Big\{ \psi_h \in C^0(\Gamma_h) \;\Big|\; \psi_h|_T \in P_k(T) \quad \forall T \in \mathcal{S}_h \Big\}. \tag{4.8}$$

Again a function $\psi_h \in M_h^k(\Gamma_h)$ is already defined by its values in the Lagrange grid points denoted by m_i, where $i = 1, \ldots, n_h^k$. Thus, the basis functions ψ_i are again characterized by

$$\psi_i(m_j) \;=\; \delta_{ij}, \quad i, j = 1, \ldots, n_h^k,$$

and the dimension of the space is $\dim(M_h^k(\Gamma_h)) = n_h^k$. Note that for the same order of Lagrange elements the boundary space $M_h^k(\Gamma_h)$ consists of the trace of the functions in $X_h^k(\Omega_h)$.

4.2 The Navier-Stokes equations

The instationary Navier-Stokes equations (2.7) form a complex dynamical system. For large values of Reynolds number one expects turbulent flow and the nonlinear term $\boldsymbol{u} \cdot \nabla \boldsymbol{u}$ becomes quite important. For such cases numerical simulations are difficult and very costly. Since our Reynolds number is relatively low ($Re = 50$) we are in a better situation. The flow induced by the temperature and concentration variations in the melting pot usually tends towards a stationary equilibrium. Nevertheless we require a discretization which is efficient and stable as well as accurate. The combination of the fractional step θ-scheme (cf. [12]) in time and the Taylor-Hood element (cf. [28]) in space is known to have these properties. In this section we give a short introduction to these techniques. Since the main focus lies on the remaining nonlinear system we will not present all details. For the theory and also some practical aspects we suggest the book of Temam [51]. For discretization aspects with finite element methods we should mention at least the book of Girault/Raviart [22].

4.2.1 Discretization in time

For the description and the properties of the discretization scheme we consider the linear scalar evolution equation

$$\begin{aligned} u_t(t) + \lambda u(t) &= 0, \\ u(0) &= u_0, \end{aligned}$$

the solution of which is given by $u(t) = e^{-\lambda t} u_0$ $(t \geq 0)$. The purpose of time discretizations with a fixed time step size τ is to calculate successively the approximations $u^n \approx u(t_n)$, where $t_n = n\tau$ $(n \in \mathbb{N})$. For one-step procedures the approximation may be written as $u^n = (\omega(\lambda\tau))^n u_0$ with a rational function $\omega : \mathbb{C} \to \mathbb{C}$. This function is called the damping factor of the scheme and it allows us to examine its basic properties (cf. for instance [16]).

For the fractional step θ-scheme used for the time discretization of the Navier-Stokes equations, we split the time interval $[t_n, t_{n+1}]$ into the three subintervals

$[t_n, t_{n+\theta\tau}]$, $[t_{n+\theta\tau}, t_{n+\theta'\tau}]$, and $[t_{n+\theta'\tau}, t_{n+1}]$, with $\theta = 1 - \sqrt{2}/2$ and $\theta' = 1 - \theta$. Furthermore we need two parameters $\alpha, \beta \in (0,1)$ where $\alpha > 0.5$ and $\beta = 1 - \alpha$.

Algorithm 4.1 (Fractional step θ-scheme).
Set $u^0 = u_0$. For $n = 0, 1, 2, \dots$ find $u^{n+\theta}$, $u^{n+\theta'}$, and u^{n+1} such that

$$\begin{array}{lrcrcrcl}
(1) & \dfrac{u^{n+\theta} - u^n}{\theta\tau} & + & \alpha\lambda u^{n+\theta} & + & \beta\lambda u^n & = & 0, \\
(2) & \dfrac{u^{n+\theta'} - u^{n+\theta}}{(1-2\theta)\tau} & + & \beta\lambda u^{n+\theta'} & + & \alpha\lambda u^{n+\theta} & = & 0, \\
(3) & \dfrac{u^{n+1} - u^{n+\theta'}}{\theta\tau} & + & \alpha\lambda u^{n+1} & + & \beta\lambda u^{n+\theta'} & = & 0.
\end{array}$$

The damping factor of the fractional step θ-scheme is given by

$$\omega(z) = \frac{(1 - \beta\theta z)^2 (1 - \alpha(1-2\theta)z)}{(1 + \alpha\theta z)^2 (1 + \beta(1-2\theta)z)}.$$

Now, by examining ω we can deduce the properties of Algorithm 4.1. First we observe that $|\omega(z) - e^{-z}| = \mathcal{O}(|z|^3)$ for $z \to 0$. Thus the scheme is of second order. Looking at the asymptotic behavior we recognize that $\sup_{z \in \mathbb{R}_+} |\omega(z)| \leq 1$ and that $\lim_{z\to\infty} |\omega(z)| = \beta/\alpha < 1$ since $\beta < \alpha$. Hence the scheme is strongly A-stable (cf. [16]). If we further choose $\alpha = (1-2\theta)/\theta'$ the implicit parts in Algorithm 4.1 are identical and $|\omega(i\tau)| \approx 1$. This means that the scheme is nearly non-dissipative (cf. [16]). So the fractional step θ-scheme has all advantages of the Crank-Nicolson scheme ($\omega(z) = (2-z)/(2+z)$) and even better stability properties. In comparison to the implicit Euler scheme ($\omega(z) = 1/(1+z)$) we observe that it has a higher approximation order and is less dissipative, meaning that the damping effect is lesser.

For the time discretization of the Navier-Stokes equations we use the operator splitting variant of the fractional step θ-scheme which was proposed by Bristeau/Glowinski/Periaux [12]. For further informations on the analysis of this variant one may also look at [43].

Algorithm 4.2 (Time discretization of the Navier-Stokes equations).
Let $\boldsymbol{u}_0$ be an initial velocity field and $\boldsymbol{f}$ the time-dependent right hand side of equation (2.7). Set $\boldsymbol{u}^0 = \boldsymbol{u}_0$. For $n \geq 0$ find $\boldsymbol{u}^{n+\theta}$, $\boldsymbol{u}^{n+\theta'}$, $\boldsymbol{u}^{n+1}$ and $p^{n+\theta}$, p^{n+1} such that

$$(1)\qquad \begin{aligned} \frac{\boldsymbol{u}^{n+\theta} - \boldsymbol{u}^n}{\theta\tau} - \frac{\alpha}{Re}\Delta \boldsymbol{u}^{n+\theta} + \nabla p^{n+\theta} & \\ = \boldsymbol{f}^{n+\theta} + \frac{\beta}{Re}\Delta \boldsymbol{u}^n - \boldsymbol{u}^n \cdot \nabla \boldsymbol{u}^n & \quad \textit{in } \Omega^\ell, \\ \nabla \cdot \boldsymbol{u}^{n+\theta} = 0 & \quad \textit{in } \Omega^\ell, \\ \boldsymbol{u}^{n+\theta} = 0 & \quad \textit{on } \partial\Omega^\ell, \end{aligned}$$

$$(2)\qquad \begin{aligned} \frac{\boldsymbol{u}^{n+\theta'} - \boldsymbol{u}^{n+\theta}}{(1-2\theta)\tau} - \frac{\beta}{Re}\Delta \boldsymbol{u}^{n+\theta'} + \boldsymbol{u}^{n+\theta'} \cdot \nabla \boldsymbol{u}^{n+\theta'} & \\ = \boldsymbol{f}^{n+\theta'} + \frac{\alpha}{Re}\Delta \boldsymbol{u}^{n+\theta} - \nabla p^{n+\theta} & \quad \textit{in } \Omega^\ell, \\ \boldsymbol{u}^{n+\theta'} = 0 & \quad \textit{on } \partial\Omega^\ell, \end{aligned}$$

$$(3)\qquad \begin{aligned} \frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^{n+\theta'}}{\theta\tau} - \frac{\alpha}{Re}\Delta \boldsymbol{u}^{n+1} + \nabla p^{n+1} & \\ = \boldsymbol{f}^{n+1} + \frac{\beta}{Re}\Delta \boldsymbol{u}^{n+\theta'} - \boldsymbol{u}^{n+\theta'} \cdot \nabla \boldsymbol{u}^{n+\theta'} & \quad \textit{in } \Omega^\ell, \\ \nabla \cdot \boldsymbol{u}^{n+1} = 0 & \quad \textit{in } \Omega^\ell, \\ \boldsymbol{u}^{n+1} = 0 & \quad \textit{on } \partial\Omega^\ell. \end{aligned}$$

This algorithm splits the Navier-Stokes equations such that for each time step two linear saddle problems (step (1) and (3)) and one nonlinear problem (step (2)) have to be solved. This is a very efficient approach even though the scheme in the operator splitting variant loses second order (cf. [43]).

Note that the algorithm keeps its characteristic features if in step (1) $\boldsymbol{f}^{n+\theta}$ is replaced by $\boldsymbol{f}^n$ and in step (2) $\boldsymbol{f}^{n+\theta'}$ by $\boldsymbol{f}^{n+1}$ (cf. [43]). This saves one calculation of $\boldsymbol{f}$ per time step.

4.2.2 Spatial discretization

For the discretization in space we use the so called Taylor-Hood element proposed in [28]. It uses Lagrange finite element spaces for the flow velocity and the pressure with the constraint that the polynomial degree of the pressure space is one less than the degree of the velocity space, i.e.

$$(\boldsymbol{u}_h, p_h) \in \mathring{V}_h^k(\Omega_h^\ell) \times P_h^{k-1}(\Omega_h^\ell)$$

for $k \geq 2$. For the definition of the spaces see equations (4.4) and (4.2). With this choice of the finite element spaces we obtain a stable discretization for the Stokes problem which occurs in the first and the third step of Algorithm 4.2. This is a saddle point problem, i.e. for the stability of the discrete solution with regard to the mesh size h the discrete spaces must fulfill the LBB-condition

$$\inf_{\substack{q_h \in P_h^{k-1}(\Omega_h^\ell)\\ q_h \neq 0}} \sup_{\substack{\boldsymbol{v}_h \in \mathring{V}_h^k(\Omega_h^\ell)\\ \boldsymbol{v}_h \neq 0}} \frac{(q_h, \nabla \cdot \boldsymbol{v}_h)_{\Omega_h^\ell}}{\|q_h\|_{\Omega_h^\ell} \|\boldsymbol{v}_h\|_{1,\Omega_h^\ell}} \geq \alpha_k > 0$$

with a constant α_k independent of h (cf. [11] and section 3.2.2 in the last chapter). For the polynomial degree $k = 2$ the proof for the LBB-condition can be found in [22], results for higher polynomial degrees can be found in [44] although are not used in this work.

4.2.3 The fully discretized problem

Before we state the complete discrete algorithm let us introduce the following bilinear and trilinear forms. For $\boldsymbol{u}_h, \boldsymbol{v}_h, \boldsymbol{w}_h \in \mathring{V}_h^k(\Omega_h^\ell)$ and $q_h \in P_h^{k-1}(\Omega_h^\ell)$ we define

$$\begin{aligned} a(\boldsymbol{u}_h, \boldsymbol{v}_h) &= \int\limits_{\Omega_h^\ell} \nabla \boldsymbol{u}_h : \nabla \boldsymbol{v}_h, \\ b(\boldsymbol{u}_h; \boldsymbol{v}_h, \boldsymbol{w}_h) &= \int\limits_{\Omega_h^\ell} \boldsymbol{u}_h \cdot \nabla \boldsymbol{v}_h \, \boldsymbol{w}_h, \\ c(q_h, \boldsymbol{v}_h) &= -\int\limits_{\Omega_h^\ell} q_h \nabla \cdot \boldsymbol{v}_h. \end{aligned}$$

Now we can formulate the algorithm for the calculation of the Navier-Stokes equations with homogeneous Dirichlet boundary conditions and initial condition $\boldsymbol{u}_0 = 0$.

Algorithm 4.3 (Discretization of the Navier-Stokes equations).
Set $\boldsymbol{u}_h^0 = 0$. *For* $n \geq 0$ *find* $\boldsymbol{u}_h^{n+\theta}, \boldsymbol{u}_h^{n+\theta'}, \boldsymbol{u}_h^{n+1} \in \mathring{V}_h^k(\Omega_h^\ell)$ *and* $p_h^{n+\theta}, p_h^{n+1} \in P_h^{k-1}(\Omega_h^\ell)$ *such that*

$$\text{(1)}\quad \begin{aligned} \Big(\frac{\boldsymbol{u}_h^{n+\theta} - \boldsymbol{u}_h^n}{\theta\tau}, \boldsymbol{v}_h\Big)_{\Omega_h^\ell} + \frac{\alpha}{Re} a(\boldsymbol{u}_h^{n+\theta}, \boldsymbol{v}_h) \qquad & \\ + c(p_h^{n+\theta}, \boldsymbol{v}_h) &= (\boldsymbol{f}_h^n, \boldsymbol{v}_h)_{\Omega_h^\ell} \\ &\quad - \frac{\beta}{Re} a(\boldsymbol{u}_h^n, \boldsymbol{v}_h) - b(\boldsymbol{u}_h^n; \boldsymbol{u}_h^n, \boldsymbol{v}_h) \quad \forall \boldsymbol{v}_h \in \mathring{V}_h^k(\Omega_h^\ell), \\ c(q_h, \boldsymbol{u}_h^{n+\theta}) &= 0 \qquad \forall q_h \in P_h^{k-1}(\Omega_h^\ell), \end{aligned}$$

$$\text{(2)}\quad \begin{aligned} \Big(\frac{\boldsymbol{u}_h^{n+\theta'} - \boldsymbol{u}_h^{n+\theta}}{(1-2\theta)\tau}, \boldsymbol{v}_h\Big)_{\Omega_h^\ell} + \frac{\beta}{Re} a(\boldsymbol{u}_h^{n+\theta'}, \boldsymbol{v}_h) \qquad & \\ + b(\boldsymbol{u}_h^{n+\theta'}; \boldsymbol{u}_h^{n+\theta'}, \boldsymbol{v}_h) &= (\boldsymbol{f}_h^{n+1}, \boldsymbol{v}_h)_{\Omega_h^\ell} \\ &\quad - \frac{\alpha}{Re} a(\boldsymbol{u}_h^{n+\theta}, \boldsymbol{v}_h) - c(p_h^{n+\theta}, \boldsymbol{v}_h) \quad \forall \boldsymbol{v}_h \in \mathring{V}_h^k(\Omega_h^\ell), \end{aligned}$$

$$\text{(3)}\quad \begin{aligned} \Big(\frac{\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^{n+\theta'}}{\theta\tau}, \boldsymbol{v}_h\Big)_{\Omega_h^\ell} + \frac{\alpha}{Re} a(\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h) \qquad & \\ + c(p_h^{n+1}, \boldsymbol{v}_h) &= (\boldsymbol{f}_h^{n+1}, \boldsymbol{v}_h)_{\Omega_h^\ell} \\ &\quad - \frac{\beta}{Re} a(\boldsymbol{u}_h^{n+\theta'}, \boldsymbol{v}_h) - b(\boldsymbol{u}_h^{n+\theta'}; \boldsymbol{u}_h^{n+\theta'}, \boldsymbol{v}_h) \quad \forall \boldsymbol{v}_h \in \mathring{V}_h^k(\Omega_h^\ell), \\ c(q_h, \boldsymbol{u}_h^{n+1}) &= 0 \qquad \forall q_h \in P_h^{k-1}(\Omega_h^\ell). \end{aligned}$$

For the calculation of the Navier-Stokes equations we use the Navier-Stokes library of the finite element toolbox ALBERTA (cf. [49]) which provides all necessary tools for an easy assemblage of the above steps. In fact all that is needed is a user defined routine for the calculation of the right hand side $\boldsymbol{f}$.

4.3 Moving interface approach

In this section we deal with the approach presented in Problem 3.9 taking the motion of the interface between solid and liquid part of the fixed domain Ω into account. To be more precise, the discrete interface Γ_h which is initially represented by a set of edges (2d) or faces (3d) of the triangulation $\mathcal{T}_h$ will move with its discrete normal velocity $v_{\Gamma,h}$. Hence the grid will be deformed. This also ensures conservation of mass for the discrete solutions $c_{i,h}$ in the domain Ω_h. The numerical treatment of the problem is very complicated mainly because the new grid position and corresponding finite element space are not known beforehand in each time step.

4.3.1 Space-time finite elements

The natural approach for discretizations on time-dependent domains with finite elements is the use of so called space-time finite elements. This technique is quite old and was already used in the seventies to calculate free boundary problems (cf. for instance [5]). For later studies on this topic see for instance [4]. Space-time finite elements are an extension of the so called Discontinuous Galerkin method for moving grids. The original idea of this method was to apply Galerkin's method not only for the space but also for the time variable. Thus the discrete solution is also a piecewise polynomial function in time, not necessarily continuous. For an extensive study of this related method we refer to the book of Thomée [52] which covers existence and uniqueness theory as well as error estimates for this technique for the case of parabolic problems.

In order to define space-time finite elements let $\hat{\Omega}_h = \Omega_h(0)$ be the reference domain and $t_n = n\tau$ $(0 \leq n \leq N_\tau)$ a partition of the time interval $[0, T]$ with fixed time step size. In $\hat{\Omega}_h$ we assume a conforming triangulation $\mathcal{T}_h$ as described in Section 4.1. For a time step t_n we denote by $\Lambda_n \in V_h^1(\hat{\Omega}_h)$ a deformation of the grid (cf. equation (4.3) for the definition of the space) acting as the identity on the boundary of Ω_h. For $n = 0$ let $\Lambda_0 = \mathrm{id}_{\hat{\Omega}_h}$. Define a time-dependent parameterization Φ of $\Omega_h(t)$ as

$$
\begin{aligned}
&\Phi_n \;:\; [t_{n-1}, t_n] \times \hat{\Omega}_h \to \Omega_h(t),\\
\text{(4.9)}\qquad &\Phi_n(t,\hat{x}) \;=\; \frac{t_n - t}{\tau}\Lambda_{n-1}(\hat{x}) + \frac{t - t_{n-1}}{\tau}\Lambda_n(\hat{x}),\\
&\Phi(t,\hat{x}) = \sum_{n=1}^{N_\tau} \Phi_n(t,\hat{x})\chi_{(t_{n-1},t_n]},
\end{aligned}
$$

where $\chi_{(t_{n-1},t_n]}$ denotes the characteristic function on $(t_{n-1}, t_n]$. To get a proper space-time domain we require that for every $t \in [0,T]$ the transformation $\Phi(t,\cdot)$ is injective and that

$$\inf_{\hat{x}\in\hat{\Omega}_h} \det D_{\hat{x}}\Phi(t,\hat{x}) \;>\; 0,$$

where $D_{\hat{x}}\Phi(t,\hat{x})$ denotes the Jacobian with respect to $\hat{x}$ of Φ at $(t,\hat{x})$. This is no restriction for our problem since the movement of the grid nodes in the domain is very small. Note, that in our special case $\Omega_h(t) = \hat{\Omega}_h$ for all $t \in [0,T]$ since the area of the whole domain does not change whereas the subdomains may change in time. Thus for the spatial integrals we will just use Ω_h as denotation for the integration area. For easier notation let $G = (0,T) \times \hat{\Omega}_h$, $G_n = (t_{n-1}, t_n) \times \hat{\Omega}_h$, $S = \big\{ (t,x) \in G \,\big|\, x \in \Gamma_h(t) \big\}$, $S_n = \big\{ (t,x) \in G_n \,\big|\, x \in \Gamma_h(t) \big\}$, and $S_n^o = \big\{ (t,x) \in G_n \,\big|\, x \in \Gamma_h^o(t) \big\}$ and define

$$\left(\varphi,\psi\right)_{G_n} \;=\; \int_{t_{n-1}}^{t_n} \int_{\Omega_h} \varphi\,\psi\,dx\,dt,$$

and $(\varphi,\psi)_{S_n}$, $(\varphi,\psi)_{S_n^o}$, $(\varphi,\psi)_G$ accordingly.

Now for a given function $\varphi : G \to \mathbb{R}$ we define its related reference function as

$$\hat{\varphi}(t,\hat{x}) \;:=\; \varphi(t,\Phi(t,\hat{x})).$$

With the help of this tool the space-time finite element spaces can be defined by means of the finite element spaces given in Section 4.1, as

$$
\begin{aligned}
Y_h^k \;&=\; \Big\{ \varphi_h : G \to \mathbb{R} \,\Big|\, \hat{\varphi}_h|_{(t_{n-1},t_n]} \in P_0((t_{n-1},t_n]) \times X_h^k(\hat{\Omega}_h),\; 1 \le n \le N_\tau \Big\},\\
\mathcal{Y}_h^k \;&=\; \Big\{ \varphi_h : G \to \mathbb{R} \,\Big|\, \hat{\varphi}_h|_{(t_{n-1},t_n]} \in P_0((t_{n-1},t_n]) \times \mathcal{X}_h^k(\hat{\Omega}_h^\bullet),\; 1 \le n \le N_\tau \Big\},
\end{aligned}
$$

and on the moving interface $\Gamma_h(t)$ as

$$N_h^k \;=\; \left\{\psi_h : S \to \mathrm{I\!R} \;\middle|\; \hat{\psi}_h|_{(t_{n-1},t_n]} \in P_0((t_{n-1},t_n]) \times M_h^k(\hat{\Gamma}_h),\ 1 \leq n \leq N_\tau\right\}.$$

Note that the functions φ_h may be discontinuous in time at t_n. On account of this we denote for $\varphi_h \in Y_h^k$ (or $\varphi_h \in \mathcal{Y}_h^k$) and $t = t_n$ with φ_h^n the limit from below t_n, i.e. $\varphi_h^n(x) = \lim_{s\to 0+} \varphi_h(t_n - s, x)$. With φ_h^{n+0} we denote the limit from above t_n, i.e. $\varphi_h^{n+0}(x) = \lim_{s\to 0+} \varphi_h(t_n + s, x)$. The jump at $t = t_n$ is denoted by $[\varphi]^n = \varphi^{n+0} - \varphi^n$. Furthermore, note that for a given function $\varphi_h \in Y_h^k$ the related function $\hat{\varphi}_h$ is piecewise constant in time whereas in general φ_h is not.

Until now we have not mentioned the grid movement. The intention is to move the discrete interface $\Gamma_h(0)$ with its discrete normal velocity $v_{\Gamma,h}$. Therefore we introduce the grid velocity $\boldsymbol{v}_h \in (Y_h^1)^d$ which for $t \in (t_{n-1}, t_n]$ is defined as

$$\hat{\boldsymbol{v}}_h(t, \hat{x}) \;=\; \frac{\Lambda_n(\hat{x}) - \Lambda_{n-1}(\hat{x})}{\tau}. \qquad (4.10)$$

Furthermore we denote by $\boldsymbol{\nu}_h \in (N_h^1)^d$ a vector field equal to the discrete outer normal of $\Omega_h^s(t_n)$ on $\Gamma_h(t_n)$ at every node $\hat{x}_i$ on $\hat{\Gamma}_h$ at time $t = t_n$. There are several possibilities of defining a discrete normal in a grid point. In our calculations we take the mean value of the normals on all neighboring sub-simplices on $\Gamma_h(t_n)$ weighted by their surface area. With this definition we restrict the pointwise movement of Γ_h to its discrete normal direction. At the interior grid nodes, we are free to choose any value for $\boldsymbol{v}_h$. For a big movement of Γ_h it would make sense to require a harmonic distribution for the grid nodes. In our case this only causes unnecessary computational costs. Thus, interior points are not moved at all.

4.3.2 The algorithm

For the derivation of the discrete formulation we consider the first term of the temperature equation in Problem 3.9 and replace the temperature with the discrete temperature $\theta_h \in Y_h^k$. Integrating by parts in time in each prism G_n we obtain

$$\tag{4.11}\begin{aligned} -\int_0^T (\theta_h, \varphi_t)_{\Omega_h}\,dt &= -\sum_{n=1}^{N_\tau}\Big((\theta_h,\varphi)_{\Omega_h}\Big|_{t_{n-1+0}}^{t_n} - \int_{G_n}(\theta_{h,t},\varphi)_{\Omega_h}\,dt\Big) \\ &= (\theta_{h,t},\varphi)_G + \sum_{n=1}^{N_\tau-1}([\theta_h]^n,\varphi^n)_{\Omega_h} + (\theta_h^{0+0},\varphi^0)_{\Omega_h}, \end{aligned}$$

where $\varphi^n(x) := \varphi(t^n, x)$ and $\varphi(T, x) = 0$ for $\varphi \in X^1(T, \Omega_h)$ (cf. equation (3.24)). If we further replace φ with a discrete function $\varphi_h \in Y_h^k$ the right hand side of equation (4.11) is given as

$$\tag{4.12}\text{r.h.s.} = (\theta_{h,t},\varphi_h)_G + \sum_{n=1}^{N_\tau-1}([\theta_h]^n,\varphi_h^{n+0})_{\Omega_h} + (\theta_h^{0+0},\varphi_h^{0+0})_{\Omega_h}.$$

Since functions in Y_h^k are not required to be continuous at t_n, we may choose φ_h to vanish outside $(t_{n-1}, t_n]$. Equation (4.12) then reduces to

$$\tag{4.13}\text{r.h.s.} = (\theta_{h,t},\varphi_h)_{G_n} + (\theta_h^{n-1+0},\varphi_h^{n-1+0})_{\Omega_h} - (\theta_h^{n-1},\varphi_h^{n-1+0})_{\Omega_h}.$$

With this choice of test function φ_h the remaining terms in the temperature equation of Problem 3.9 are simply restricted to the time interval $(t_{n-1}, t_n]$ since there is no further time derivative. Before we state the algorithm introduce two further modifications. First we again integrate by parts in time and so equation (4.13) becomes

$$\tag{4.14}\text{r.h.s.} = -(\theta_h,\varphi_{h,t})_{G_n} + (\theta_h^n,\varphi_h^n)_{\Omega_h} - (\theta_h^{n-1},\varphi_h^{n-1+0})_{\Omega_h}.$$

Second we consider the trace of any particle $x(t) = \Phi(t, \hat{x})$. Velocities of these are given by $\boldsymbol{v}_h(t, x(t))$ (cf. equation (4.10)). Since the discrete functions are constant on the trace of a particle we observe for any $\psi_h \in Y_h^k$ and $t \in (t_{n-1}, t_n)$ that

$$\tag{4.15} 0 = \frac{d}{dt}\psi_h(t,x(t)) = \psi_{h,t}(t,x(t)) + \nabla\psi_h(t,x(t))\cdot\boldsymbol{v}_h(t,x(t)).$$

To eliminate the time derivative in equation (4.13) we substitute it using equation (4.15) and obtain

$$\tag{4.16}\text{r.h.s.} = (\theta_h,\nabla\varphi_h\cdot\boldsymbol{v}_h)_{G_n} + (\theta_h^n,\varphi_h^n)_{\Omega_h} - (\theta_h^{n-1},\varphi_h^{n-1+0})_{\Omega_h}.$$

Applying the same steps to the concentration equations now yields the algorithm with moving interface.

Algorithm 4.4 (Moving interface).
Let $\boldsymbol{u}_h^0 = 0$ *be the initial velocity field. Set* $\theta_h^0 = 0$ *and* $c_{i,h}^0 = \tilde{I}_h c_{i,0}$ $(i = 1, 2)$. *For* $n = 1, \dots, N_\tau$ *find* $\theta_h \in Y_h^k$, $c_{1,h}, c_{2,h} \in \mathcal{Y}_h^k$, *and* Λ_n *such that*

$$\begin{aligned}
&\big(\theta_h^n, \varphi_h^n\big)_{\Omega_h} + \big(\theta_h, \nabla\varphi_h \cdot \boldsymbol{v}_h\big)_{G_n} + \big((\boldsymbol{u}_h^{n-1} \circ \Phi^{-1}) \cdot \nabla\theta_h, \varphi_h\big)_{G_n} \\
&\qquad + \big(\hat{a}\nabla\theta_h, \nabla\varphi_h\big)_{G_n} + \hat{R}\big(\theta_h, \varphi_h\big)_{S_n^o} - \big(\boldsymbol{v}_h \cdot \boldsymbol{\nu}_h, \varphi_h\big)_{S_n} \\
&\qquad\qquad = \big(\theta_h^{n-1}, \varphi_h^{n-1+0}\big)_{\Omega_h} + \hat{R}\big(\hat{\theta}_{ext}, \varphi_h\big)_{S_n^o} \qquad \forall \varphi_h \in Y_h^k, \\
&\big(c_{i,h}^n, \varphi_h^n\big)_{\Omega_h} + \big(c_{i,h}, \nabla\varphi_h \cdot \boldsymbol{v}_h\big)_{G_n} + \big((\boldsymbol{u}_h^{n-1} \circ \Phi^{-1}) \cdot \nabla c_{i,h}^n, \varphi_h\big)_{G_n} \\
&\qquad + \big(\hat{D}_i \nabla c_{i,h}, \nabla\varphi_h\big)_{G_n} = \big(c_{i,h}^{n-1}, \varphi_h^{n-1+0}\big)_{\Omega_h} \quad (i = 1, 2) \qquad \forall \varphi_h \in Y_h^k,
\end{aligned}$$

and

$$\begin{aligned}
\big(c_{1,h}^s, \psi_h\big)_{S_n} &= \big(\tfrac{0.11 c_{1,h}^\ell}{1 - 0.78 c_{1,h}^\ell - c_{2,h}^\ell}, \psi_h\big)_{S_n} && \forall \psi_h \in N_h^k, \\
\big(c_{2,h}^s, \psi_h\big)_{S_n} &= \big(0.5, \psi_h\big)_{S_n} && \forall \psi_h \in N_h^k, \\
\big(\theta_h^n, \mu_h\big)_{S_n} &= \big(1.36 - 0.56 c_{2,h}^{\ell,n} + 0.65 c_{1,h}^{\ell,n} - \tfrac{1.03 c_{1,h}^{\ell,n}}{1 - c_{2,h}^{\ell,n}}, \mu_h\big)_{S_n} && \forall \mu_h \in N_h^m,
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{v}_h^n(\Lambda_n(\hat{x}_i)) &\text{ is a multiple of } \boldsymbol{\nu}_h^n(\Lambda_n(\hat{x}_i)) && \text{for } i = 1, \dots, j_h^1, \\
\boldsymbol{v}_h^n(\Lambda_n(\hat{x}_i)) &= 0 && \text{for } i = j_h^1 + 1, \dots, N_h^1.
\end{aligned}$$

Finally solve the Navier-Stokes equations on $\hat{\Omega}_h$ *for time step* n *with right hand side*

$$\boldsymbol{f}_h^n = -\Big(\hat{\beta}_\theta\,(\theta_h^n \circ \Phi) + \sum_{i=1,2} \hat{\beta}_{c_i}\big((c_{i,h}^n \circ \Phi) - \tilde{I}_h c_{i,0}\big)\Big)\hat{\boldsymbol{g}},$$

to get the new flow velocity $\boldsymbol{u}_h^n \in X_h^p(\hat{\Omega}_h)^d$.

Let us make some completing remarks on this algorithm.

Remark 4.5.

- *The thickness of the interface can be calculated directly on* $\hat{\Gamma}_h$ *as*

$$d_h(\hat{x}) \;=\; |\Lambda_{N_\tau}(\hat{x}) - \hat{x}|.$$

- *The space-time integrals in the algorithm may be approximated by space integrals through the application of a quadrature formula in time.*

- *The geometrical movement of the domain is neglected for the calculation of the Navier-Stokes equations. Since the movement is very small this has no visible effect to the results.*

4.4 Fixed interface approach

In this section we present the discretization of the system as it is stated in Problem 3.11. The movement of the free boundary Γ in the whole process is only about 20 microns. Compared to the size of the whole domain this is very small and thus neglected.
The method presented here is a very simple approach to the problem. It is used for most of the simulations done in this work because it is a faster and more efficient method compared to Algorithm 4.4. The liquid and solid domain are fixed. Nevertheless the velocity of the interface is still an unknown in the system and has to be calculated in every time step. The thickness of the layer can thus be reconstructed.

4.4.1 Discretization

For the discretization in time we use a simple backward Euler scheme. The time step size τ will be fixed and we denote by t_n the time for the n-th time step, i.e. $t_n = n\tau$, and by N_τ the total number of time steps, i.e. $T = N_\tau \tau$. Furthermore the temperature at time t_n will be denoted by $\theta^n = \theta(t_n, \cdot)$, the concentrations and the interface velocity accordingly. The time derivative of the temperature at time t_n is then approximated by the difference quotient

$$\theta^n \;\approx\; \frac{\theta^n - \theta^{n-1}}{\tau}.$$

For the spatial discretization we assume an appropriate conforming triangulation $\mathcal{T}_h$ of Ω_h inducing a conforming triangulation $\mathcal{S}_h$ of Γ_h. In terms of Section 4.1 the temperature at time t_n is then approximated by

$$\theta_h^n(x) \;=\; \sum_{j=1}^{N_h^k} \theta_j^n \varphi_j(x), \qquad x \in \Omega_h, \tag{4.17}$$

the concentrations by

$$c_{i,h}^n(x) \;=\; \sum_{j=1}^{N_h^k + j_h^k} c_{i,j}^n \varphi_j(x), \quad x \in \Omega_h^{\bullet}, \tag{4.18}$$

and the interface velocity by

$$v_{\Gamma,h}^n(x) \;=\; \sum_{j=1}^{n_h^m} v_{\Gamma,j}^n \mu_j(x), \qquad x \in \Gamma_h, \tag{4.19}$$

with real coefficients θ_j^n, $c_{i,j}^n$ and $v_{\Gamma,j}^n$. The fully discrete temperature for $t \in (0,T]$ and $x \in \Omega_h$ is hence given by

$$\theta_h^\tau(t,x) \;=\; \sum_{n=1}^{N_\tau} \theta_h^n(x)\, \chi_{(t_{n-1},t_n]}. \tag{4.20}$$

4.4.2 The algorithm

The algorithm for the fixed interface approach does not differ much from the weak formulation in Problem 3.11. The continuous spaces for the weak formulation are now replaced by the discrete spaces. Recall that for the boundary equations we dealt with the two spaces $H^{\frac{1}{2}}(\Gamma)$ and $H^{-\frac{1}{2}}(\Gamma)$. The first space is the trace space of $H^1(\Omega)$. Thus the related discrete spaces need to use the same polynomial degree k, i.e. $X_h^k(\Omega_h)$ and $M_h^k(\Gamma_h)$. However, for representing the discrete interface velocity we are free to choose a different degree m yielding $M_h^m(\Gamma_h)$. The relation between the two degrees k and m is of utmost importance for the stability of the discretization (cf. Section 4.5.1).

Algorithm 4.6 (Fixed interface).
Let $\boldsymbol{u}_h^0 = 0$ be the initial velocity field. Set $\theta_h^0 = 0$ and $c_{i,h}^0 = \tilde{I}_h c_{i,0}$ $(i=1,2)$.
For $n = 1, \ldots, N_\tau$ find $\theta_h^n \in X_h^k(\Omega_h)$, $c_{i,h}^n \in \mathcal{X}_h^k(\Omega_h^\bullet)$ $(i=1,2)$, and $v_{\Gamma}{}_h^n \in M_h^m(\Gamma_h)$ satisfying

$$\begin{aligned}
\tfrac{1}{\tau}\big(\theta_h^n, \varphi_h\big)_{\Omega_h} + \big(\boldsymbol{u}_h^{n-1}\cdot \nabla\theta_h^n, \varphi_h\big)_{\Omega_h} + \big(\hat{a}\nabla\theta_h^n, \nabla\varphi_h\big)_{\Omega_h} + \hat{R}\big(\theta_h^n, \varphi_h\big)_{\Gamma_h^o}& \\
-\big(v_{\Gamma,h}^n, \varphi_h\big)_{\Gamma_h} \;=\; \tfrac{1}{\tau}\big(\theta_h^{n-1}, \varphi_h\big)_{\Omega_h} + \hat{R}\big(\hat{\theta}_{ext}, \varphi_h\big)_{\Gamma_h^o} & \qquad \forall \varphi_h \in X_h^k(\Omega_h), \\
\tfrac{1}{\tau}\big(c_{i,h}^n, \varphi_h\big)_{\Omega_h} + \big(\boldsymbol{u}_h^{n-1}\cdot \nabla c_{i,h}^n, \varphi_h\big)_{\Omega_h} + \big(\hat{D}_i \nabla c_{i,h}^n, \nabla\varphi_h\big)_{\Omega_h}& \\
-\big(v_{\Gamma,h}^n \left[c_{i,h}^n\right]_s^\ell, \varphi_h\big)_{\Gamma_h} \;=\; \tfrac{1}{\tau}\big(c_{i,h}^{n-1}, \varphi_h\big)_{\Omega_h} \quad (i=1,2) & \qquad \forall \varphi_h \in X_h^k(\Omega_h),
\end{aligned}$$

and

$$\begin{aligned}
\big(c_{1,h}^{s,n}, \psi_h\big)_{\Gamma_h} &= \big(\tfrac{0.11 c_{1,h}^{\ell,n}}{1 - 0.78 c_{1,h}^{\ell,n} - c_{2,h}^{\ell,n}}, \psi_h\big)_{\Gamma_h} && \forall \psi_h \in M_h^k(\Gamma_h), \\
\big(c_{2,h}^{s,n}, \psi_h\big)_{\Gamma_h} &= \big(0.5, \psi_h\big)_{\Gamma_h} && \forall \psi_h \in M_h^k(\Gamma_h), \\
\big(\theta_h^n, \mu_h\big)_{\Gamma_h} &= \big(1.36 - 0.56 c_{2,h}^{\ell,n} + 0.65 c_{1,h}^{\ell,n} - \tfrac{1.03 c_{1,h}^{\ell,n}}{1 - c_{2,h}^{\ell,n}}, \mu_h\big)_{\Gamma_h} && \forall \mu_h \in M_h^m(\Gamma_h).
\end{aligned}$$

Finally solve the Navier-Stokes equations for time step n with right hand side

$$\boldsymbol{f}_h^n \;=\; -\Big(\hat{\beta}_\theta\, \theta_h^n + \sum_{i=1,2} \hat{\beta}_{c_i}\big(c_{i,h}^n - \tilde{I}_h c_{i,0}\big)\Big)\hat{\boldsymbol{g}},$$

to get the new flow velocity $\boldsymbol{u}_h^n \in X_h^p(\Omega_h)^d$.

The thickness d_h of the interface is reconstructed as

$$d_h(x) \;=\; \tau \sum_{n=1}^{N_\tau} v_{\Gamma,h}^n(x).$$

Remark 4.7. *The duality pairing on the discrete interface Γ_h is replaced by the L^2 inner product on Γ_h since we are not able to calculate the former.*

4.5 Stability aspects and damping methods

It turns out that many difficulties arise in the numerical realization of Algorithm 4.4 and 4.6. Mainly, we have to face oscillations of the interface velocity on the free boundary Γ_h which may gain enough influence on the convergence behavior that the algorithms break down. Therefore we consider two important aspects in this section. First, the knowledge of saddle point theory is used to increase the stability of the discretizations. Second, an additional penalty term is included into the algorithms to damp the oscillations on the interface. This allows simulations on coarser grids as well as more stability in the case of finer grids. The second point is crucial for obtaining usable results as is shown in Chapter 6.

4.5.1 The discrete LBB condition

As already mentioned in Section 3.2.2 it is very important to fulfill the discrete analogon of the LBB condition to get a stable discretization of the problem. Restricting ourself to the fixed interface approach (cf. Section 4.4) this means that there exists a constant $\alpha > 0$ independent of the mesh size h such that

$$\inf_{\substack{\mu_h \in M_h^m(\Gamma_h) \\ \mu_h \neq 0}} \sup_{\substack{\varphi_h \in X_h^k(\Omega_h) \\ \varphi_h \neq 0}} \frac{\left\langle \mu_h, \varphi_h \right\rangle_{\frac{1}{2},\Gamma_h}}{\|\mu_h\|_{-\frac{1}{2},\Gamma_h} \|\varphi_h\|_{1,\Omega_h}} \geq \alpha. \tag{4.21}$$

Many works in the past dealt with the problem of proving the LBB condition for different discretizations. The first work by Babuška (cf. [3]) showed stability of linear Lagrange elements, i.e. $X_h^1(\Omega_h)$, and linear Lagrange elements for the boundary space, i.e. $M_h^1(\Gamma_h)$, under the constraint of having a coarser grid on the boundary subspace. Based on his work Pitkäranta (cf. [45]) presented a slightly different stability condition which is easier to fulfill in practice than the one given by equation (4.21). With this condition the author was able to maintain stability for the same combination of spaces as Babuška without the constraint of having a coarser grid. The price for this are iso-parametric elements for a perfect alignment of Γ_h to Γ (where Γ must be smooth). Much more attention was given to other saddle point problems requiring different finite element spaces and hybrid finite element methods (cf. for instance [11,47]). Thus we are not able to

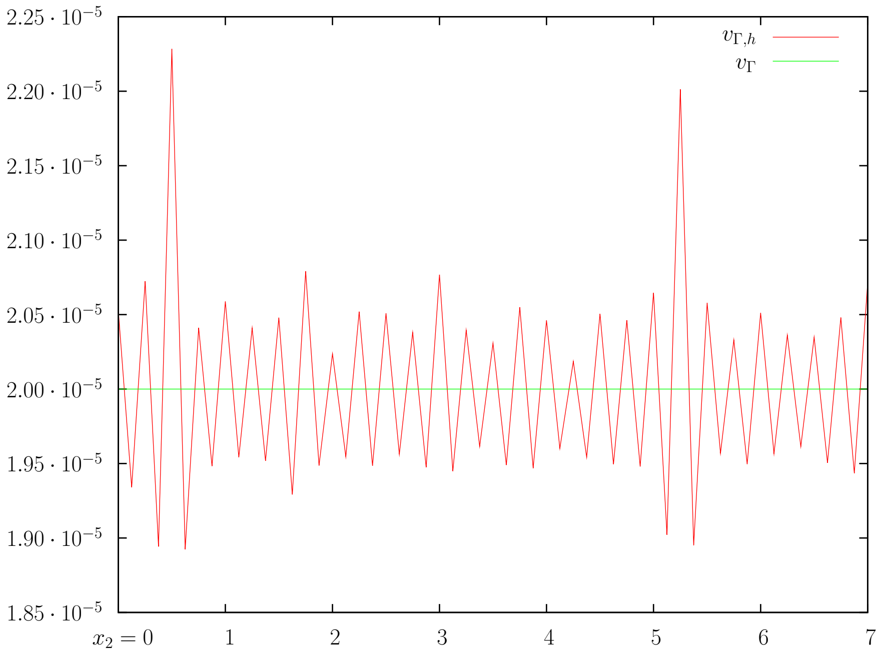


Figure 4.5: Oscillation of the discrete interface velocity in a testing problem (red) and exact solution (green).

cite a proof of the discrete LBB condition here. Nevertheless, we recognize that for equation (4.21) to hold, it is crucial that the discrete space $X_h^k(\Omega_h)$ is large enough compared to the boundary space $M_h^m(\Gamma_h)$. Since in our calculations the degree of the boundary space is always equal to one, i.e. $m = 1$. Increasing the polynomial degree k stabilizes the discretizations. This effect is shown in our numerical examples (cf. Section 6.1.1 and 6.1.2).

Furthermore, we remark that we are not dealing with a true saddle point problem. Our situation is even worse since oscillation effects on the interface Γ_h are amplified by the nonlinear phase diagram equations (2.32). Thus, even in the case that the combination of $X_h^1(\Omega_h)$ and $M_h^1(\Gamma_h)$ would satisfy the discrete LBB condition (4.21) the Algorithms 4.4 and 4.6 may still be unstable. From this point of view it is absolutely necessary to perform convergence tests of the algorithms as done in Chapter 6.

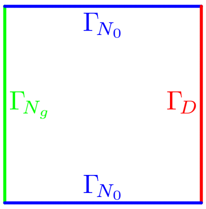
4.5.2 Damping of oscillations on the interface

To shed some light on the origin of the occuring oscillations depicted in Figure 4.5 we consider a reaction/diffusion equation in the domain $\Omega = [0,1]^2$:

$$u - a\,\Delta u \;=\; 0 \quad \text{in } \Omega \tag{4.22}$$

with constant boundary conditions

$$\begin{aligned} u \;&=\; g_D && \text{on } \Gamma_D, \\ \frac{\partial u}{\partial \vec{n}} \;&=\; 0 && \text{on } \Gamma_{N_0}, \\ a\frac{\partial u}{\partial \vec{n}} \;&=\; g_N && \text{on } \Gamma_{N_g}, \end{aligned} \tag{4.23}$$



$\Omega \;=\; [0,1]^2$

where a is a small positive constant. An exact solution of the above equations is given by

$$u(x) \;=\; \exp(-\tfrac{x_1}{\sqrt{a}}), \quad g_D(x) \;=\; \exp(-\tfrac{1}{\sqrt{a}}), \quad g_N(x) \;=\; -\sqrt{a}.$$

If we discretize the above system with linear Lagrange finite elements we expect a discrete solution as shown in Figure 4.6. In fact this only happens for a special choice of the grids where the support of all nodal basis functions on Γ_{N_g} has identical area (with halfed area for the nodes at the edges of the domain). For different grids, as shown in Figure 4.7, we observe small oscillations of the discrete solution along the boundary part Γ_{N_g}.

Normally we would not care about this effect since it does not affect convergence of the discretization of problem (4.22). However, in our case the situation is more complicated. In principle the discrete temperature as well as the discrete concentrations show such a behavior on the discrete interface Γ_h. In fact their oscillations are very small. However they are amplified by the nonlinear phase diagram equations. The result of this effect is shown in Figure 4.5. Since the system is coupled the amplified oscillations enter again into the temperature and the concentration equations. Thus, errors can build up and after many time steps even destroy the convergence of the method.

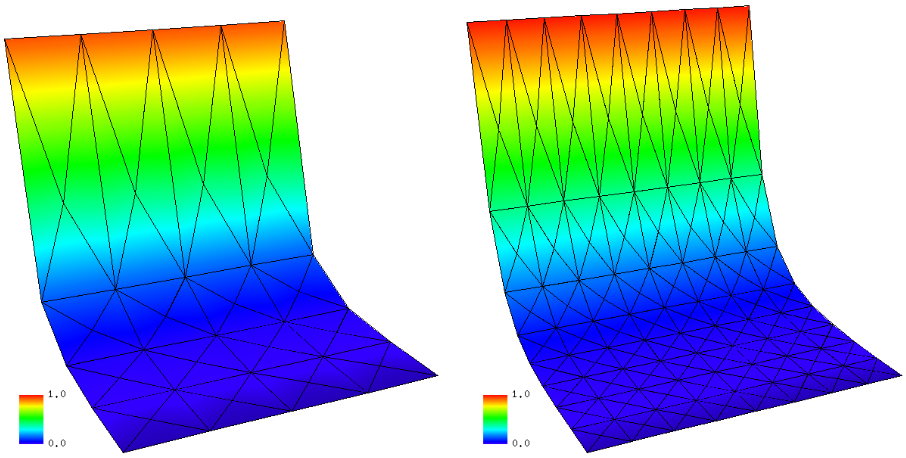


Figure 4.6: Discrete solution of equation (4.22) with parameter $a = 0.05$, maximum grid width $h = \frac{1}{4}$ (left), and $h = \frac{1}{8}$ (right).

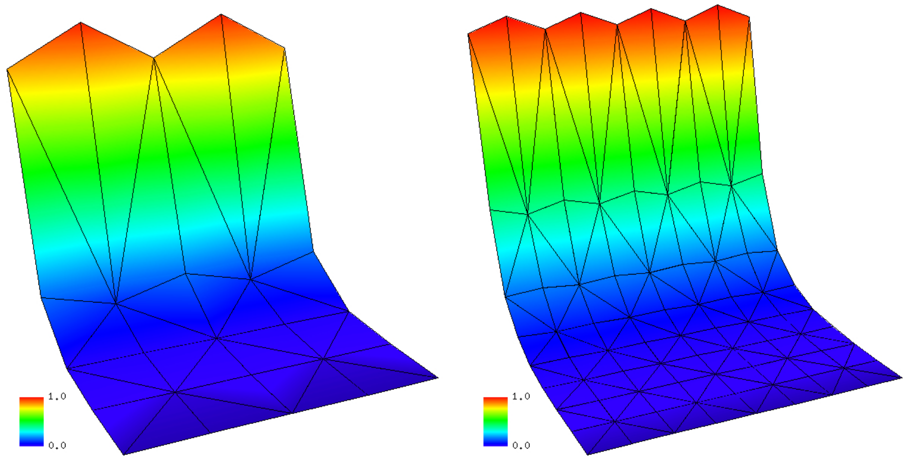


Figure 4.7: Discrete solution of equation (4.22) with parameter $a = 0.05$, maximum grid width $h = \frac{\sqrt{2}}{4}$ (left), and $h = \frac{\sqrt{2}}{8}$ (right).

On special grids (as mentioned above) these oscillations vanish completely and convergence is not affected. Since we are not able to construct such grids for our geometry we have to overcome the problem with a different approach. Following the idea of Burman and Hansbo (cf. [13] and [18]) we add a penalizing term to the phase diagram equation concerning the melting temperature (the fifth equation in Algorithm (4.4) and (4.6)). In three space dimensions the term is given as

$$
\begin{aligned}
J(\lambda_h, \mu_h) &= \gamma \sum_{T \in \mathcal{S}_h} \frac{1}{2} \sum_{K \in \partial T} h_K^2 \int\limits_K \big[\nabla \lambda_h\big] \cdot \big[\nabla \mu_h\big] \, do_x \\
&= \gamma \sum_{T \in \mathcal{S}_h} \frac{1}{2} \sum_{K \in \partial T} h_K^2 \int\limits_K \big[\boldsymbol{\nu} \cdot \nabla \lambda_h\big] \big[\boldsymbol{\nu} \cdot \nabla \mu_h\big] \, do_x ,
\end{aligned}
\tag{4.24}
$$

where γ is a small positive constant, h_K the length of the edge K, $[v]$ denotes the jump of v across K if $K \cap \partial\Gamma_h = \emptyset$ and $[v] = 0$ if $K \cap \partial\Gamma_h \neq \emptyset$, and $\boldsymbol{\nu}$ is the unit outer normal of T. In the two dimensional case where Γ_h is only a polygonal line the term is given as

$$
J(\lambda_h, \mu_h) = \gamma \sum_{T \in \mathcal{S}_h} \frac{h_T^2}{2} \sum_{K \in \partial T} \big[\nabla \lambda_h \cdot \boldsymbol{\nu}\big] \big[\nabla \mu_h \cdot \boldsymbol{\nu}\big] .
\tag{4.25}
$$

Here h_T denotes the length of the edge T, K the endpoints of T and $\boldsymbol{\nu}$ the unit vector parallel to T. The sign of $\boldsymbol{\nu}$ is irrelevant as it later cancels out.

The term J penalizes the jump of the gradient over the boundary of the elements $S \in \mathcal{S}_h$ on the discrete interface Γ_h. Through this it tends to smoothen the discrete solution of the interface velocity. The factors h_K, h_T respectively, ensures consistency and correct scaling during mesh refinement. The factor γ is a control parameter. Increasing γ reduces selectively the oscillations of $v_{\Gamma,h}$ and increases the overall error. Furthermore, the system turns stiffer with increasing γ slowing down the calculation decisively. For values of γ on the order of one we arrive at a good balance between the positive and the negative effects.

Chapter 5

Efficient Solution of the nonlinear algebraic system

In this chapter we deal with the derivation and the numerical solution of the nonlinear algebraic system arising from the discretizations in the last chapter. We present a Newton method with step-size control and discuss the assemblage of the linear system that must be solved in every Newton iteration. For the solution of this system we introduce iterative methods, the Krylov solvers GMRES, BiCGStab, and TFQMR. They are compared with respect to efficiency and robustness. Then, we address the preconditioning method ILUT which speeds up the solution process enormously.

5.1 The Newton method

In Algorithm 4.4 and 4.6 we presented the discretizations of the problem. In each timestep we have to solve a nonlinear sytem of equations coupled with the phase diagram equations (2.32) on the interface Γ_h. In the case of Algorithm 4.4 we have an additional coupling with the motion of the grid. The Newton method (also known as Newton-Rhapson method) is the most popular technique for the iterative solution of a system of nonlinear algebraic equations. It is used to solve the complete system simultaneously.

5.1.1 Newton with step size control

In this section we derive the Newton method and present a corresponding algorithm. Given a nonlinear function $F : \mathbb{R}^n \to \mathbb{R}^n$ we seek a vector $x_\star \in \mathbb{R}^n$ such that

$$F(x_\star) = 0. \tag{5.1}$$

We denote by $x_0 \in \mathbb{R}^n$ an arbitrary initial guess for the solution of the above equation. The idea of Newton's method is to perform a Taylor expansion of F about x_0 in order to find a and update $d \in \mathbb{R}^n$ such that $|F(x_0+d)| < |F(x_0)|$:

$$F(x_0 + d) = F(x_0) + DF(x_0)d + r(x_0).$$

Here $r(x_0)$ is a higher order remainder term (in d) and $DF(x_0)$ denotes the Jacobian of F in x_0. If the remainder term is ignored, the requirement $F(x_0+d) = 0$ yields

$$d = -(DF(x_0))^{-1}F(x_0).$$

If this process it iterated we obtain the following algorithm for the solution of equation (5.1). Here s denotes the step size, m_{max} controls the maximal number of step size reductions per Newton iteration.

Algorithm 5.1 (Newton method with step size control).

```
Compute $r_0 = F(x_0)$;
Set $s = 1.0$, $err = err_{old} = \|r_0\|_2$;
For $m = 0, 1, 2, \ldots$ until $(err < TOL)$ Do:
    Solve $DF(x_m)d = r_m$;
    If $s$ is modified set $s = 2 * s$;
    For $j = 0, \ldots, m_{max}$ Do:
        Set $y = x_m - s * d$, $r_{m+1} = F(y)$, $err = \|r_{m+1}\|_2$;
        If $(err \leq (1 - 0.5 * s) * err_{old})$ break;
        Else set $s = 0.5 * s$;
    EndFor
    Set $x_{m+1} = y$, $err_{old} = err$;
EndFor
```

It is well known that in the neighborhood of the solution $x_\star$ the Newton method has quadratic convergence order. For our problem, for instance, in the majority of cases we need only one Newton iteration to reach the required tolerance. For a deeper discussion of the Newton method and its variants we refer to [15] or [56].

5.1.2 Assemblage of the Jacobian

In this section we take a closer look at the Jacobian to be inverted in step 4 of Algorithm 5.1. For the moment we restrict ourselves to the fixed interface approach as presented in Algorithm 4.6). At the end of the section we will append some remarks about the moving interface approach as presented in Algorithm 4.4.

The unknowns in the algebraic system to be solved in every time step of Algorithm 4.6 are the coefficients for discrete temperature $\theta_h^n(x)$, concentrations $c_{i,h}^n(x)$, and interface velocity $v_{\Gamma,h}^n(x)$ as defined in equation (4.17)–(4.19). To simplify notation we assume the coefficients as follows:

$$
\begin{aligned}
\boldsymbol{\theta^n} &:= (\theta_1^n, \dots, \theta_{N_h^k}^n)^T,\\
\boldsymbol{c_i^n} &:= (c_{i,1}^n, \dots, c_{i,N_h^k}^n)^T,\\
\boldsymbol{c_{i,z}^n} &:= (c_{i,N_h^k+1}^n, \dots, c_{i,N_h^k+j_h^k}^n)^T,\\
\boldsymbol{v_\Gamma^n} &:= (v_{\Gamma,1}^n, \dots, v_{\Gamma,n_h^m}^n)^T.
\end{aligned}
$$

The iteration loop in step 3 of Algorithm 5.1 would require a second iteration index m for the coefficient vectors, $\boldsymbol{\theta^{n,m}}$ for instance. We avoid this and always denote the coefficient vector in each iteration by $\boldsymbol{\theta^n}$. Furthermore the initial temperature coefficient vector $\boldsymbol{\theta^{n-1}}$ in the first Newton iteration is denoted by $\boldsymbol{\theta^n}$ as well. For the concentrations, interface velocity, and the matrices we proceed in the same manner.

To show how the Jacobian is generated we consider exemplarily the first term $\frac{1}{\tau}\left(\theta_h^n, \varphi_h\right)_{\Omega_h}$ of the first equation in Algorithm 4.6. Since we have a basis of the discrete test space $X_h^k(\Omega_h)$ it is enough to test the equation with all basis elements φ_l with $l = 1, \dots, N_h^k$. If we use the representation formula (4.17) for the temperature we obtain

$$\frac{1}{\tau}\big(\theta_h^n,\varphi_l\big)_{\Omega_h} = \frac{1}{\tau}\big(\sum_{j=1}^{N_h^k}\theta_j^n\varphi_j,\varphi_l\big)_{\Omega_h} = \frac{1}{\tau}\sum_{j=1}^{N_h^k}\theta_j^n\big(\varphi_j,\varphi_l\big)_{\Omega_h},$$

and further

$$\frac{d}{d\theta_j^n}\Big(\frac{1}{\tau}\sum_{j=1}^{N_h^k}\theta_j^n\big(\varphi_j,\varphi_l\big)_{\Omega_h}\Big) = \frac{1}{\tau}\big(\varphi_j,\varphi_l\big)_{\Omega_h}.$$

Calculating all derivatives with respect to the temperature coefficients $\boldsymbol{\theta^n}$ for the left hand side of the first equation in Algorithm 4.6 yields the following submatrix of the Jacobian

(5.2)
$$\boldsymbol{M_\theta^n} = \Big(\tfrac{1}{\tau}\big(\varphi_j,\varphi_l\big)_{\Omega_h} + \big(u_h^{n-1}\cdot\nabla\varphi_j,\varphi_l\big)_{\Omega_h} + \big(\hat{a}\nabla\varphi_j,\nabla\varphi_l\big)_{\Omega_h}\Big)_{l,j=1,\dots,N_h^k} + \Big(\hat{R}\big(\varphi_j,\varphi_l\big)_{\Gamma_h^o}\Big)_{l,j=1,\dots,N_h^k}.$$

The first part of $\boldsymbol{M_\theta^n}$ is the usual mass matrix, the second one is the antisymmetric convective part (assuming that the discrete velocity field is divergence free), and the third part is the usual stiffness matrix. The last part arises from the Robin boundary condition and is again symmetric. The first equation of Algorithm 4.6 contains further a term with the interface velocity $v_{\Gamma,h}$ on the left hand side. Using equation (4.19) we obtain

$$\frac{d}{dv_{\Gamma,j}^n}\Big(-\big(v_{\Gamma,h}^n,\varphi_l\big)_{\Gamma_h}\Big) = \frac{d}{dv_{\Gamma,j}^n}\Big(-\big(\sum_{j=1}^{n_h^m}v_{\Gamma,j}^n\mu_j,\varphi_l\big)_{\Gamma_h}\Big) = -\big(\mu_j,\varphi_l\big)_{\Gamma_h}$$

and hence an additional submatrix for the derivatives of the first equation in Algorithm 4.6 with respect to the coefficients $\boldsymbol{v_\Gamma^n}$:

$$\boldsymbol{V_\theta^n} = \Big(-\big(\mu_j,\varphi_l\big)_{\Gamma_h}\Big)_{\substack{l=1,\dots,j_h^k\\ j=1,\dots,n_h^m}}. \tag{5.3}$$

Note, that the submatrix consists only of j_h^k rows since we ordered the basis functions $\varphi_j\in X_h^k(\Omega_h)$ such that $\varphi_j|_{\Gamma_h}=0$ for $j_h^k<j\le N_h^k$ (cf. Section 4.1). Altogether, the part of the Jacobian arising from the first equation of Algorithm 4.6 consists of the submatrices defined in equation (5.2) and (5.3) (cf. Figure

5.1). The remaining entries in these rows are zero.

For the second equation in Algorithm 4.6 we obtain similar submatrices. The derivatives with respect to the concentration coefficients $\boldsymbol{c_i^n}$ yield

(5.4)
$$\begin{aligned}\boldsymbol{M_i^n} &= \Big(\tfrac{1}{\tau}\big(\varphi_j,\varphi_l\big)_{\Omega_h} + \big(u_h^{n-1}\cdot\nabla\varphi_j,\varphi_l\big)_{\Omega_h} + \big(\hat{D}_i\nabla\varphi_j,\nabla\varphi_l\big)_{\Omega_h}\Big)_{\substack{l=1,\dots,N_h^k\\ j=1,\dots,N_h^k+j_h^k}} \\ &+ \quad \Big(\big(v_{\Gamma,h}^n\varphi_j,\varphi_l\big)_{\Gamma_h}\Big)_{\substack{l=1,\dots,j_h^k\\ j=N_h^k+1,\dots,N_h^k+j_h^k}} .\end{aligned}$$

Note, that the submatrix is not quadratic since the solution space for the concentrations $\mathcal{X}_h^k(\Omega_h^\bullet)$ was provided with additional basis functions φ_j with $N_h^k < j \leq N_h^k + j_h^k$ to allow the concentrations to jump on Γ_h (cf. Section 4.1). Let us briefly explain the last term of the submatrix $\boldsymbol{M_i^n}$ which arises from the derivation of the nonlinear term $-\big(v_{\Gamma,h}^n\left[c_{i,h}^n\right]_s^\ell,\varphi_h\big)_{\Gamma_h}$. The limit concentrations on the interface Γ_h on the liquid side (cf. equation (4.6)) are given as

$$c_{i,h}^{\ell,n}(x) \;=\; \sum_{j=1}^{j_h^k} c_{i,j}^n\varphi_j|_{\Gamma_h}(x). \tag{5.5}$$

The limit concentrations on the interface regarded on the solid side (cf. equation (4.7)) are given as

$$\begin{aligned}c_{i,h}^{s,n}(x) &= \sum_{l=1}^{j_h^k} c_{i,l}^n\varphi_l|_{\Gamma_h}(x) + \sum_{j=N_h^k+1}^{N_h^k+j_h^k} c_{i,j}^n\varphi_j|_{\Gamma_h}(x) \\ &= c_{i,h}^{\ell,n} + \sum_{j=N_h^k+1}^{N_h^k+j_h^k} c_{i,j}^n\varphi_j|_{\Gamma_h}(x).\end{aligned} \tag{5.6}$$

Hence we may rewrite the nonlinear boundary integral as

$$-\big(v_{\Gamma,h}^n\left[c_{i,h}^n\right]_s^\ell,\varphi_l\big)_{\Gamma_h} \;=\; \Bigg(\Big(\sum_{j=1}^{n_h^m} v_{\Gamma,j}^n\mu_j\Big)\Big(\sum_{j=N_h^k+1}^{N_h^k+j_h^k} c_{i,j}^n\varphi_j\Big),\varphi_l\Bigg)_{\Gamma_h} . \tag{5.7}$$

If now the derivatives with respect to the coefficients $\boldsymbol{c_{i,z}^n}$ are taken we gain the last term of $\boldsymbol{M_i^n}$. Taking further the derivative regarding the coefficients of the

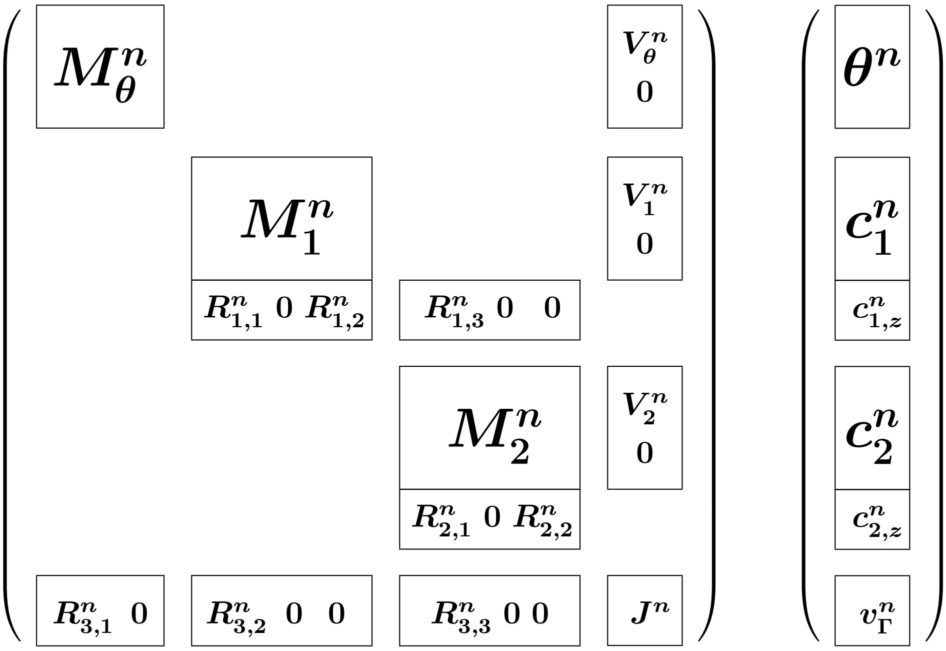


Figure 5.1: Shape of the Jacobian with associated coefficient vectors for the fixed interface approach (Algorithm 4.6). The entries in all missing blocks are zero.

interface velocity $\boldsymbol{v}_\Gamma^n$ an additional submatrix arises:

$$\boldsymbol{V}_i^n = \Big(-\big(\big[c_{i,h}^n\big]_s^\ell \mu_j, \varphi_l\big)_{\Gamma_h} \Big)_{\substack{l=1,\dots,j_h^k \\ j=1,\dots,n_h^m}} .$$

This completes the part of the Jacobian with respect to the second equation in Algorithm 4.6. In Figure 5.1 these are the rows associated with the coefficient vectors $\boldsymbol{c}_1^n$ and $\boldsymbol{c}_2^n$.

From the remaining three equations in Algorithm 4.6 valid on the boundary Γ_h we get additional submatrices related to the rows of the coefficients $\boldsymbol{c}_{1,z}^n$, $\boldsymbol{c}_{2,z}^n$, and $\boldsymbol{v}_\Gamma^n$. Since the first boundary equation contains the variables $c_{1,h}^{s,n}$, $c_{1,h}^{\ell,n}$, and $c_{2,h}^{\ell,n}$ we will have three submatrices $\boldsymbol{R}_{1,i}^n$ $(i = 1, 2, 3)$ of the Jacobian. The submatrix

$\boldsymbol{R^n_{1,3}}$, for instance, has the form

$$\boldsymbol{R^n_{1,3}} \;=\; \left(\left(-\frac{0.11c^{\ell,n}_{1,h}}{(1-0.78c^{\ell,n}_{1,h}-c^{\ell,n}_{2,h})^2}\,\psi_j,\psi_l\right)\right)_{l,j=1,\ldots,j^k_h},$$

where $\psi_j = \varphi_j|_{\Gamma_h}$. The second boundary equation yields two submatrices $\boldsymbol{R^n_{2,i}}$ $(i = 1,2)$. For the third equation we again get three submatrices $\boldsymbol{R^n_{3,i}}$ $(i = 1,2,3)$. Since we further added the penalization term J (cf. equation (4.24) and (4.25)) to this last equation we obtain one more matrix $\boldsymbol{J^n}$ related to the coefficients of the discrete interface velocity $\boldsymbol{v^n_\Gamma}$.

The whole Jacobian arising in each Newton iteration is shown in Figure 5.1. Since the mesh in Algorithm 4.6 is fixed all submatrices arising from linear equations do not change either. On the other hand all submatrices arising from nonlinear equations have to be reassembled in every Newton iteration. Altogether the Jacobian is a sparse matrix. It is regular, but neither symmetric nor definite. In fact it possesses real and complex eigenvalues. This necessitates the use of a robust linear solver suitable for non-symmetric matrices with complex spectrum.

The calculation of the Jacobian for the moving interface approach (Algorithm 4.4) is far more complicated. If we take, for example, a closer look to the first integral $(\theta^n_h, \varphi^n_h)_{\Omega_h}$ of the first equation in Algorithm 4.4, we recognize that its value depends on Λ_n since θ^n_h depends on Λ_n. The derivative of this term with respect to a temperature coefficient is still easy to calculate and yields the same result as for the fixed interface approach

$$\frac{d}{d\theta^n_j}(\theta^n_h, \varphi^n_l)_{\Omega_h} \;=\; (\varphi^n_j, \varphi^n_l)_{\Omega_h}.$$

Concerning the derivatives with respect to a grid velocity coefficients we need to go into more detail. For the sake of simplicity we restrict our considerations to the 2-dimensional case. We consider a single triangle T^n of the triangulation $\mathcal{T}^n_h$ with vertices x^n_1, x^n_2, x^n_3 and the related edge vectors

$$w_1 \;=\; x^n_3 - x^n_2, \quad w_2 \;=\; x^n_1 - x^n_3, \quad w_3 \;=\; x^n_2 - x^n_1. \tag{5.8}$$

Now a parameterization Φ_{T^n} of T^n over the unit simplex $\hat{T}$ is defined as

$$\Phi_{T^n}(\hat{x}) \;=\; \begin{pmatrix} w_3 & -w_2 \end{pmatrix} \hat{x} + x_1.$$

With the upper indices x, y we denote the x, y-component of the vectors or coordinates. Consider the derivative with respect to the x-component of the i-th grid velocity coefficient:

$$\begin{aligned}
\frac{d}{d\boldsymbol{v}_i^{n,x}}\big(\theta_h^n, \varphi_h^n\big)_{T^n} &= \frac{d}{d\boldsymbol{v}_i^{n,x}}\Big(\int_{T^n} \theta_h^n \varphi_h^n \, dx\Big) \\
&= \frac{d}{d\boldsymbol{v}_i^{n,x}}\Big(\int_{\hat{T}} \hat{\theta}_h^n \hat{\varphi}_h^n \det(D\Phi_{T^n})\, d\hat{x}\Big) \\
&= \frac{1}{\det(D\Phi_{T^n})} \int\limits_{\hat{T}} \hat{\theta}_h^n \hat{\varphi}_h^n \det(D\Phi_{T^n})\, d\hat{x}\, \frac{d}{d\boldsymbol{v}_i^{n,x}} \det(D\Phi_{T^n}).
\end{aligned}$$

Since the Jacobian $D\Phi_{T^n} = \begin{pmatrix} w_3 & -w_2 \end{pmatrix}$ we obtain

$$\det(D\Phi_{T^n}) \;=\; w_2^x w_3^y - w_2^y w_3^x \;=\; w_2 \wedge w_3. \tag{5.9}$$

The vertices x_i^n arise from the vertices in the previous timestep as

$$x_i^n = x_i^{n-1} + \tau \boldsymbol{v}_i^n,$$

due to the definition of the transformation Λ_n (cf. equation (4.10)). Thus, using equation (5.8) and (5.9) we obtain

$$\frac{d}{d\boldsymbol{v}_i^{n,x}} \det(D\Phi_{T^n}) \;=\; -\tau w_i^y,$$

and finally

$$\frac{d}{d\boldsymbol{v}_i^{n,x}}\big(\theta_h^n, \varphi_h^n\big)_{T^n} \;=\; -\frac{\tau w_i^y}{w_2 \wedge w_3}\big(\theta_h^n, \varphi_h^n\big)_{T^n}. \tag{5.10}$$

To obtain the submatrix with respect to the x-coordinate of the grid velocity $\boldsymbol{v}_h^n$ equation (5.10) must be evaluated for every triangle of the triangulation $\mathcal{T}_h^n$. For the rest of the terms in Algorithm 4.4 the procedure is much more complicated and is not presented here.

5.2 Linear solver

The linear system we have to handle may become very large even in two space dimensions. Since the sparse Jacobian is non-symmetric and badly conditioned we need robust iteratives techniques for solving it. The Krylov subspace methods which we apply are among the most important iterative techniques available. Since they are based on projection processes we start with an introduction to this field. Afterwards the Krylov methods based on Lanczos' Biorthogonalization process are adressed. We focus on the TFQMR solver which we use for solving the linear system and compare it to the other applicable Krylov solvers GMRES and BiCGStab. For a detailed introduction to iterative methods for sparse systems we refer to [42, 48].

5.2.1 Projection methods

There are many different possibilities of solving a linear problem. For a small number of unknowns direct solution methods such as Gauss elimination are usually most efficient. In many numerical simulations, however, huge systems with sparse matrices occur. In this case iterative techniques are more appropriate. In this section we give an introduction to projection methods which are the basis of many iterative solvers. Starting point is the linear system

$$Ax = b, \tag{5.11}$$

where A is an $n \times n$ real matrix and $x, b \in \mathbb{R}^n$. The idea of projection methods is to extract an approximate solution x_m of equation (5.11) from an affine subspace $\mathcal{K} \subset \mathbb{R}^n$ of candidate approximants, the so-called search subspace, with dimension $m \ll n$. To identify x_m we introduce the subspace of constraints $\mathcal{L}$ and request that the residual vector $r_m := b - Ax_m$ is orthogonal to all vectors in $\mathcal{L}$,

$$\forall w \in \mathcal{L} : \qquad w \perp r_m, \tag{5.12}$$

known as Petrov-Galerkin conditions.

There are two types of projections: For orthogonal projections the subspaces $\mathcal{K}$ and $\mathcal{L}$ are chosen to be the same. In this special case the Petrov-Galerkin conditions are called Galerkin conditions. In an oblique projection method the subspace $\mathcal{L}$ is different from $\mathcal{K}$ and may be totally unrelated to it.

Now let $V = [v_1, \ldots, v_m]$ be an $n \times m$ matrix whose column vectors form a basis of $\mathcal{K}$ and $W = [w_1, \ldots, w_m]$ an $n \times m$ matrix whose column vectors form a basis of $\mathcal{L}$. If we write the approximate solution as

$$x_m = x_0 + Vy, \tag{5.13}$$

where x_0 is an initial guess for the solution of equation (5.11), the orthogonality conditions (5.12) generate the following equations for y and the initial residual vector $r_0 = b - Ax_0$:

$$W^T AVy = W^T r_0. \tag{5.14}$$

The solution of the reduced system (5.14) requires that the matrix $W^T AV$ is nonsingular. There are two important particular cases where this is guaranteed:

(i) A is positive definite and $\mathcal{L} = \mathcal{K}$.

(ii) A is nonsingular and $\mathcal{L} = A\mathcal{K}$.

A proof of this statement is for instance given in [48].

As an example we consider the one-dimensional projection with $\mathcal{K} = \mathrm{span}\{v\}$ and $\mathcal{L} = \mathrm{span}\{w\}$, where v and w are two arbitrary vectors. In this case the approximate solution x_1 can be written as $x_1 = x_0 + \alpha v$ and the Petrov-Galerkin condition (5.12) as $(r_0 - A(\alpha v), w) = 0$. Here (v, w) denotes the usual scalar product of the two vectors $v, w \in \mathbb{R}^n$ and $\|v\| = \sqrt{(v, v)}$ the related norm. The unknown α can now be calculated as

$$\alpha = \frac{(r_0, w)}{(Av, w)}.$$

We iterate this one-dimensional projection and consider the special choice $v = w = r_i$ with $i = 0, 1, \ldots,$ where r_i denotes the residual in the i-th step. One can prove (cf. [48]) that if the matrix A is symmetric and positive definite this iteration process yields the exact solution $x_\star$ independently of the initial guess x_0 in at most n steps. Furthermore the procedure minimizes in every step the function

$$f(x) = \|x - x_\star\|_A^2 = (A(x - x_\star), (x - x_\star)),$$

over all vectors of the form $x_i + \alpha d$, where d is the direction of $-\nabla f(x_i)$. This fact gives the algorithm its name: steepest descent algorithm. Probably the most famous iterative method, the conjugate gradient algorithm (CG), is based on this projection process.

For non-symmetric positive definite matrices A and the choice $v = r_i,\; w = Ar_i$ we obtain again a converging algorithm (cf. [48]). It is called minimal residual method since each step minimizes the residual function

$$f(x) = \|b - Ax\|_2^2$$

over all vectors of the form $x_i + \alpha d$, where d is the direction of the residual r_i. The general minimum residual method (GMRES) (cf. next section) is based on this kind of projection methods.

5.2.2 Krylov methods and Lanczos Biorthogonalization

In this section we give a definition of the Krylov subspaces which are the basis of the powerful Krylov methods. Then, different choices for the subspace of constraints as introduced in the last section are discussed.

Krylov methods are projection methods employing special subspaces for the candidate approximants. These subspaces are related to the matrix A of the linear system (5.11): The m-th Krylov subspace for a vector v and a matrix A is defined as

$$\mathcal{K}_m(A, v) \;=\; \mathrm{span}\{v, Av, A^2v, \ldots, A^{m-1}v\}.$$

Thus, the approximation of the exact solution obtained by a Krylov subspace method is of the form

$$x_\star \;=\; A^{-1}b \;\approx\; x_m \;=\; x_0 + q_{m-1}(A)v,$$

where q_{m-1} is a certain polynomial of degree $m-1$. This is the reason for denoting Krylov methods as a technique of polynomial approximation.

The main difference between the different Krylov algorithms lies in the choice of the subspace of constraints. As already mentioned in the last section $\mathcal{L}_m = \mathcal{K}_m$ leads to the important CG solver for symmetric positive definite matrices. The

CG solver is not of interest to us since the system matrix is neither symmetric nor positive definite. Furthermore we mentioned that for the choice $\mathcal{L}_m = A\mathcal{K}_m$ one obtains the GMRES method which is applicable also for non-symmetric and non-singular matrices. Compared to other Krylov methods GMRES has a big disadvantage. For the calculation of the approximate solution x_m the GMRES method requires the storage of all basis vectors of the Krylov subspace $\mathcal{K}_{m+1}(A, r_0)$. This results in large memory requirements, e.g. in the case $m = n$ this method requires the same amount of memory as a direct solver like Gaussian elimination does — a non-sparse $n \times n$ matrix. Thus it is often used in the restarting variant GMRES(p). Here the dimension of the Krylov subspace is restricted to a fixed $p \ll n$. After the calculation of the approximate solution x_p for the Krylov subspace $\mathcal{K}_p(A, r_i)$ a restart with the new residual $r_{i+1} = b - Ax_p$ as basis for the Krylov subspace $\mathcal{K}_p(A, r_{i+1})$ is performed. One iteration thus consists of solving a projection onto a p-dimensional subspace of $\mathbb{R}^n$.

Next we focus on $\mathcal{L}_m = \mathcal{K}_m(A^T, r_0)$ as subspace of constraints. The basis for the derivation of algorithms for this special choice of $\mathcal{L}_m$ is Lanczos biorthogonalization procedure for $\mathcal{K}_m(A, v_1)$ and $\mathcal{K}_m(A^T, w_1)$:

Algorithm 5.2 (Lanczos biorthogonalization procedure).

```
Choose two vectors $v_1, w_1$ such that $(v_1, w_1) = 1$;
Set $\beta_1 = \delta_1 \equiv 0$, $v_0 = w_0 \equiv 0$;
For $j = 1, 2 \ldots, m$ Do:
    $\alpha_j = (Av_j, w_j)$;
    $\hat{v}_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}$;
    $\hat{w}_{j+1} = A^T w_j - \alpha_j w_j - \delta_j w_{j-1}$;
    $\delta_{j+1} = |(\hat{v}_{j+1}, \hat{w}_{j+1}|^{1/2}$; If $\delta_{j+1} = 0$ Stop;
    $\beta_{j+1} = (\hat{v}_{j+1}, \hat{w}_{j+1})/\delta_{j+q}$;
    $w_{j+1} = \hat{w}_{j+1}/\beta_{j+1}$;
    $v_{j+1} = \hat{v}_{j+1}/\delta_{j+1}$;
EndDo
```

Assuming this algorithm does not break down before step m it is possible to prove (cf. [48]) that the vectors v_i, w_j $(i, j = 1, \ldots, m)$ form a biorthogonal system, i.e.

$$(v_i, w_j) \;=\; \delta_{ij} \qquad 1 \le i, j \le m. \tag{5.15}$$

Additionally $\{v_i\}_{i=1,\dots,m}$ is a basis of $\mathcal{K}_m(A, v_1)$, $\{w_i\}_{i=1,\dots,m}$ is a basis of $\mathcal{K}_m(A^T, w_1)$, and the relation

$$W_m^T A V_m = T_m \tag{5.16}$$

holds with

$$T_m = \begin{pmatrix} \alpha_1 & \beta_2 & & & \\ \delta_2 & \alpha_2 & \beta_3 & & \\ & . & . & . & \\ & & \delta_{m-1} & \alpha_{m-1} & \beta_m \\ & & & \delta_m & \alpha_m \end{pmatrix},$$

for $\alpha_i, \delta_i, \beta_i$ as calculated in step 4,7,8 of Algorithm 5.2. Lanczos' procedure is the idea behind two different Krylov methods with $\mathcal{K} = \mathcal{K}_m(A, r_0/\beta)$ and $\mathcal{L} = \mathcal{K}_m(A^T, r_0/\beta)$:

(i) The Biconjugate Gradient Algorithm (BCG). Using equation (5.13), (5.14), (5.15), and (5.16) the approximate solution x_m can be expressed as

$$x_m \;=\; x_0 + V_m T_m^{-1}(W_m^T r_0) \;=\; x_0 + V_m T_m^{-1}(\beta e_1),$$

where e_1 denotes the first unit vector in $\mathbb{R}^m$. The algorithm performs a successive LU-decomposition of T_m (cf. [48]). This permits the successive calculation of all intermediate steps x_k and r_k $(k = 1, \dots, m)$. If the required tolerance for the approximate solution is fulfilled, the algorithm stops.

(ii) The Quasi-Minimal Residual solvers (QMR). Using Algorithm 5.2 the residual can be written as (cf. [48])

$$r_m \;=\; V_{m+1}(\beta e_1 - \overline{T}_m y), \tag{5.17}$$

where $\overline{T}_m$ is the $(m-1)\times m$ tridiagonal matrix with

$$\overline{T}_m = \begin{pmatrix} T_m \\ \delta_{m+1} e_m^T \end{pmatrix}.$$

The idea of QMR methods is to minimize in every iteration the function

$$J(y) \equiv \|\beta e_1 - \overline{T}_m y\|_2$$

instead of the real residual as given in equation (5.17). However, this is still reasonable and the resulting solution is called the quasi-minimal residual approximation.

A big disadvantage of these methods is that convergence is not ensured although in practice they mostly turn out to be robust and efficient. Even if roundoff errors are neglected they rely on Lanczos' Biorthogonalization procedure 5.2, where so called serious breakdowns may occur which cannot be prevented (cf. [48]).

5.2.3 TFQMR solver

The Lanczos biorthogonalization process calculates the basis of $\mathcal{K}_m(A, v_1)$ and $\mathcal{K}_m(A^T, w_1)$ simultaneously. Thus, the two methods BCG and QMR do not only provide the approximate solution of the problem $Ax = b$ but also of its dual $A^T x = b$. However, we are not interested in this dual problem and the calculation of the matrix-vector product with the transposed of the matrix only causes unnecessary computational costs. Looking carefully at the algorithms, it turned out that the basis of $\mathcal{K}_m(A^T, w_1)$ is not directly needed for the calculation of the solution x_m. It is only used for the calculation of coefficients in the matrix T_m.

In 1989 Sonneveld [50] developed the Conjugate Gradient Squared algorithm (CGS) based on the BCG avoiding the use of A^T. Thus, he gained faster convergence for roughly the same computational costs. A variant of his algorithm is the BiConjugate Gradient Stabilized algorithm (BiCGStab). In 1993 Freund [20] derived a transpose-free variant of the QMR algorithms. Following the already transposed-free algorithm of Sonneveld's CGS he used the idea of QMR algorithms not to minimize the real residual but a quasi-residual. For the derivation of the algorithm we refer to [20] or [48].

Algorithm 5.3 (Transpose-free QMR).

```
1. Compute $w_0 = u_0 = r_0 = b - Ax_0$, $v_0 = Au_0$;
2. Set $d_0 = 0$, $\tau_0 = \|r_0\|_2$, $\theta_0 = \eta_0 = 0$;
3. Choose $r_0^\star$ such that $\rho_0 \equiv (r_0^\star, r_0) \neq 0$;
4. For $m = 0, 1, 2, \ldots,$ until convergence Do:
5.    If $m$ is even Then:
6.       Set $\alpha_{m+1} = \alpha_m = \rho_m/(v_m, r_0^\star)$;
7.       Set $u_{m+1} = u_m - \alpha_m v_m$;
8.    EndIf
9.    Set $w_{m+1} = w_m - \alpha_m A u_m$;
10.   Set $d_{m+1} = u_m + (\theta_m^2/\alpha_m)\eta_m d_m$;
11.   Set $\theta_{m+1} = \|w_{m+1}\|_2/\tau_m$, $c_{m+1} = (1 + \theta_{m+1}^2)^{-1/2}$;
12.   Set $\tau_{m+1} = \tau_m \theta_{m+1} c_{m+1}$, $\eta_{m+1} = c_{m+1}^2 \alpha_m$;
13.   Set $x_{m+1} = x_m + \eta_{m+1} d_{m+1}$;
14.   If $m$ is odd Then:
15.      Set $\rho_{m+1} = (r_{m+1}, r_0^\star)$, $\beta_{m-1} = \rho_{m+1}/\rho_{m-1}$;
16.      Set $u_{m+1} = w_{m+1} + \beta_{m-1} u_m$;
17.      Set $v_{m+1} = A u_{m+1} + \beta_{m-1}(A u_m + \beta_{m-1} v_{m-1})$;
18.   EndIf
19. EndDo
```

5.2.4 Comparison of GMRES, BiCGStab, and TFQMR

The Krylov solvers GMRES, BiCGStab, and TFQMR are suited for the solution of the linear probem arising in the Newton algorithm 5.1 in step 4 since they all apply to non-symmetric non-definite matrices. To ascertain which method works best for our linear system we made a comparison with respect to the evolution of the residual (see Figure 5.2) and the computational cost (see Table 5.1).
Looking at Figure 5.2 we observe that GMRES(25) shows by far the best behavior. The TFQMR solver needs many iterations more to reach the required tolerance whereas the BiCGStab solver does not converge at all. Comparing the computing times in Table 5.1 shows that the GMRES(25) solver is very slow. This is not astonishing since it has to perform a projection into a 25-dimensional subspace of $\mathbb{R}^n$ in each iteration whereas the TFQMR method only performs a one-dimensional projection per iteration. The TFQMR solver shows the best overall behavior and for this reason was chosen as the standard linear solver.

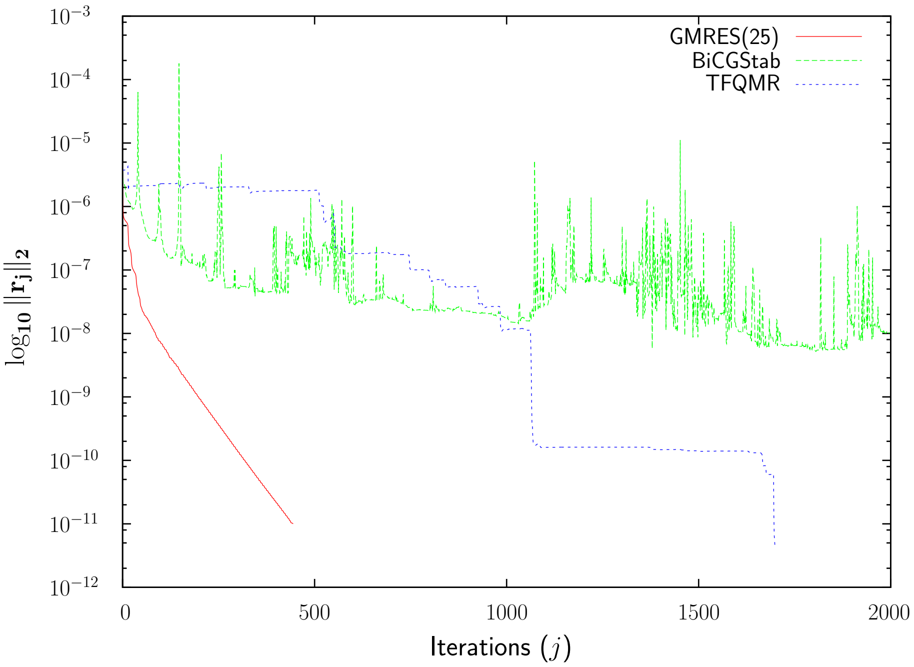


Figure 5.2: Residuals of different Krylov methods for the solution of the linear system in one Newton iteration for a 2-dimensional simulation of our problem with 18387 unknowns. For the CPU usage see Table 5.1.

Krylov method	Iterations	CPU-time	Residual
GMRES(25)	443	$54.63s$	$9.984 \cdot 10^{-12}$
TFQMR	1699	$7.38s$	$4.104 \cdot 10^{-12}$
BiCGStab	2000	$7.26s$	$7.683 \cdot 10^{-8}$

Table 5.1: Comparison of different Krylov methods for the solution of the linear system in one Newton iteration in a 2-dimensional simulation of our problem with 18387 unknowns. The maximum of iterations is set to 2000. For the evolution of the residual see Figure 5.2.

5.3 Preconditioning

For the efficient solution of a given sparse linear system preconditioning is crucial. However, there are not many theoretical results applicable to matrices with unpleasant properties as considered here. Hence the choice of an appropriate preconditioner is difficult. We came to the conclusion that ILU preconditioning techniques generate good results for our problem.

5.3.1 ILU factorization preconditioners

The linear system we have to solve in each Newton iteration does not change very much. Hence if a preconditioner generates a good pseudo inverse of the system matrix this preconditioning matrix may be used in all Newton iterations of one timestep. One method which allows the generation of a preconditioning matrix with user-defined accuracy is the ILU factorization process.

We consider a sparse $n \times n$ matrix A with entries a_{ij}. A general incomplete LU (ILU) factorization process can be described as follows: Compute a sparse lower triangular matrix L and a sparse upper triangular matrix U by a LU factorization process while dropping some of the obtained elements in L and U. The LU factorization can be done, for instance, by Gaussian elimination. We use the so called IKJ variant of the algorithm (cf. [48]) as basis for the ILU factorization process since it processes each row of the matrix successively instead of each column. This is more appropriate since most data structures in scientific computing are row-oriented, e.g. the CSR format for the efficient storage of sparse matrices which we employ in our algorithm.

Now let P be any zero pattern set with excluded diagonal, i.e.

$$P \subset \left\{ (i,j) \mid i \neq j,\, 1 \leq i,j \leq n \right\}.$$

The ILU factorization is nothing else then a modified LU factorization of the matrix A.

Algorithm 5.4 (General ILU factorization, IKJ Version).

```
For $i = 2, \dots, n$ Do:
    For $k = 1, \dots, i-1$ and if $(i,k) \notin P$ Do:
        Set $a_{ik} = a_{ik}/a_{kk}$;
        For $j = k+1, \dots, n$ and for $(i,j) \notin P$ Do:
            Set $a_{ij} = a_{ij} - a_{ik}a_{kj}$;
        EndDo
    EndDo
EndDo
```

There are many different possibilities for the choice of the zero pattern P. Perhaps the best known variant is the zero fill-in ILU, denoted by ILU(0). In this case P is identical to the zero pattern of the sparse matrix A. However, the accuracy of this method is often insufficient to yield an adequate rate of convergence. To overcome this problem additional entries for the pseudo inverse are required. So ILU(p) was developed (cf. [48]) which allows additional entries. However, this method has still some disadvantages. First, the number of additional elements can not be estimated. Second, the decision if an entry is set to zero or not depends only on the structure of A and thus the algorithm is blind to the magnitude of the dropped value.

5.3.2 The ILUT approach

As we have seen in the last section a method is required which allows a good estimate for the size of the factors L and U and furthermore drops small while keeping large values emerging in the LU factorization process. Based on the IKJ variant of the general ILU (cf. Algorithm 5.4) we present a general ILUT approach which provides for variable dropping rules. These rules mean that if a certain set of criteria is fulfilled the concerned element is set to zero. In the following algorithm, which to be found in [48], $a_{i\star}$ denotes the i-th row of the matrix A.

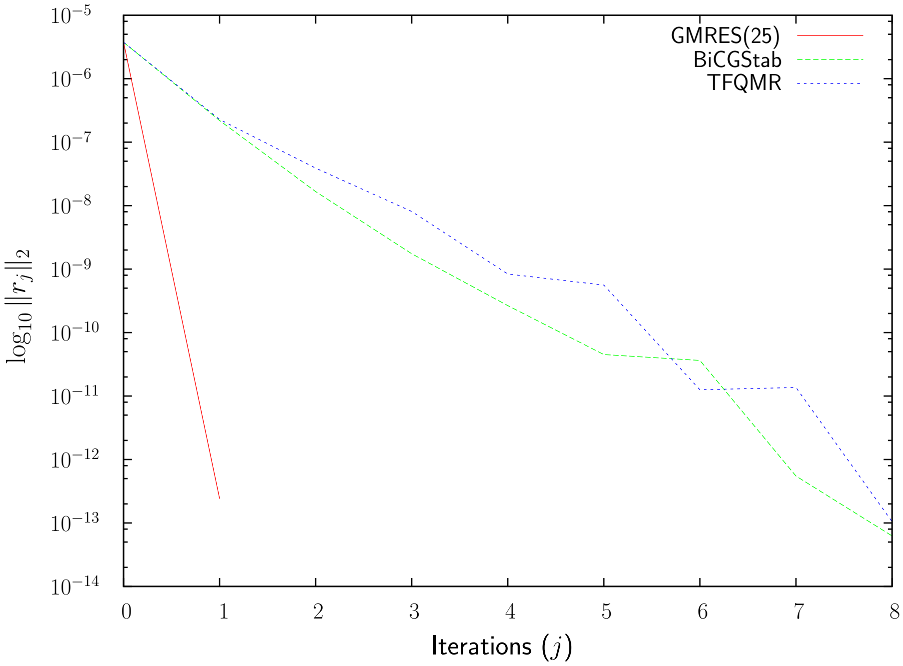


Figure 5.3: Residuals of different Krylov methods for the solution of the preconditioned (with ILUT(10,10^{-4})) linear system in one Newton step for a 2-dimensional simulation of our problem with 18387 unknowns. For the related CPU usage look at Table 5.2.

Krylov method	Iterations	CPU-time (Precon)	Residual
GMRES(25)	1	$4.44s$ $(4.35s)$	$2.409 \cdot 10^{-13}$
TFQMR	8	$4.45s$ $(4.35s)$	$1.052 \cdot 10^{-13}$
BICGSTAB	8	$4.44s$ $(4.35s)$	$6.250 \cdot 10^{-14}$

Table 5.2: Comparison of different Krylov methods for the solution of the preconditioned linear system in one Newton step for a 2-dimensional simulation of our problem with 18387 unknowns. As preconditioner ILUT(10,10^{-4}) is used. Its setup always needs $4.35s$ and thus almost the whole CPU-time of the solution process. For the related residual development see Figure 5.3.2.

Algorithm 5.5 (ILUT).

```
1.  For $i = 1, \dots, n$ Do:
2.     Set $w = a_{i\star}$;
3.     For $k = 1, \dots, i-1$ and when $w_k \neq 0$ Do:
4.        Set $w_k = w_k / a_{kk}$;
5.        Apply a dropping rule to $w_k$;
6.        If $w_k \neq 0$ Then:
7.           Set $w = w - w_k * u_{k\star}$;
8.        EndIf
9.     EndDo
10.    Apply a dropping rule to row $w$;
11.    Set $l_{ij} = w_j$ for $j = 1, \dots, i-1$;
12.    Set $u_{ij} = w_j$ for $j = i, \dots, n$;
13.    Set $w = 0$;
14. EndDo
```

The following dropping rules in line 5 and 10 are used by our algorithm:

- In line 5 an element w_k is dropped if it is less than a relative tolerance tol_i, obtained by multiplying tol by the original norm of the i-th row.
- In line 10 an element in the row is dropped if it is less than the tolerance tol_i. Afterwards only the l_i largest elements in L are kept, where l_i is the sum of the original elements in the i-th row of A in the lower diagonal part plus the fill-in p. For the factor U we proceed accordingly.

This method is denoted by ILUT(p,tol). The number of entries in L and U can easily be estimated as the number of entries in A plus two times p. This is a strict upper bound.

The disadvantage of this algorithm is that for the extraction of the largest elements every row must be sorted. For a large algebraic system this causes a lot of computational costs. Hence for 3-dimensional calculations the setting up of the preconditioner requires a major part of the whole computational time. Nevertheless the benefit is greater. For 2-dimensional calculations it is already faster to set up the preconditioner and solve the system than solving the unpreconditioned system (cf. Table 5.1 and 5.2). And since the same pseudo inverse is used for all Newton steps and often for several following timesteps (sometimes more than one hundred) the overall speedup is enormous, also for 3-dimensional simulations. Furthermore, the use of the preconditioner in practices ensures the convergence of the fast TFQMR and BiCGStab solvers. The development of the residuals for GMRES, BiCGStab, and TFQMR is shown in Figure 5.3.2, the iteration number and the calculation time in Table 5.2. The iteration numbers for all solvers are very low compared to the numbers for the unpreconditioned system (cf. Table 5.1). Note, that with the use of the same pseudo inverse for following timestep the iteration numbers increase again. This can be used as a criterion: If the current iterations are higher than the initial iterations times a factor a new pseudo inverse must be calculated.

Chapter 6

Numerical results

Before we embark on the presentation of numerical results the reliability of our algorithms must be shown. For this purpose we implement test problems and perform convergence tests for the discretizations. Afterwards, the two Algorithms 4.4 and 4.6 are compared.

For the presentation of the numerical results we consider three different situations. First, a calculation with constant temperature at the outside of the melting pot and without gravity is shown. This would be the situation in outer space where no convection occurs. Second, gravity is included and convection takes place due to the solutal effects in the melt. Third, we perform simulations with gravity and an external temperature field as in the real process. The results of these simulations are compared afterwards and the final conclusions about the modeling of the process is enclosed.

All the programming in this work is done within the finite element toolbox ALBERTA developed by Prof. A. Schmidt, University of Bremen, and Prof. K.G. Siebert, University of Augsburg (cf. [49]). It is a free powerful package providing tools for assembling and solving the discrete system as well as support for input and output of data and online graphics. Further development is still done mainly at the University of Augsburg to extend the functionality. For detailed information about ALBERTA and for a download link please visit the web-page www.alberta-fem.de.

6.1 Convergence tests

In this section we define the errors for the test problems and introduce the experimental order of convergence (EOC). Then, we analyze the two discretizations as given by Algorithm 4.4 and 4.6. A comparison between the two approaches is appended.

For the convergence tests of the algorithms we need to calculate the error between the exact and the discrete solution of our problem. We now define these error terms. Note, that for the moving interface approach 4.4 it is necessary to take the different motions of the continuous and the discrete domains into account.

Definition 6.1 (Error for the moving interface approach).
Let $(\hat{\theta}, \hat{c}_1, \hat{c}_2, \hat{v}_\Gamma)$ *be the exact solution of Problem 3.9 on the fixed reference domain* $\hat{\Omega}_h = \Omega_h(0)$*. Let* $(\theta_h^\tau, c_{1,h}^\tau, c_{2,h}^\tau, v_{\Gamma,h}^\tau)$ *be the discrete solutions of Algorithm 4.4 and* Φ *the parameterization as defined in equation* (4.9)*. Then the error of temperature and concentration fields is defined as*

$$\begin{aligned} Err^{\tau,h}_{\Omega_h,\Phi} \; &= \; \Big(\int_0^T \|\hat{\theta} \circ \Phi^{-1} - \theta_h^\tau\|^2_{H^1(\Omega_h(t))} \, dt \\ &\quad + \sum_{i=1}^{2} \int_0^T \|\hat{c}_i \circ \Phi^{-1} - c_{i,h}^\tau\|^2_{\mathcal{H}^1(\Omega_h^\bullet(t))} \, dt \Big)^{\frac{1}{2}}. \end{aligned}$$

The relative error of the interface velocity is defined as

$$Err^{\tau,h;p}_{\Gamma_h,\Phi} \; = \; \frac{\operatorname{ess\,sup}_{t\in(0,T)} \|\hat{v}_\Gamma \circ \Phi^{-1} - v_{\Gamma,h}^\tau\|_{L^p(\Gamma_h(t))}}{\operatorname{ess\,sup}_{t\in(0,T)} \|v_\Gamma\|_{L^p(\Gamma_h(t))}}, \quad p = 2, \infty.$$

For the fixed interface approach 4.6 we additionally define an error for the flow velocity $\boldsymbol{u}$ and the pressure p of the Navier-Stokes equations (2.7).

Definition 6.2 (Error for the fixed interface approach).
Let $(\theta, c_1, c_2, v_\Gamma)$ *be the exact solution of Problem 3.11 and* $(\theta_h^\tau, c_{1,h}^\tau, c_{2,h}^\tau, v_{\Gamma}{}_h^\tau)$ *the discrete solutions of Algorithm 4.6 as defined in equation* (4.20)*. The error of temperature and concentration fields on the calculation domain* Ω_h *is defined*

as

$$Err_{\Omega_h}^{\tau,h} = \Big(\|\theta - \theta_h^\tau\|^2_{L^2(0,T;H^1(\Omega_h))} + \sum_{i=1}^{2} \|c_i - c_{i,h}^\tau\|^2_{L^2(0,T;\mathcal{H}^1(\Omega_h^\bullet))} \Big)^{\frac{1}{2}}.$$

The relative error of the interface velocity is defined as

$$Err_{\Gamma_h}^{\tau,h;p} = \frac{\|v_\Gamma - v_{\Gamma,h}^\tau\|_{L^\infty(0,T;L^p(\Gamma_h))}}{\|v_\Gamma\|_{L^\infty(0,T;L^p(\Gamma_h))}}, \quad p = 2, \infty.$$

The error of the Navier-Stokes equations between $(\boldsymbol{u}, p)$ *and* $(\boldsymbol{u}_h^\tau, p_h^\tau)$ *is defined as*

$$Err_{flow}^{\tau,h} = \Big(\|\boldsymbol{u} - \boldsymbol{u}_h^\tau\|^2_{L^2(0,T;H^1(\Omega_h^\ell))} + \|p - p_h^\tau\|^2_{L^\infty(0,T;L^2(\Omega_h^\ell))} \Big)^{\frac{1}{2}}.$$

To analyze if the error of a test problem converges with the expected order we next define the experimental order of convergence (EOC). Due to the approximation order of finite element spaces (cf. [9]) we can obtain an error reduction (e.g. for the error of temperature and concentrations) of at most

$$Err_{\Omega_h}^{\tau,h} \leq C(\tau + h^k), \tag{6.1}$$

for decreasing τ and h. Here k is the order of the finite element space for temperature and concentrations and C is a positive constant independent of τ and h.

Definition 6.3 (Experimental order of convergence).
Let err_1 *be the error of one simulation and* err_2 *the error of the same simulation with halved maximum grid size and coupled time step size* $\tau = ch^k$*, where* c *is a positive constant and* k *represents the order of the finite element space. Then, the experimental order of convergence is defined as*

$$EOC = \frac{\ln(err_1/err_2)}{\ln 2}.$$

Thus, if an error estimation as given by equation (6.1) would hold the numerical experiments should yield $EOC \approx k$ for this error.

Simulations for convergence tests require a lot of computational power. Since simulations with high grid resolutions and small time step sizes must be performed they may need several days to finish, even in two space dimension. In three space dimensions our problem becomes so large that the calculation time on finer grids explodes. For this reason we do without convergence tests in three space dimensions.

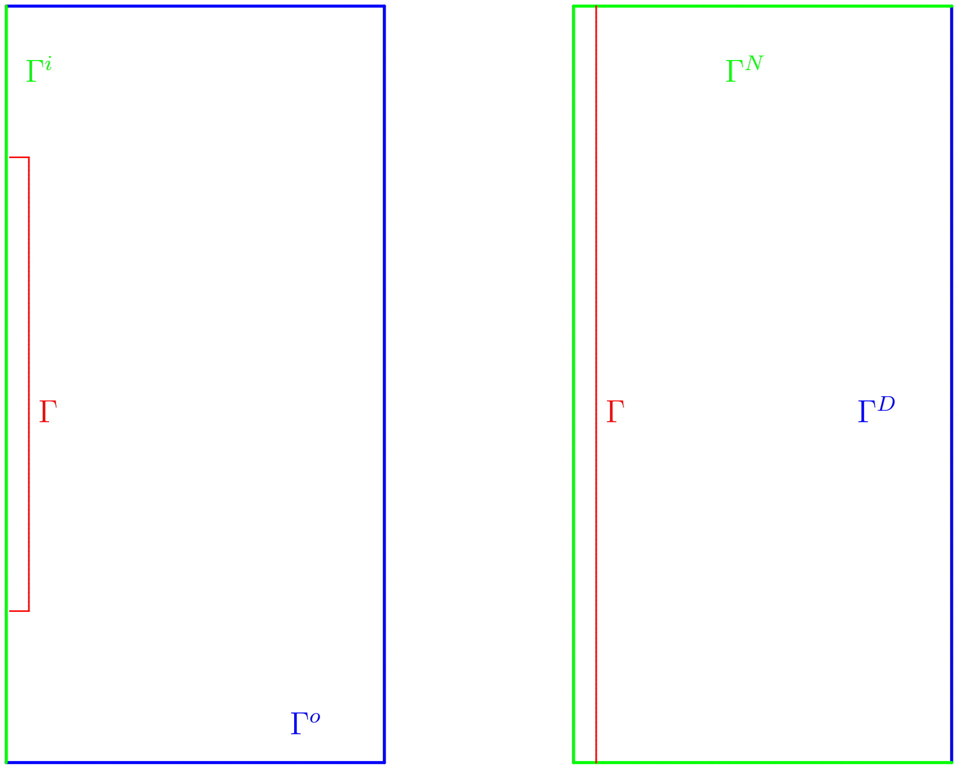


Figure 6.1: 2-dimensional calculation domain Ω_h for the normal problem (left) and the test problems (right) with $\Omega_h = [-0.1, 3.4] \times [0.0, 7.0]$.

The design of test problems for our system of partial differential equations is very complicated since it is rather difficult to fulfill all boundary conditions on the interface Γ. For example the observance of the nonlinear phase diagram equations

(2.32) is only easily possible by requiring temperature and concentrations to be constant on Γ. Even then it is still complicated enough to fulfill the Stefan conditions (2.31). Hence, we decided to simplify the geometry and consider the interface Γ to be a straight line. A sketch of the new geometry is shown in Figure 6.1. Furthermore we adapted the boundary conditions for the outer boundaries. The inner boundary Γ^i where homogeneous Neumann boundary conditions are valid is extended to Γ^N (marked with green color in Figure 6.1) and the outer boundary Γ^o with different boundary conditions for the temperature and the concentrations is reduced to Γ^D (blue color).

For comparing purposes we always perform four runs of each test problem with two different finite element spaces and with and without damping term as introduced in Section 4.5.2. For all runs we start on the same grid with maximum grid size $h_0 = 0.7$ and with time step size $\tau_0 = 1.0$. Then, the same simulation is repeated while successively halving the grid size. Accordingly, the time step size is halved or quartered depending on the order of the finite element space for temperature and concentrations. If the test problem includes the Navier-Stokes equations (2.7) they are always discretized with second order Taylor-Hood elements (cf. Section 4.2.2). A rough description of the four runs is as follows:

1. Linear elements are used for temperature, concentrations, and interface velocity. The time step size for successive simulations is halved.
2. Identical to the first with additional damping term.
3. Quadratic elements are used for temperature and concentration, linear elements are used for the interface velocity. The time step size for successive simulations is quartered.
4. Identical to the third with additional damping term.

6.1.1 Moving interface approach

In this section we analyze a test problem for the moving interface approach as presented in Algorithm 4.4. The Navier-Stokes equations (2.7) are not included in this test. In the convective parts of equation (2.30) we set $\boldsymbol{u} = 0$.

For the construction of the test problem we assume that the solution is constant on straight lines parallel to the interface Γ_h. Thus the problem is independent of the x_2-coordinate.

Example 1 (Traveling wave).
Let $\Omega_h(0) = [-0.1, 3.4] \times [0, 7]$, $\Omega_h^s(0) = [-0.1, 0] \times [0, 7]$, $\Omega_h^\ell(0) = [0, 3.4] \times [0, 7]$, *and* $\Gamma_h(0) = \{\, x \in \Omega_h \,|\, x_1 = 0 \,\}$. *The temperature is given as*

$$\theta(t,x) \;=\; \begin{cases} 0, & x_1 \le 0 \\ \exp\bigl(-\frac{v_\Gamma(t,x)}{\hat{a}}(x_1 - t v_\Gamma(t,x))\bigr) - 1, & x_1 > 0, \end{cases}$$

the concentrations with constant initial values $c_{i,0}^{s,\ell}$ *in* $\Omega_h^{s,\ell}$ *are given as*

$$c_i(t,x) \;=\; \begin{cases} c_{i,0}^s, & x_1 \le 0 \\ c_{i,0}^s + (c_{i,0}^\ell - c_{i,0}^s)\exp\bigl(-\frac{v_\Gamma(t,x)}{\hat{D}_i}(x_1 - t v_\Gamma(t,x))\bigr), & x_1 > 0 \end{cases}$$

for $i = 1, 2$, *the (constant) interface velocity as*

$$v_\Gamma(t,x) \;=\; 2 \cdot 10^{-5}$$

and thus the interface as $\Gamma(t) = \{\, x \in [-0.1, 3.4] \times [0, 7] \,|\, x_1 = 2t \cdot 10^{-5} \,\}$. *The flow velocity* $\boldsymbol{u}$ *is set to zero.*

It is left to the reader to check that the functions $(\theta, c_1, c_2, v_\Gamma)$ as defined in Example 1 are a solution of the equations (2.30), (2.31), and (2.32). On the outer boundary we require different boundary conditions as in equation (2.33). On Γ_N we prescribe homogeneous Neumann boundary conditions for temperature and concentrations, on Γ_D inhomogeneous Dirichlet boundary conditions appropriate to temperature and concentrations as given in Example 1. Note, that the discrete problem is not independent of the x_2-coordinate, since the starting grid

	undamped		damped	
n	$Err^{\tau,h}_{\Omega_h,\Phi}$	EOC	$Err^{\tau,h}_{\Omega_h,\Phi}$	EOC
linear Lagrange finite elements				
0	0.077688		0.080847	
		0.83		0.64
1	0.043720		0.051965	
		1.05		1.15
2	0.021163		0.023450	
		1.02		1.09
3	0.010456		0.011000	
		1.00		1.05
4	0.005233		0.005315	
		0.82		1.02
5	0.002954		0.002615	
quadratic Lagrange finite elements				
0	0.001035		0.001080	
		1.60		1.36
1	0.000341		0.000420	
		2.13		2.19
2	0.000078		0.000092	
		2.03		2.15
3	0.000019		0.000021	
		1.96		2.06
4	0.000005		0.000005	

Table 6.1: Error of temperature and concentration fields in Example 1 for run 1-4, with linear and quadratic elements with and without damping term.

(cf. Figure 6.2 on page 105) is not symmetric.

For this test problem we perform 5 successive iterations of run 1 and 2 and 4 successive iterations of run 3 and 4. The results of all runs are shown in Table 6.1 and 6.2. Looking at Table 6.1 we recognize that the runs with quadratic elements for temperature and concentrations yield the expected experimental convergence order $EOC \approx 2$. In the case of linear elements the convergence order is around one and seems to break down in the last iteration for the undamped discretization. If we consider the errors of the interface velocity in run 1 (cf. Table 6.2) we

	undamped		damped	
n	$Err^{\tau,h,2}_{\Gamma_h,\Phi}/Err^{\tau,h,\infty}_{\Gamma_h,\Phi}$	EOC	$Err^{\tau,h,2}_{\Gamma_h,\Phi}/Err^{\tau,h,\infty}_{\Gamma_h,\Phi}$	EOC
linear Lagrange finite elements				
0	0.018061/0.115761		0.009262/0.046068	
1	0.021183/0.232423	-0.23/-1.01	0.006855/0.042441	0.43/0.12
2	0.006525/0.105552	1.70/1.14	0.003272/0.010256	1.07/2.05
3	0.002266/0.049754	1.53/1.09	0.001671/0.003479	0.97/1.56
4	0.000764/0.035648	1.57/0.48	0.000826/0.001414	1.02/1.30
5	0.000962/0.063108	-0.33/-0.82	0.000366/0.000597	1.17/1.24
quadratic Lagrange finite elements				
0	0.000189/0.001259		0.000115/0.000639	
1	0.000087/0.000965	1.11/0.38	0.000050/0.000236	1.24/1.44
2	0.000020/0.000253	2.14/1.93	0.000016/0.000032	1.66/2.86
3	0.000005/0.000062	1.87/2.03	0.000006/0.000008	1.52/2.05
4	0.000002/0.000015	1.43/2.09	0.000002/0.000002	1.36/1.66

Table 6.2: Error of the interface velocity in Example 1 for run 1-4, with linear and quadratic elements with and without damping term.

observe that we indeed loose convergence. Thus, for linear elements without damping term the discretization is not stable. With damping we get perfect convergence. In essence, the damping term is crucial for linear elements whereas for quadratic elements it is not. Nevertheless, it is still very efficient to use the damping term also for quadratic elements since $Err^{\tau,h,\infty}_{\Gamma_h,\Phi}$ is reduced significantly (cf. Table 6.2). Note that the convergence order in Table 6.2 for quadratic elements in iteration 3 and 4 is significantly below 2. This is not remarkable since the absolute error of the discrete interface velocity in these iterations is already far below the tolerance 10^{-10} of the Newton solver.

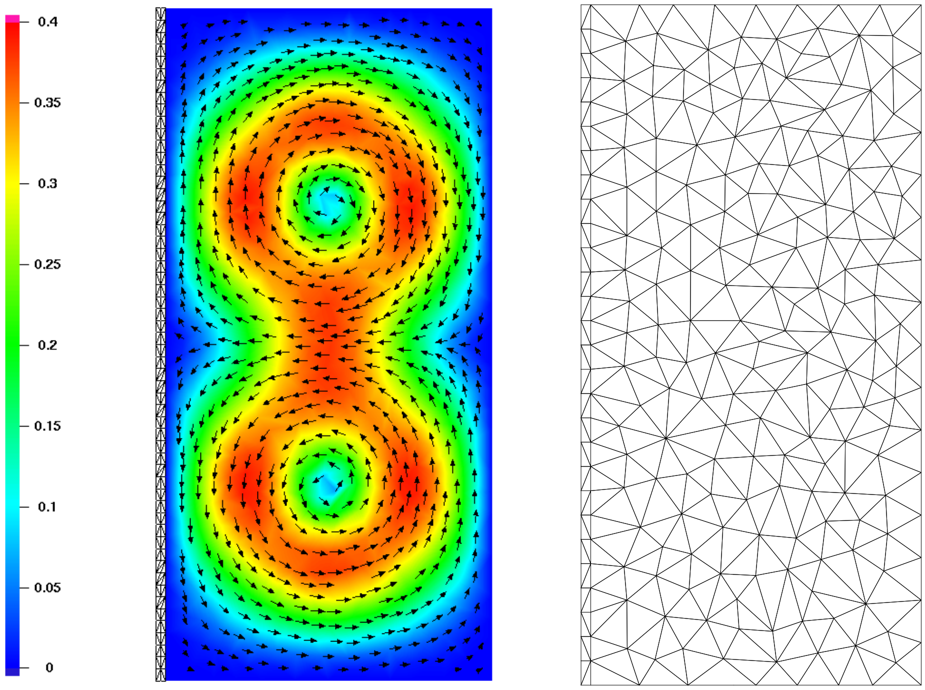


Figure 6.2: Flow field in Example 2 at time $t = 100$ (left) and macro triangulation (right). In the left picture the underlying grid is shown for the solid part Ω^s of the domain. The flow velocity is given in [cm/s].

6.1.2 Fixed interface approach

In this section we analyze a test problem for the fixed interface approach as presented in Algorithm 4.6. The Navier-Stokes equations are now included. Since we are not able to construct a flow for the given right hand side in the Navier-Stokes equations (cf. (2.7)) we decided to construct a flow which is similar to the flow in the real application and calculate the appropriate right hand side. Such a flow is shown in Figure 6.2. Thus, the flow in our test problem is not influenced by the values of temperature and concentrations. However, the flow itself influences the calculation of temperature and concentrations due to the convective terms in their equations (cf. (2.30)).

Again as in the last section we assume that the solution of the test problem is constant on straight lines parallel to the interface Γ_h. The exact solution is given as follows:

Example 2.
Let $\Omega_h = [-0.1, 3.4] \times [0, 7]$, $\Omega_h^s = [-0.1, 0] \times [0, 7]$, $\Omega_h^\ell = [0, 3.4] \times [0, 7]$, *and* $\Gamma_h = \big\{\, x \in \Omega_h \,\big|\, x_1 = 0 \,\big\}$. *The temperature is given as*

$$\theta(t,x) \;=\; \begin{cases} 0, & x_1 \le 0 \\ \exp\big(-\frac{v_\Gamma(t,x)}{\hat{a}} x_1\big) - 1, & x_1 > 0, \end{cases}$$

the concentrations with constant initial values $c_{i,0}^{s,\ell}$ *in* $\Omega_h^{s,\ell}$ *are given as*

$$c_i(t,x) \;=\; \begin{cases} c_{i,0}^s, & x_1 \le 0 \\ c_{i,0}^s + (c_{i,0}^\ell - c_{i,0}^s)\exp\big(-\frac{v_\Gamma(t,x)}{\hat{D}_i} x_1\big), & x_1 > 0 \end{cases}$$

for $i = 1, 2$, *and the interface velocity as*

$$v_\Gamma(t,x) \;=\; v_0 \sin\big(\tfrac{1}{25}t\big)\Big(1 - \cos\big(\tfrac{2\pi}{7}x_2\big)\Big)$$

with a constant $v_0 = 10^{-5}$. *The flow velocity in* Ω_h^ℓ *is given as*

$$\boldsymbol{u}(t,x) \;=\; \Big(1 - \exp\big(-\tfrac{1}{50}t\big)\Big)\,\frac{3}{5}\begin{pmatrix} -\frac{17}{10\pi} f\big(\frac{10\pi}{17}x_1\big) g'\big(\frac{2\pi}{7}x_2\big) \\ \frac{7}{2\pi} f'\big(\frac{10\pi}{17}x_1\big) g\big(\frac{2\pi}{7}x_2\big) \end{pmatrix},$$

where

$$f(s) \;=\; \frac{1-\cos(s)}{2} \quad \text{and} \quad g(s) \;=\; \frac{126\sin(s) - 63\sin(2s) + 112\sin^3(s)}{72\pi},$$

and the pressure as $p(t,x) = 0$.

As already mentioned we get a different right hand side for the first Navier-Stokes equation (2.7), whereas the second equation is fulfilled since the given flow $\boldsymbol{u}$ in Example 2 is divergence free by construction. Also temperature and concentrations do not fulfill equation (2.30) due to the convective terms. Similar to the Navier-Stokes equations we modify (2.30) and require appropriate right hand sides. The boundary conditions (2.31) and (2.32) on the inner boundary Γ are fulfilled. As in the last section we require different boundary conditions on the outer boundary: On Γ_N we prescribe again homogeneous Neumann boundary conditions for temperature and concentrations. On Γ_D inhomogeneous Dirichlet boundary conditions appropriate to temperature and concentrations as given in Example 2 are prescribed.

For the test runs with linear elements we perform 4 successive iterations, for the runs with quadratic elements 3 successive iterations. The results of all runs are shown in Table 6.3, 6.4, and 6.5. First, we consider Table 6.3. The calculation of the flow velocity is independent of the values of temperature and concentrations. Hence, the error $Err_{flow}^{\tau,h}$ is independent of the presence of a damping term J and there is no difference between run 1 and 2 as well as between run 3 and 4. For all runs we use second oder Taylor-Hood elements which maximal yields $EOC \approx 2$. The numerical simulations show the correct convergence order for run 3 and 4. For the run 1 and 2 convergence starts to break down on finer grids. This is reasonable since the time step size for these runs is only halved which actually yields $EOC \approx 1$.

Looking at Table 6.4 we recognize that the error $Err_{\Omega_h}^{\tau,h}$ for temperature and concentration fields in Ω_h shows the expected convergence order for linear ($EOC \approx 1$, run 1 and 2) and quadratic ($EOC \approx 2$, run 3 and 4) elements. Comparing the error between the simulations with and without damping we observe that the error for discretizations with included damping term is slightly higher. On finer grids this difference vanishes. Thus, as expected, the damping term does not harm the convergence of the algorithm.

Comparing the errors of the interface velocity as shown in Table 6.5 is the most interesting and important part. As we have already mentioned in Section 4.5.2 small oscillations in temperature and concentration fields cause strong oscillations of the interface velocity on Γ_h. This effect is shown in Figure 6.3 and 6.4. Hence, for the run with linear elements without damping term $Err_{\Gamma_h}^{\tau,h,\infty}$ in iteration $n = 3$ is still above one, i.e. the maximal difference between discrete and exact solution is larger than the maximal value of the phase velocity. In practice this means unusable results for the discrete interface velocity. The damping term is apparently crucial since it lowers the error decisively without the requirement of

	linear		quadratic	
n	$Err^{\tau,h}_{flow}$	EOC	$Err^{\tau,h}_{flow}$	EOC
0	0.478778	1.39	0.478778	1.41
1	0.182507	2.01	0.180249	2.01
2	0.045310	1.97	0.044615	1.98
3	0.011595	1.92	0.011300	
4	0.003066			

Table 6.3: Error and EOC of the flow velocity for the discretization with second order Taylor-Hood elements in Example 2.

	undamped		damped	
n	$Err^{\tau,h}_{\Omega_h}$	EOC	$Err^{\tau,h}_{\Omega_h}$	EOC
linear Lagrange finite elements				
0	1.251986	0.91	2.256089	0.83
1	0.664574	1.52	1.265357	1.97
2	0.231879	1.05	0.322066	1.50
3	0.111965	0.95	0.113830	0.96
4	0.057818		0.058636	
quadratic Lagrange finite elements				
0	2.283755	2.87	1.860669	2.36
1	0.312435	2.49	0.362020	2.68
2	0.055549	1.94	0.056298	1.96
3	0.014512		0.014518	

Table 6.4: Error of temperature and concentration fields in Example 2 for run 1-4, with linear and quadratic elements with and without damping term.

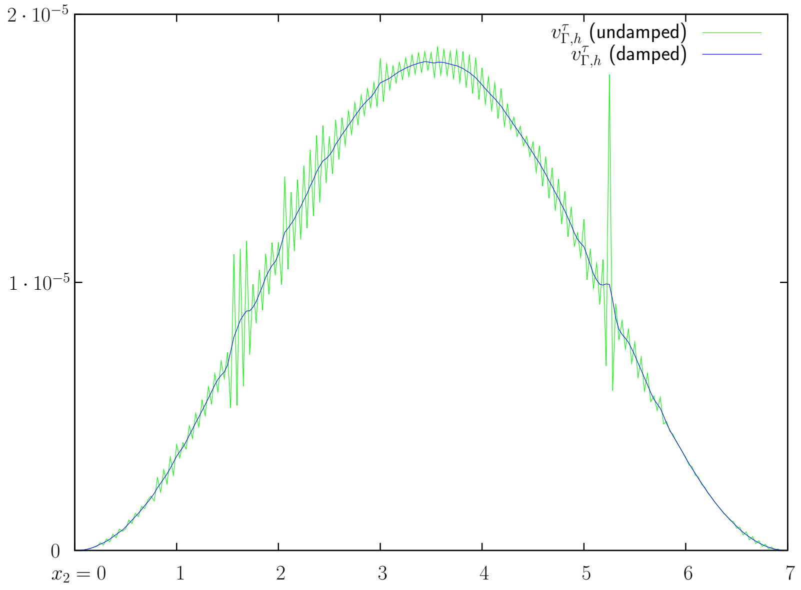


Figure 6.3: Discrete interface velocity in iteration $n = 3$ of run 1 (green) and run 2 (blue) in Example 2 at time $t = 50$.

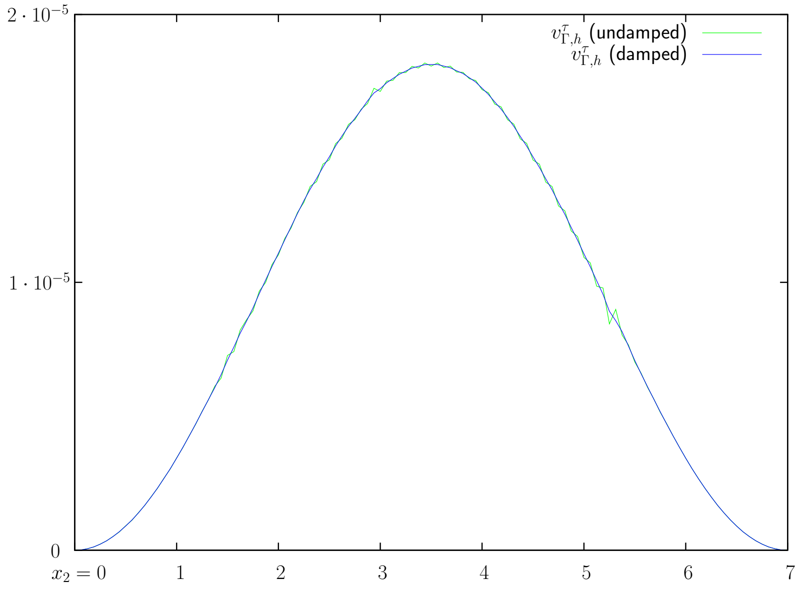


Figure 6.4: Discrete interface velocity in iteration $n = 2$ of run 3 (green) and run 4 (blue) in Example 2 at time $t = 50$.

	undamped		damped	
n	$Err^{\tau,h,2}_{\Gamma_h} / Err^{\tau,h,\infty}_{\Gamma_h}$	EOC	$Err^{\tau,h,2}_{\Gamma_h} / Err^{\tau,h,\infty}_{\Gamma_h}$	EOC
linear Lagrange finite elements				
0	52.73450/166.9586	2.24/1.03	14.16901/20.78887	2.01/1.50
1	11.18194/81.84830	3.38/2.88	3.506300/7.366075	2.87/2.51
2	1.074471/11.14254	2.74/3.44	0.481142/1.289791	1.63/2.89
3	0.160627/1.025683	1.07/3.86	0.155904/0.173491	1.03/1.40
4	0.076598/0.070723		0.076570/0.065572	
quadratic Lagrange finite elements				
0	6.910988/28.61913	3.37/2.54	4.916611/9.269585	3.30/3.32
1	0.670390/4.921356	3.14/4.49	0.500244/0.929274	2.70/3.75
2	0.076273/0.219301	2.00/3.72	0.075747/0.068927	1.99/2.08
3	0.019036/0.016663		0.019038/0.016248	

Table 6.5: Error for the interface velocity in Example 2 for run 1-4, with linear and quadratic elements with and without damping term.

further grid refinements — very important at least for 3d-simulations.

Furthermore the results show that using quadratic elements is very effective. To clarify this we compare two simulations: first, iteration $n = 4$ of run 2, second, iteration $n = 3$ of run 4. Looking at Table 6.4 and 6.5 we observe that all errors of the second simulation are significantly smaller. However, the number of unknowns for temperature, concentrations, and interface velocity in the algebraic system of both simulations is identical ($n_{DOF} = 170941$) since the first simulation uses linear elements for temperature and concentrations on a finer grid and the second quadratic elements on a coarser grid. The time step size is much smaller for the second simulation: $\tau^{quad.}_{n=3} = \frac{1}{64}$ compared to $\tau^{lin.}_{n=4} = \frac{1}{16}$. Nevertheless, the second simulation requires less computational time since the Navier-Stokes equations for the flow velocity are also calculated on a coarser grid.

6.1.3 Comparison of the two approaches

In this section we compare the approaches in Algorithm 4.4 and 4.6. We show that the difference in the results produced by the algorithms is negligible compared to influences of inaccurate material constants.

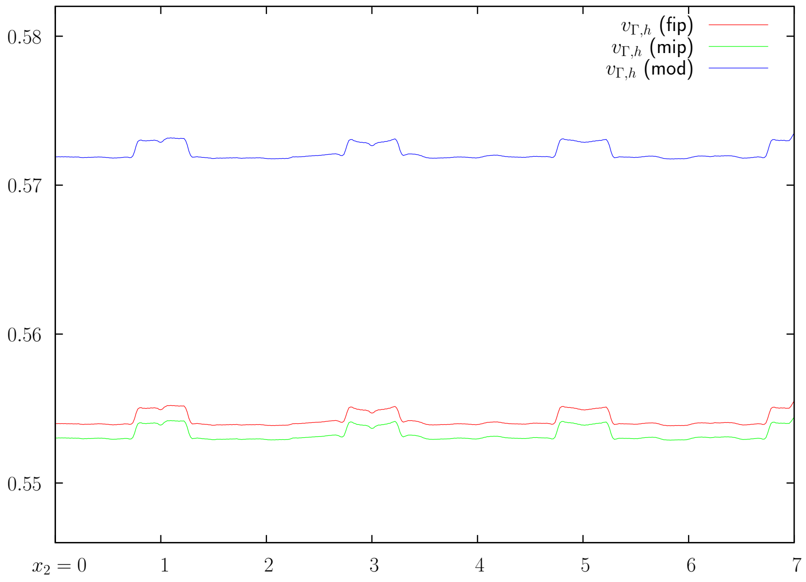
For the comparison we perform an experiment under conditions which are similar to the conditions of the real experiment: We start with a perfectly mixed melt with initial concentrations as in the real experiments. The initial temperature is equal to the melting temperature of the initial concentrations. Only the Navier-Stokes equations are not included. Hence, the flow velocity $\boldsymbol{u}$ in the convective parts in equation (2.30) is set to zero. The above initial values are as given in Chapter 2. (A summarization of them can also be found in the next section.) To trigger growth on the interface Γ_h the external temperature is set to

$$\theta_{ext} = \theta_0 - 0.153 \frac{℃}{min} t, \quad t \in [0, T], \quad T = 30min,$$

which is the same cooling rate as in the real experiment.

For the comparison of the two approaches we perform 3 different 2-dimensional simulations on the same grid as in the previous examples (cf. Figure 6.2). The first is done with the fixed interface approach, the second with the moving interface approach. Then, again the fixed interface approach is used to perform a third simulation with the modified value $\hat{D}_2^\ell = 8.6 \cdot 10^{-5}$ for the diffusion coefficient of tellurium. This is a deviation of only about 5 percent compared to its original value $\hat{D}_2^\ell = 8.2 \cdot 10^{-5}$.

The results of the interface velocity and thickness of the layer in the different simulations after 30 minutes simulated time are shown in Figure 6.5. We can see that the difference between the two approaches with fixed and moving interface is very small. The numerical oscillations – the interface velocity is expected to be constant in space for the above mentioned choice of the external temperature – already cause larger variations. In comparison, the simulation with modified material constant yields a clearly different result. The resulting layer is nearly half a micron thicker than the layer obtained by the simulations with the standard material constant. The explanation of this behavior is that a higher diffusion



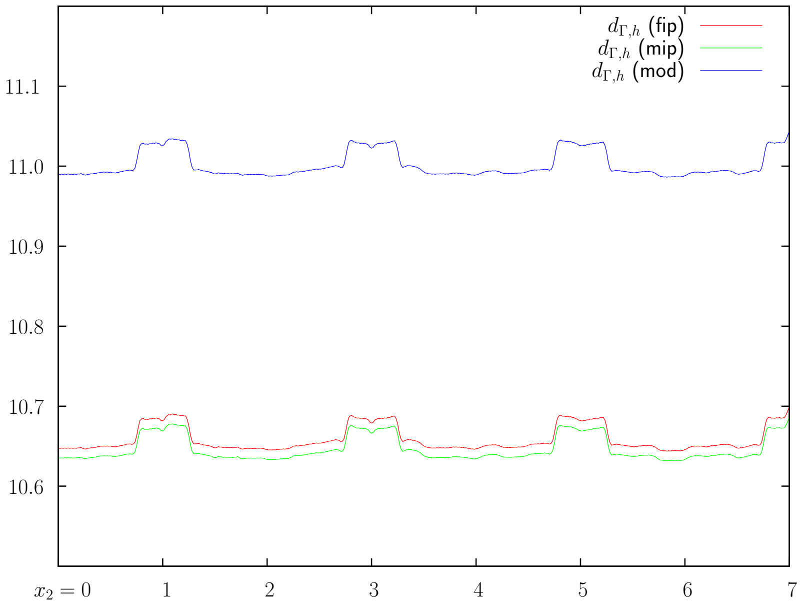


Figure 6.5: Discrete interface velocity (in $[\frac{\mu m}{min}]$, upper picture) and thickness of the layer (in $[\mu m]$, lower picture) calculated with fixed interface approach (red), moving interface approach (green), and with modified parameter $\hat{D}_2^\ell$ (blue) after 30 minutes simulated time.

coefficient of tellurium allows a faster equalization of its concentration and thus enables a higher growth velocity.

We can conclude that small inaccuracies of material constants have a significantly larger influence on the simulation results as enforced by neglecting the interface motion. This justifies the simplification done in the fixed interface approach.

6.2 Simulation results

In this section we present the numerical results for the simulations of the liquid phase epitaxial process. First, the requirements for the homogeneity of the layer are described. Then, different numerical experiments are performed and their results are analyzed.

For all numerical experiments we use the fixed interface approach as presented in Algorithm 4.6 with damping term J as given in equation (4.24) and (4.25). The simulations are always started with a perfectly mixed melt with initial concentrations as in the real production process performed by AIM. The initial concentrations for mercury and tellurium in the melt Ω^ℓ are

$$c^\ell_{1,0} = 0.146377 \qquad \text{and} \qquad c^\ell_{2,0} = 0.846. \tag{6.2}$$

Using the phase diagram equations (2.26a) and (2.26b) in Chapter 2 we calculate the related initial concentrations in the solid phase Ω^s and obtain

$$c^s_{1,0} = 0.404296 \qquad \text{and} \qquad c^s_{2,0} = 0.5. \tag{6.3}$$

This is equivalent to the composition

$$x = 0.191408 \tag{6.4}$$

for the growing crystal layer $Hg_{1-x}Cd_xTe$ (cf. equation (2.25)). The melting temperature related to the concentrations in equation (6.2) is obtained by the phase diagram equation (2.26c),

$$\theta_0 = \theta_{m,0} = 467.85334\,^\circ\!C. \tag{6.5}$$

This value is chosen as initial temperature in the whole domain Ω.

The geometry of the simulations is the halved melting pot or its 2d-section as shown in Figure 3.1 on page 42. The height as well as the diameter of the melting pot is 7cm. All 3-dimensional simulations are performed on the macro triangulation as shown in Figure 6.6 without further refinements. This results in 83 696 unknowns for the Navier-Stokes equations and altogether 752 664 unknowns for temperature, concentrations, and interface velocity. Here we use cubic elements

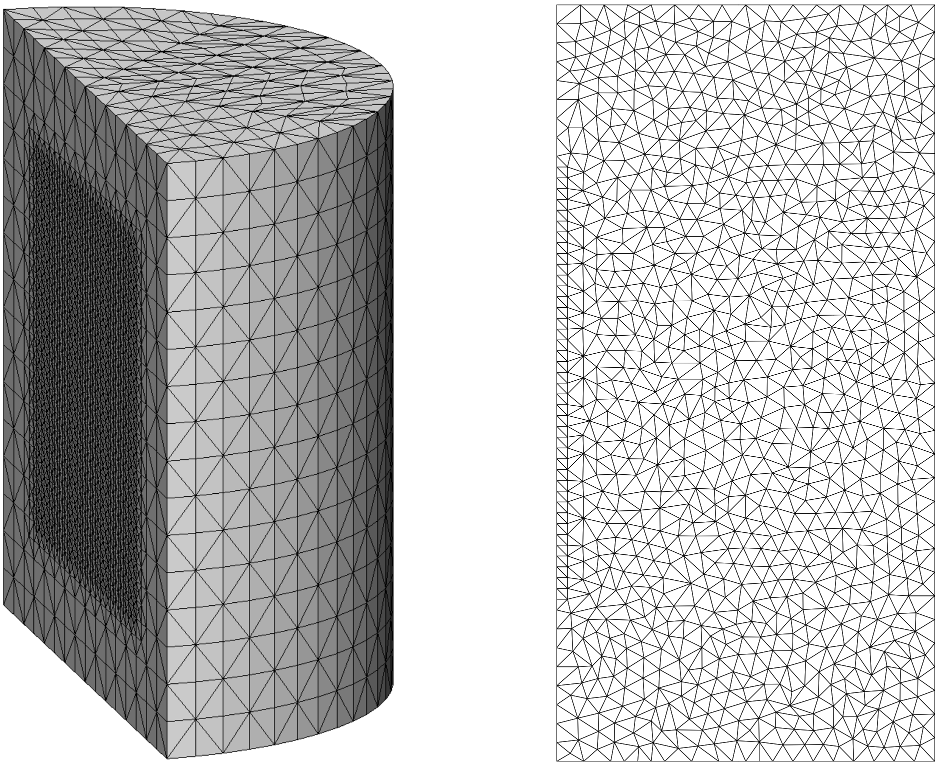


Figure 6.6: Macro grid for the 3-dimensional and the 2-dimensional simulations of the liquid phase epitaxy.

for temperature and concentrations, quadratic elements for the flow and linear elements for pressure and interface velocity. Since at least nearly 10 000 time steps must be calculated a 3-dimensional calculation takes around 2-3 weeks.

For all 2-dimensional simulations we performed several iterations with different refinements of the macro triangulation as shown in Figure 6.6 to ensure that the results are reliable. On the macro triangulation we get 5 056 unknowns for the Navier-Stokes equations and altogether 12 270 unknowns for temperature, concentrations, and interface velocity. Here we use quadratic elements for flow, temperature, and concentrations and linear elements for pressure and interface velocity. A simulation on the macro grid without further refinement only takes around 1.5 hours.

6.2.1 Homogeneity of the film

Large-area detectors with a size of $10 \times 3\,mm$ require a high homogeneity of the composition and the thickness of the layered substrate. For the area of one square centimeter a deviation of about $x = \pm 0.001$ for the composition and of about $d = \pm 2\mu m$ for the thickness should not be exceeded.

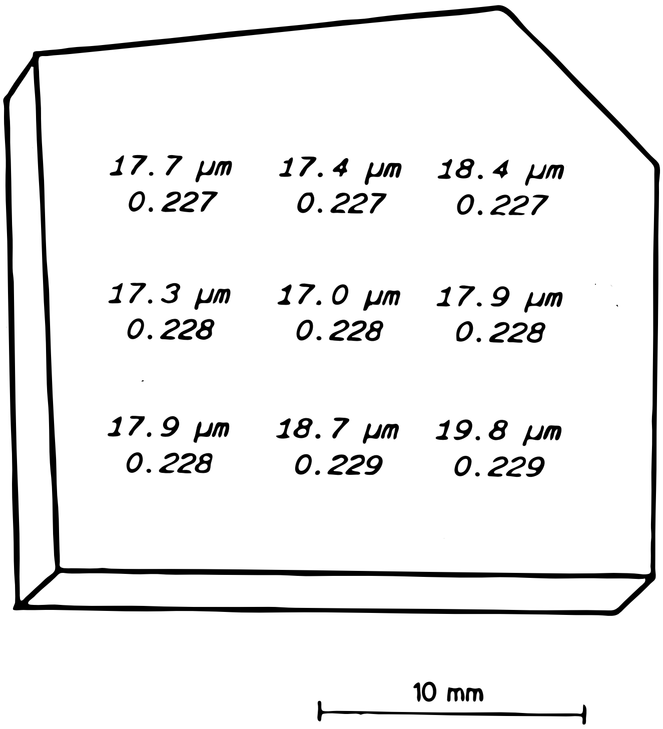


Figure 6.7: Homogeneity of composition ($x = 0.228 \pm 0.001$) and thickness ($d = 18.4 \pm 1.4\mu m$) of a typical epitaxial layer. (Source: AIM)

A typical layer fulfilling these requirements is shown in Figure 6.7. Note, that the value of the composition $x = 0.228$ from AIM does not match well the values obtained from the phase diagram of Harman (cf. Figure 2.2). For an initial concentration as given in equation (6.2) the phase diagram equation (2.25) yields $x \approx 0.191$ (cf. equation (6.4)). And, as we can see in the numerical experiments in the following sections, x tends to decrease during the process.

6.2.2 Experiment I

In our first numerical experiment of the liquid phase epitaxial process we assume that we have no gravity and that at any fixed time t the temperature outside the melting pot is constant. The external temperature decreases in time with a constant ramp

$$\Delta\theta = -0.153 \frac{℃}{min}. \tag{6.6}$$

This yields

$$\theta_{ext} \ = \ \theta_0 + t\Delta\theta. \tag{6.7}$$

The initial values for the temperature and the concentrations are as given by equation (6.2), (6.3), and (6.5). Since we have no gravity no flow in the liquid phase arises. This prevents any mixing due to convection inside the melting pot and the equalization of the concentrations occurs only due to diffusion.

The results of the simulation are shown in Figure 6.8, 6.9, and 6.10. In the 2-dimensional and the 3-dimensional simulation the homogeneity of the grown layer is nearly identical. Since concentration exchange occurs only due to diffusion, the growth velocity of the layer is slow. Thus, obtaining a thickness of about 18-19 microns takes around 42 minutes. The deviation in thickness and concentration along the surface of the layer is very small. The thickness of the layer in the 3-dimensional simulation is $d = 18.75 \pm 0.15 \mu m$, the composition $x = 0.1642 \pm 0.0001$, this being less than 10 percent of the allowed deviations on $1cm^2$ for an area of $4cm^2$. The homogeneity of the composition normal to the surface varies much more as can be seen by the composition of the layer at different times (cf. Figure 6.8 and 6.9). However, this is as expected since the concentrations in the melt near the layer varies due to the growth process yielding a change of composition. To reduce this effect additional mixing of the melt would be required. Furthermore we recognize that the concentration equalization of tellurium occurs more slowly than for mercury due to its smaller diffusion coefficient (cf. Figure 6.10, for the diffusion coefficients confer Table 2.1). Altogether we notice that epitaxial growth in the outer space yields a very homogeneous layer which cannot be achieved under the influence of gravity.

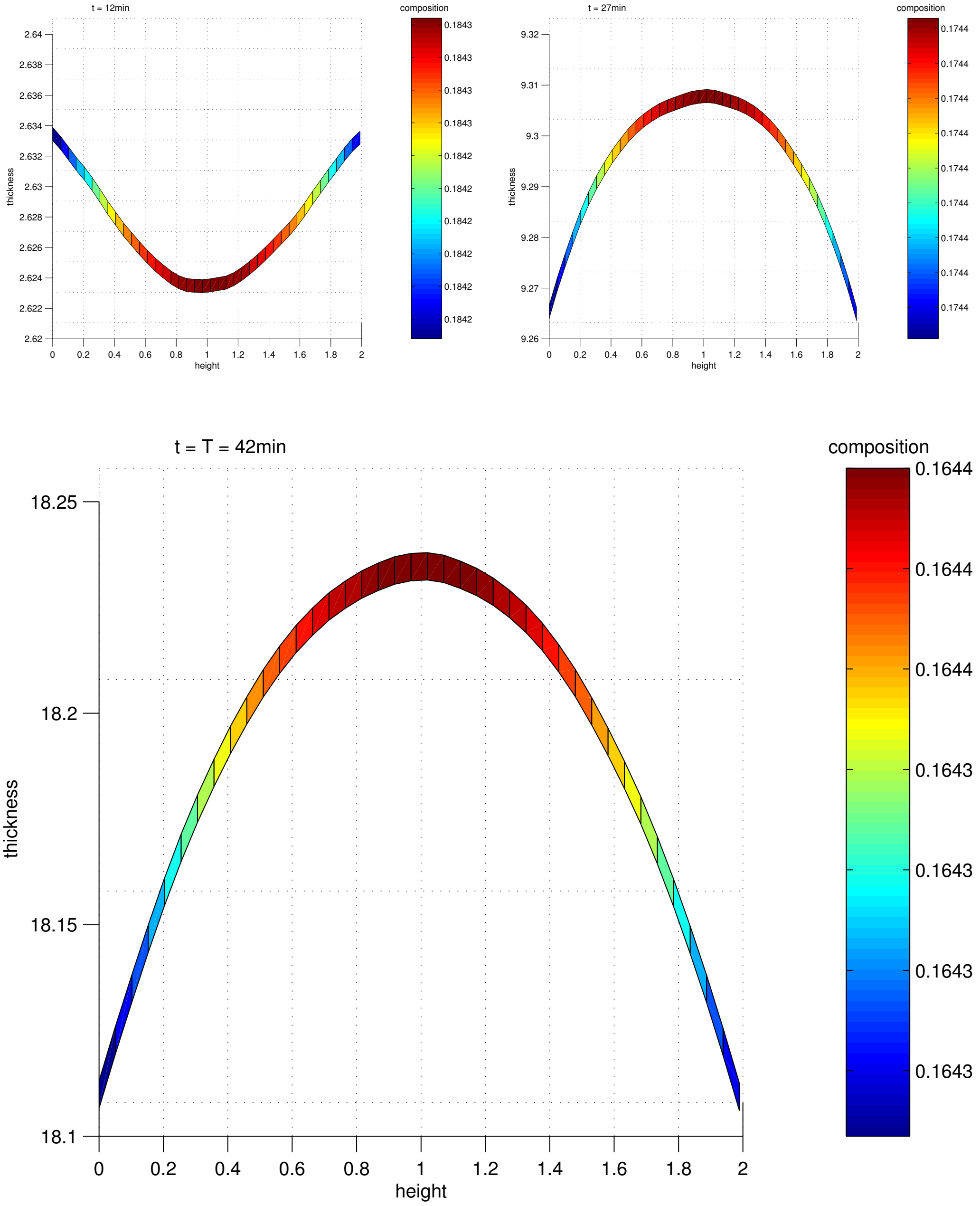


Figure 6.8: Evolution for the homogeneity of the layer at time $t = 12\,min$, $t = 27\,min$, and $t = 42\,min$ in the 2-dimensional simulation of Experiment I.

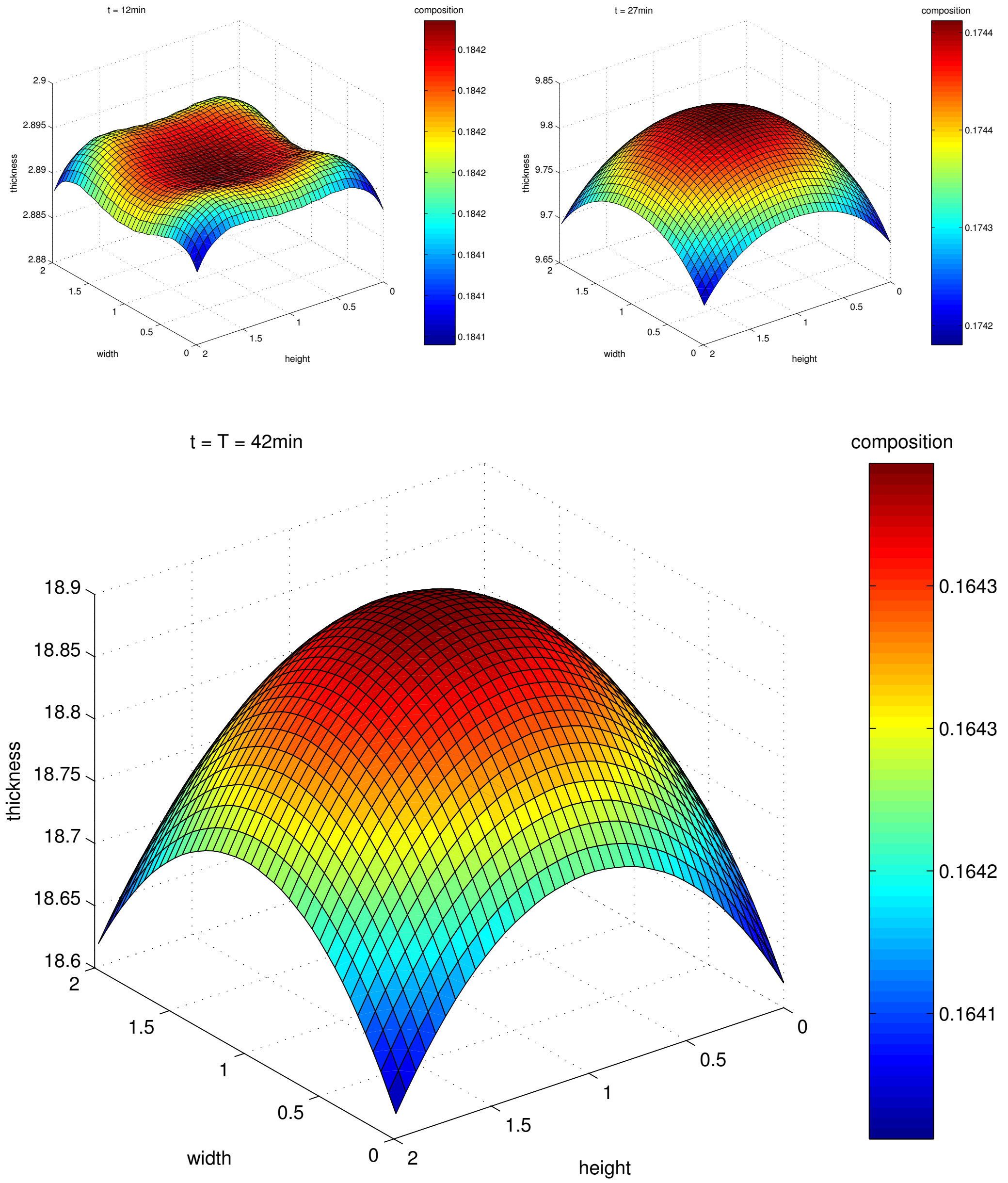


Figure 6.9: Evolution for the homogeneity of the layer at time $t = 12\,min$, $t = 27\,min$, and $t = 42\,min$ in the 3-dimensional simulation of Experiment I.

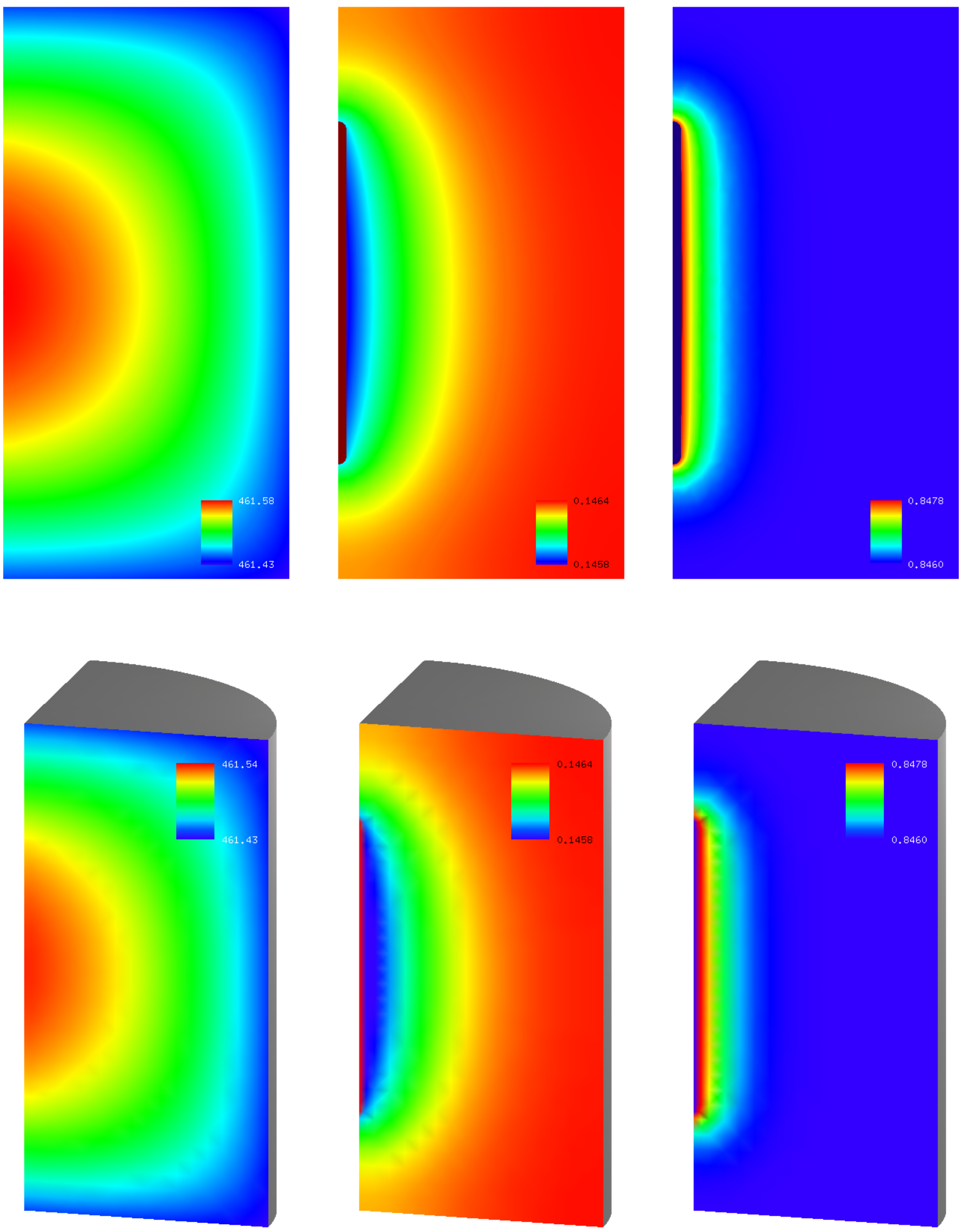


Figure 6.10: Temperature, concentration of mercury, concentration of tellurium, and homogeneity of the layer in the 2-dimensional simulation (above) and the 3-dimensional simulation (below) of Experiment I at the end time $T = 42\,min$.

6.2.3 Experiment II

In contrast to the numerical experiment in the last section the influence of gravity on the process is now included. We still assume a constant spatial temperature field outside the melting pot. Thus, the external temperature is chosen as in the last experiment (cf. equation (6.7)). The arising flow inside the melt is now driven by the variation of concentrations. The initial values of the concentrations and the temperature are as in equation (6.2), (6.3) and (6.5).

The results of the 2-dimensional simulation are shown in Figure 6.11, 6.13, and 6.15, the results of the 3-dimensional simulation in Figure 6.12, 6.14, and 6.16. One can see (cf. Figure 6.11 and 6.12) that the flow velocity is very small (maximal around $0.06\frac{cm}{s}$). Thus, the mixing effect of the flow is not big. Nevertheless we observe that it has a decisive influence on the thickness and composition of the layer (cf. Figure 6.15 and 6.16). The film grows faster at places where the flow equalizes the concentrations. Compared to the simulations without gravity in the last section we obtain the same thickness of the layer in less time. A layer of about 18 microns is already obtained after 28 minutes. Furthermore the homogeneity with respect to the composition in normal direction is much higher (cf. Figure 6.8, 6.9 and 6.15, 6.16). The homogeneity with respect to the thickness of the layer is not as good as in the last section but is still in the required tolerance of $d = \pm 2\mu m$ in one square centimeter. Again as in the last section we obtain a very good accordance between the 2- and the 3-dimensional simulations. The maximal flow velocity as well as the maximal deviation of the concentrations from their initial values are nearly the same. Furthermore we observe that the homogeneity of the layer in the 3-dimensional simulation mainly varies for different heights and not for different widths (cf. Figure 6.16).

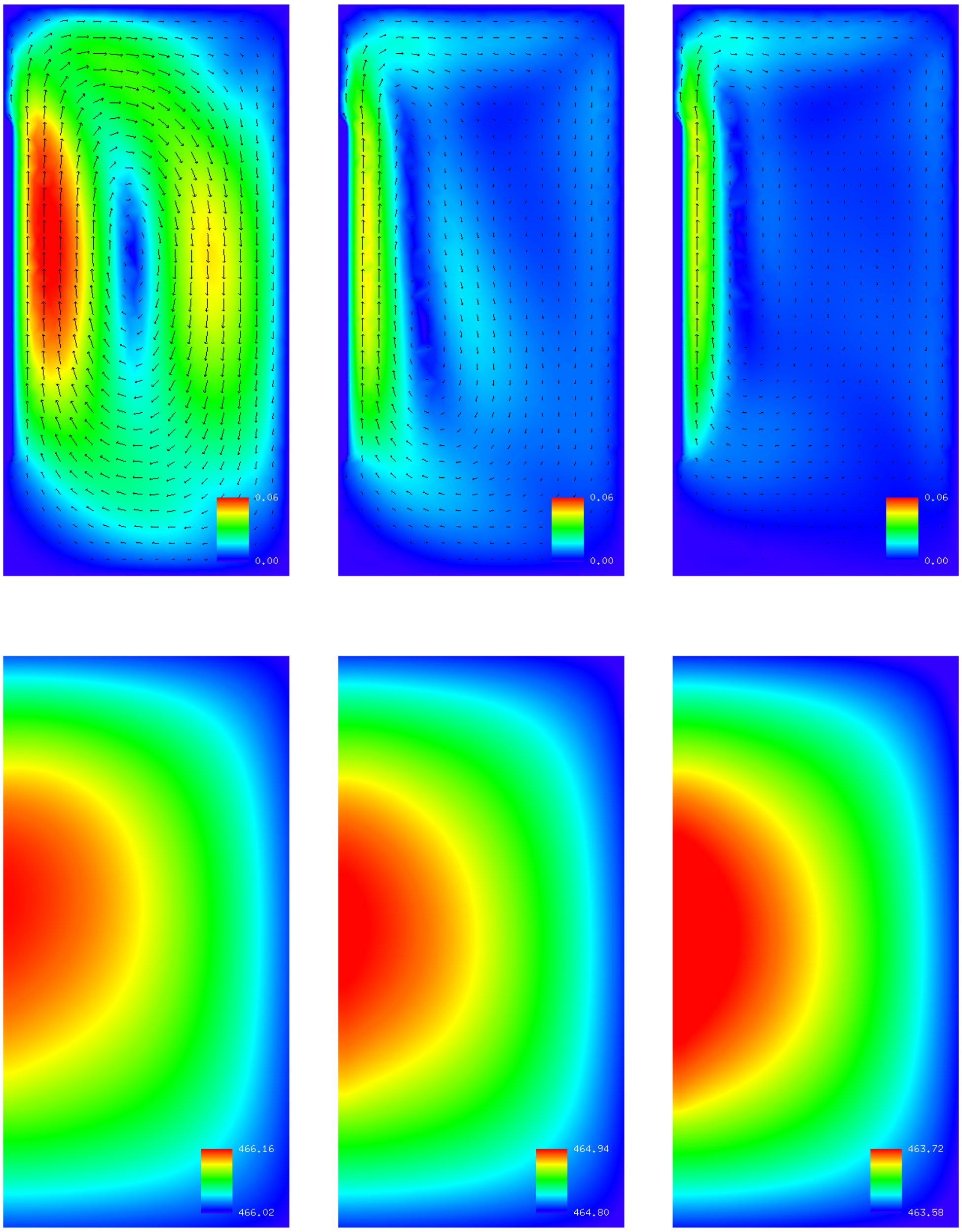


Figure 6.11: Evolution of the flow velocity (above) and the temperature (below) in the 2-dimensional simulation at time $t = 12\,min$, $t = 20\,min$, and $t = 28\,min$ in Experiment II.

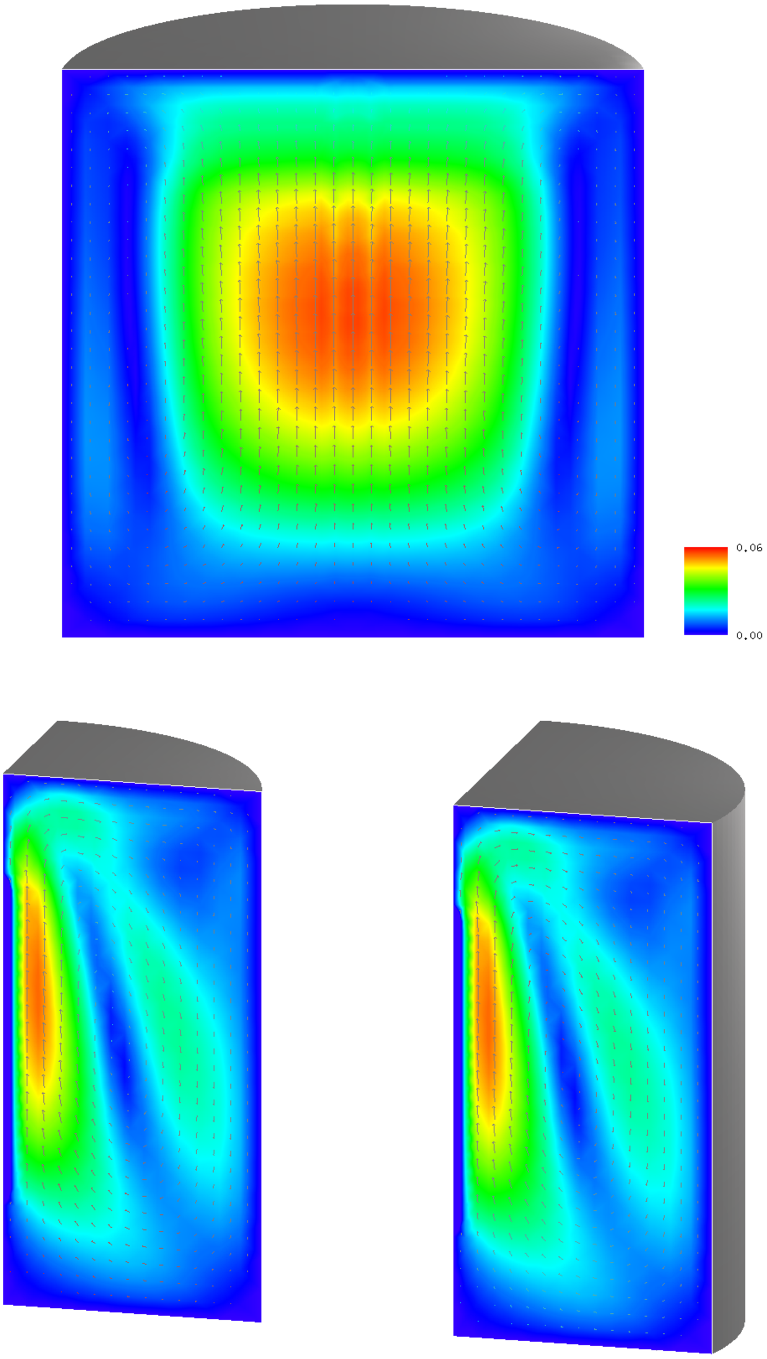


Figure 6.12: Flow velocity in the 3-dimensional simulation at time $t = 12\,min$ in Experiment II.

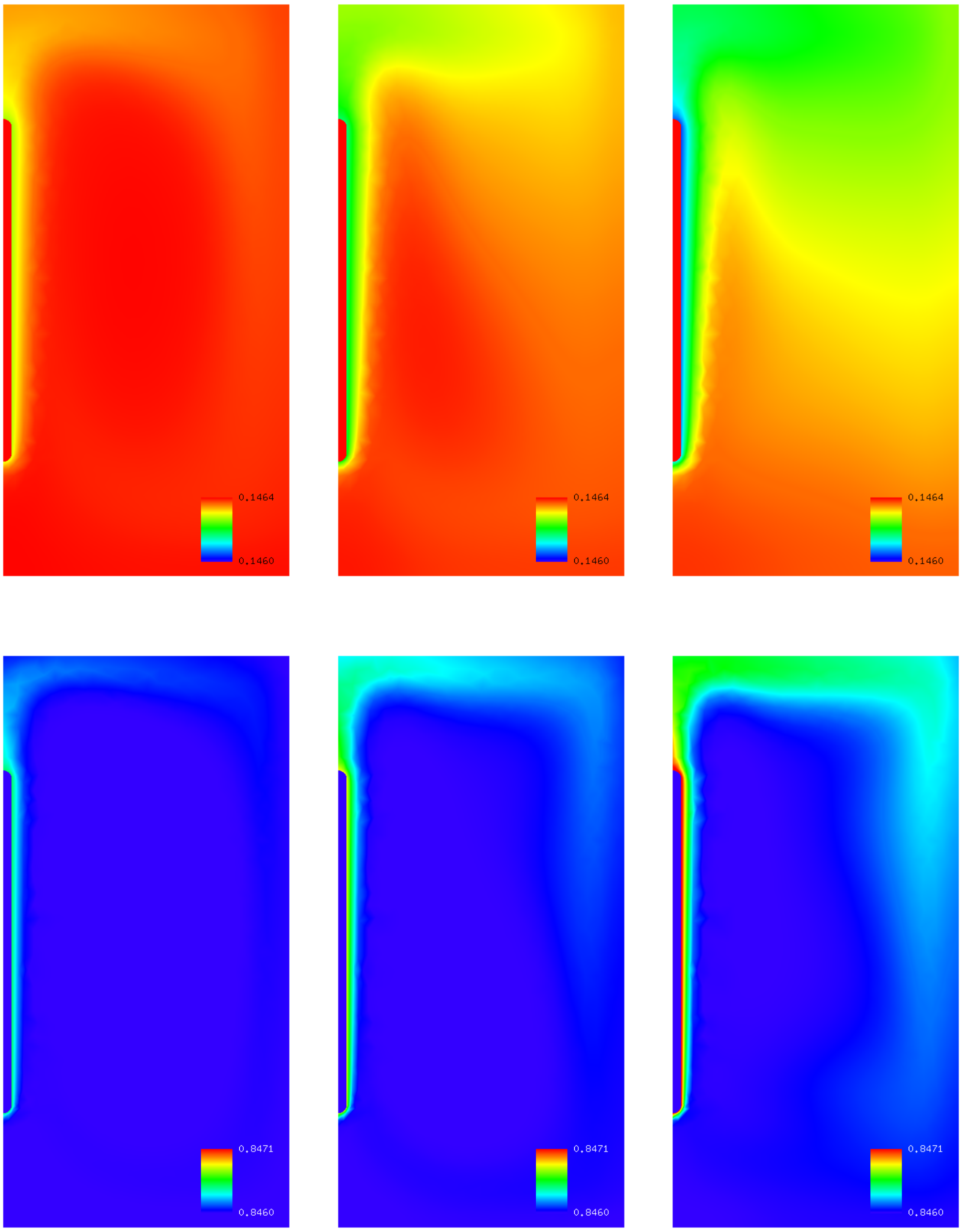


Figure 6.13: Evolution of the concentration fields of mercury (above) and tellurium (below) in the 2-dimensional simulation at time $t = 12\,min$, $t = 20\,min$, and $t = 28\,min$ in Experiment II.

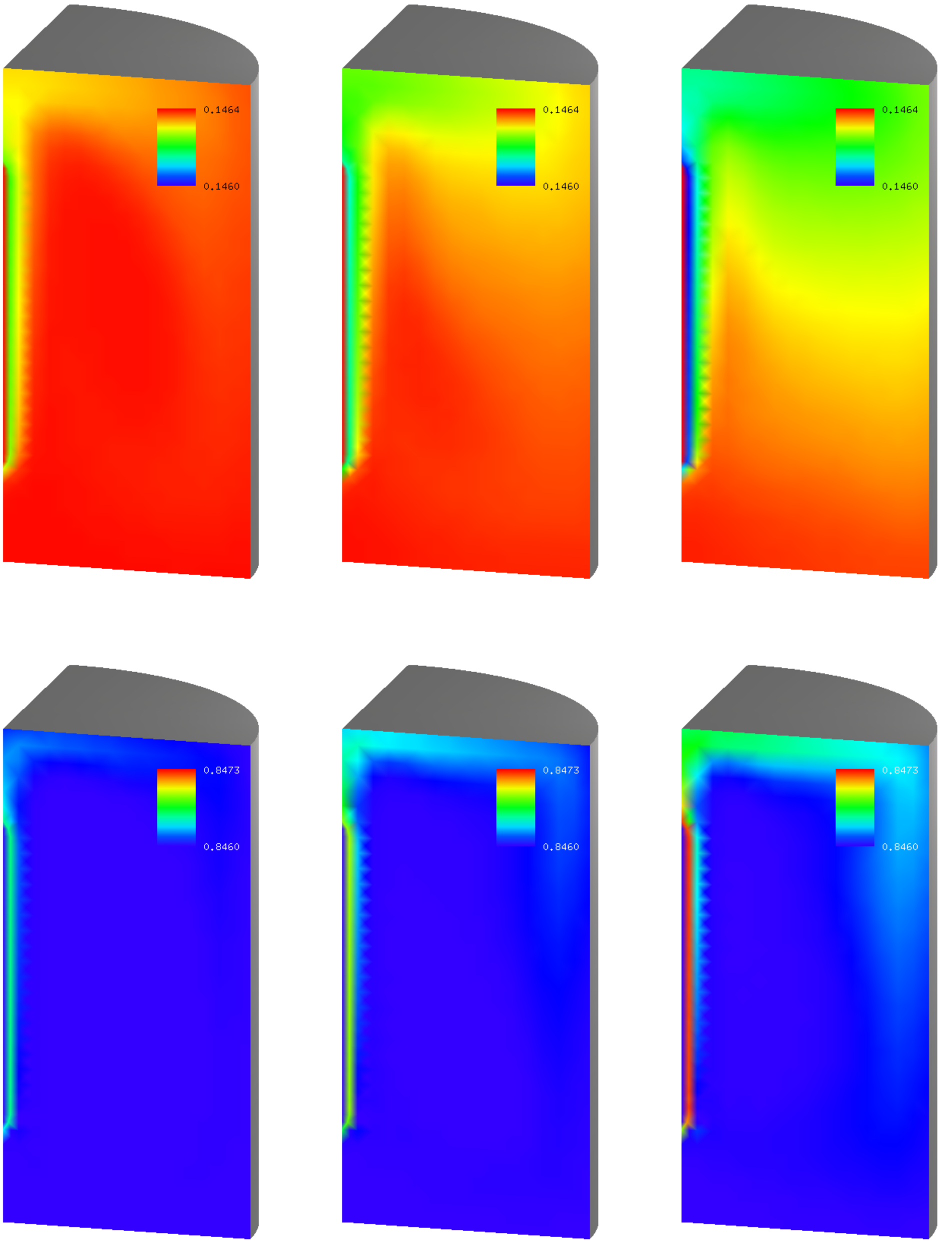


Figure 6.14: Evolution of the concentration fields of mercury (above) and tellurium (below) in the 3-dimensional simulation at time $t = 12\,min$, $t = 20\,min$, and $t = 28\,min$ in Experiment II.

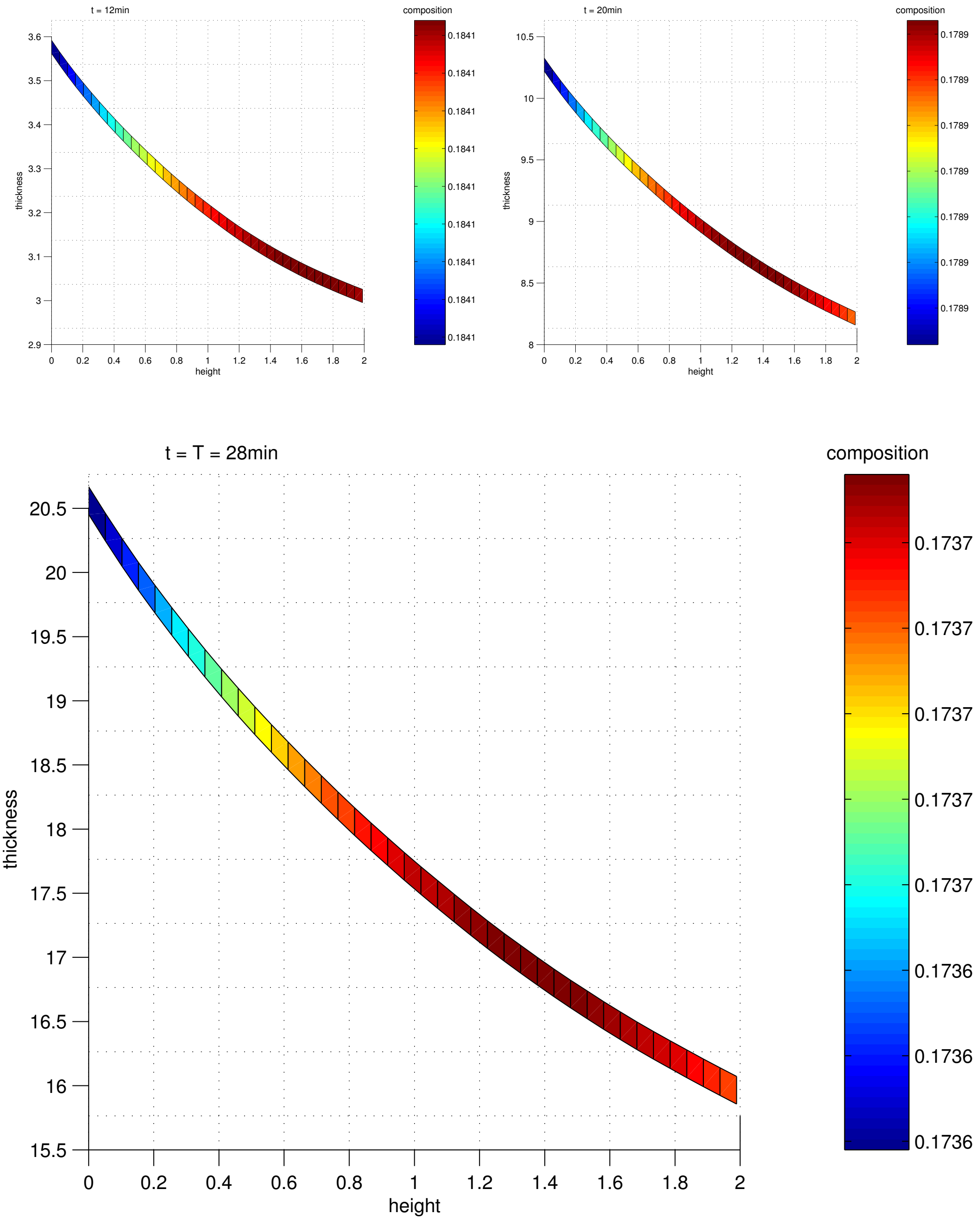


Figure 6.15: Evolution for the homogeneity of the layer at time $t = 12\,min$, $t = 20\,min$, and $t = 28\,min$ in the 2-dimensional simulation of Experiment II.

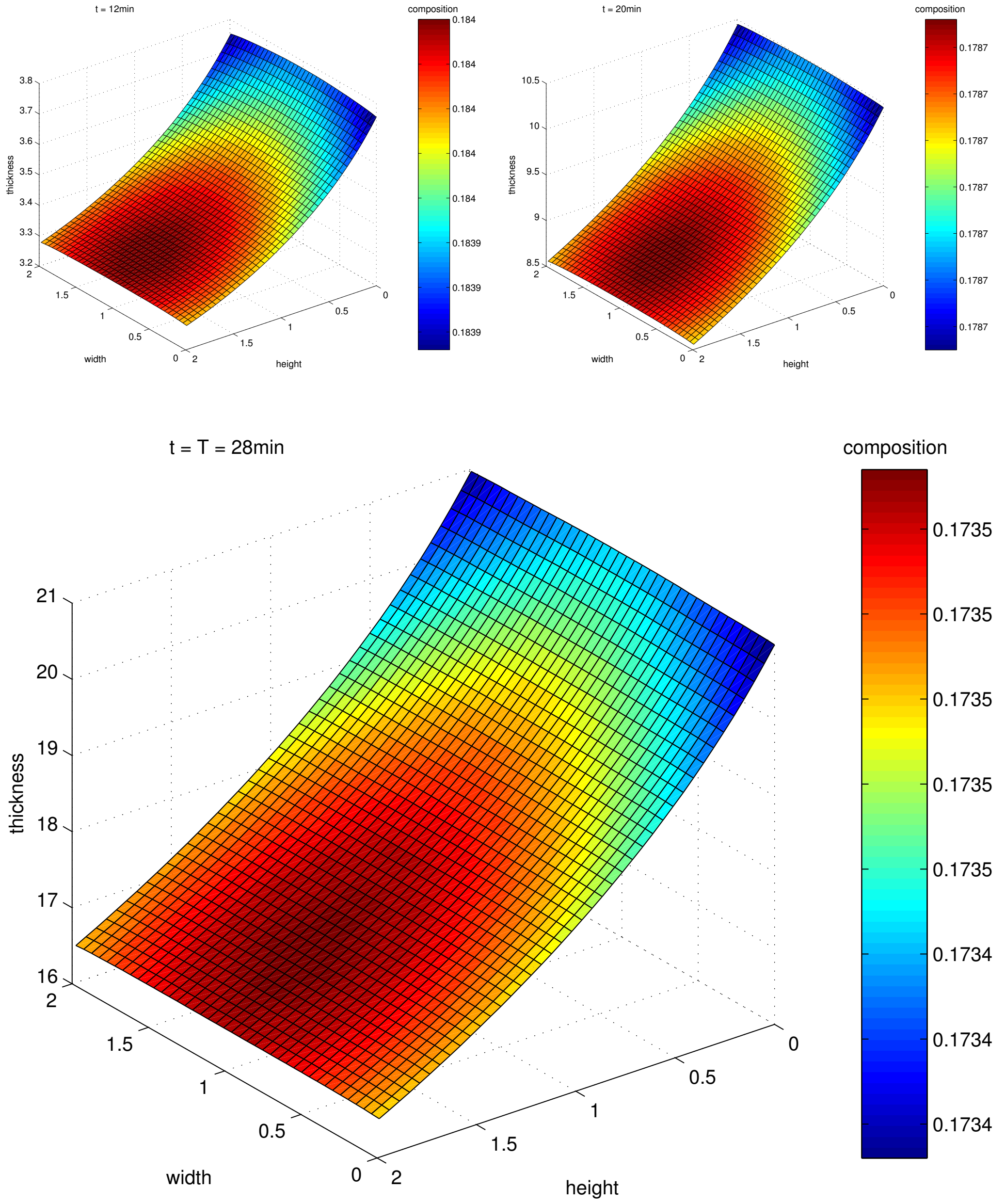


Figure 6.16: Evolution for the homogeneity of the layer at time $t = 12\,min$, $t = 20\,min$, and $t = 28\,min$ in the 3-dimensional simulation of Experiment II.

6.2.4 Experiment III

In this numerical experiment we attempt to simulate the liquid phase epitaxial process as it happens in the real experiment. To prevent the segregation of the melt due to gravity a temperature ramp along the side walls of the melting pot is generated to trigger a strong flow. In case of this strong flow numerical simulations are time-consuming. Compared to the last experiments a smaller time step size as well as a higher grid resolution is required. Since we have already seen that there is a very good accordance between 2- and 3-dimensional simulations we do without 3-dimensional simulations for this experiment. To obtain results of a 3-dimensional simulation in a reasonable time-frame parallelization of the code would be required. If it runs on a single processor one 3-dimensional simulation would need several months to finish.

Initial conditions and external temperature

The external temperature field on the boundary of the pot for a fixed time is shown in Figure 1.6 on page 7. To specify it for a 2-dimensional simulation we define the radius r and the height h in $[0,1]$ as

$$r = \tfrac{x_1+0.1}{3.5} \quad \text{and} \quad h = \tfrac{x_2}{7} \qquad \text{for } x \in \overline{\Omega} = [-0.1, 3.4] \times [0,7].$$

Then, the external temperature on the top, along the side walls, and on the bottom of the pot is given as

$$\begin{aligned} \theta_{top}(r) &= \theta_0 - t\Delta\theta - (2.4 + 0.625r^2 - 0.85r^3 + 1.825r^4)\,℃, \\ \theta_{sw}(r) &= \theta_0 - t\Delta\theta + (4.0 - 12.0h + 3.0h^2 + h^3)\,℃, \\ \theta_{bot}(r) &= \theta_0 - t\Delta\theta + (2.9 + 0.0625r^2 + 0.275r^3 + 0.7625r^4)\,℃, \end{aligned} \tag{6.8}$$

with $\Delta\theta$ as given by equation (6.6). The initial concentrations are set as in equation (6.2) and (6.3). The initial temperature is set to

$$\theta_0 = 470.54\,℃ \tag{6.9}$$

which is above the melting temperature of the initial concentrations (cf. equation (6.5)). The reason for this is as follows: In the epitaxial process the substrate is dipped into the melt. After this step the temperature is first raised to clean the surface of the substrate from impurities. In this process the mean value of the temperature in the melt reaches approximately $470.54\,℃$. Then, the furnace is cooled down with the temperature ramp as given in equation (6.6). This is done for around 34 minutes. After that, the substrate is pulled out of the melt and the process is completed.

Modification of the boundary conditions on the interface

In our model the interface Γ exists from the beginning of the simulation between the solid and the liquid part. With the choice of the external and initial temperature as in equation (6.8) and (6.9) the layer would start to melt at the beginning of the process although its thickness is zero. To prevent this non physical effect the conditions on the interface Γ are modified:

1. If the temperature is above the melting temperature, movement of the interface is set to zero, i.e. $v_\Gamma = 0$. Therefore the equations of the Stefan conditions (cf. (2.31)) are modified:

   $$\left[\hat{a}\frac{\partial \theta}{\partial \boldsymbol{\nu}}\right]_s^\ell = 0, \qquad \frac{\partial c_i^s}{\partial \boldsymbol{\nu}} = \frac{\partial c_i^\ell}{\partial \boldsymbol{\nu}} = 0 \quad (i = 1, 2). \tag{2.31'}$$

   The equations of the phase diagram (2.32) are dropped since they are not needed in this case.

2. If the temperature is below or equal the melting temperature the Stefan conditions (2.31) and the phase diagram equations (2.32) are required to hold as usual.

Let us briefly explain why the first item is meaningful. In the case where the temperature on a part of Γ is above the melting temperature the phase diagram equation for the melting temperature (cf. equation (2.32)) is not needed since we require that the interface has no motion and thus is known. The phase diagram equations of the coupling between liquid and solid concentrations (cf. again equation (2.32)) are also not required since (2.31') prescribes homogeneous Neumann boundary conditions for the concentrations on both sides of the interface Γ. With the modifications above melting of the substrate is prevented.

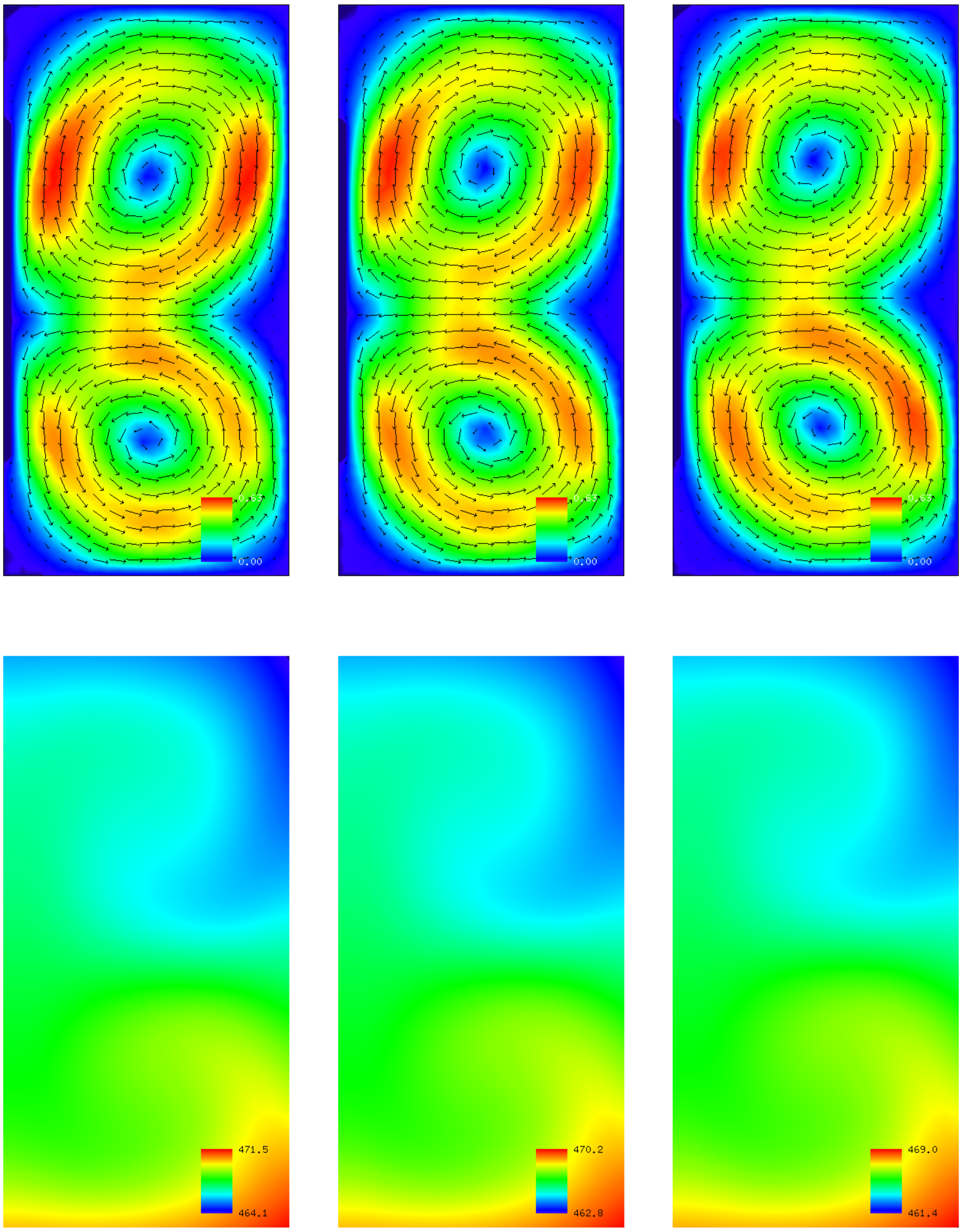


Figure 6.17: Evolution of the flow velocity (above) and the temperature (below) in the simulation at time $t = 18\,min$, $t = 26\,min$, and $t = 34\,min$ in Experiment III.

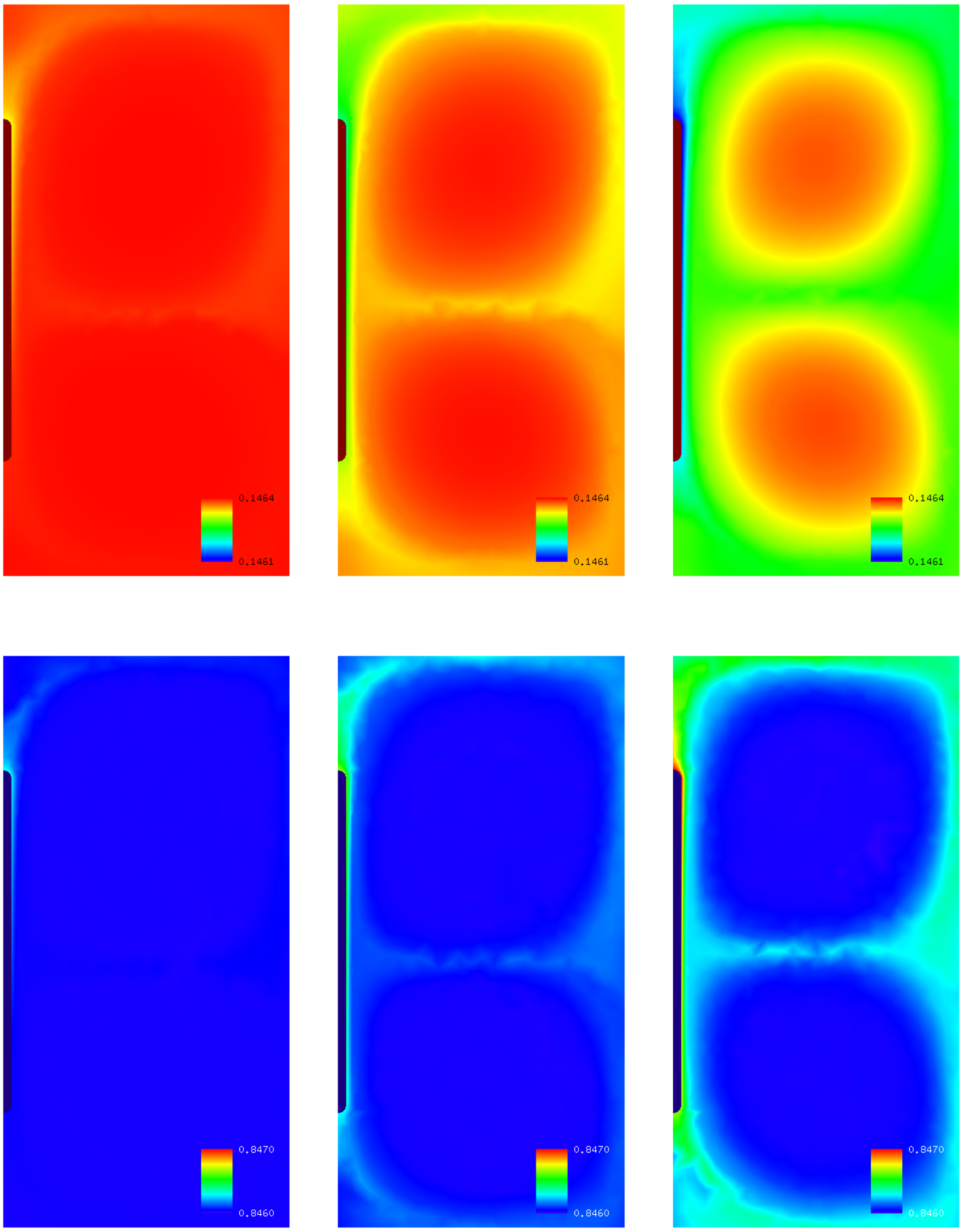


Figure 6.18: Evolution of the concentration fields of mercury (above) and tellurium (below) in the simulation at time $t = 18\,min$, $t = 26\,min$, and $t = 34\,min$ in Experiment III.

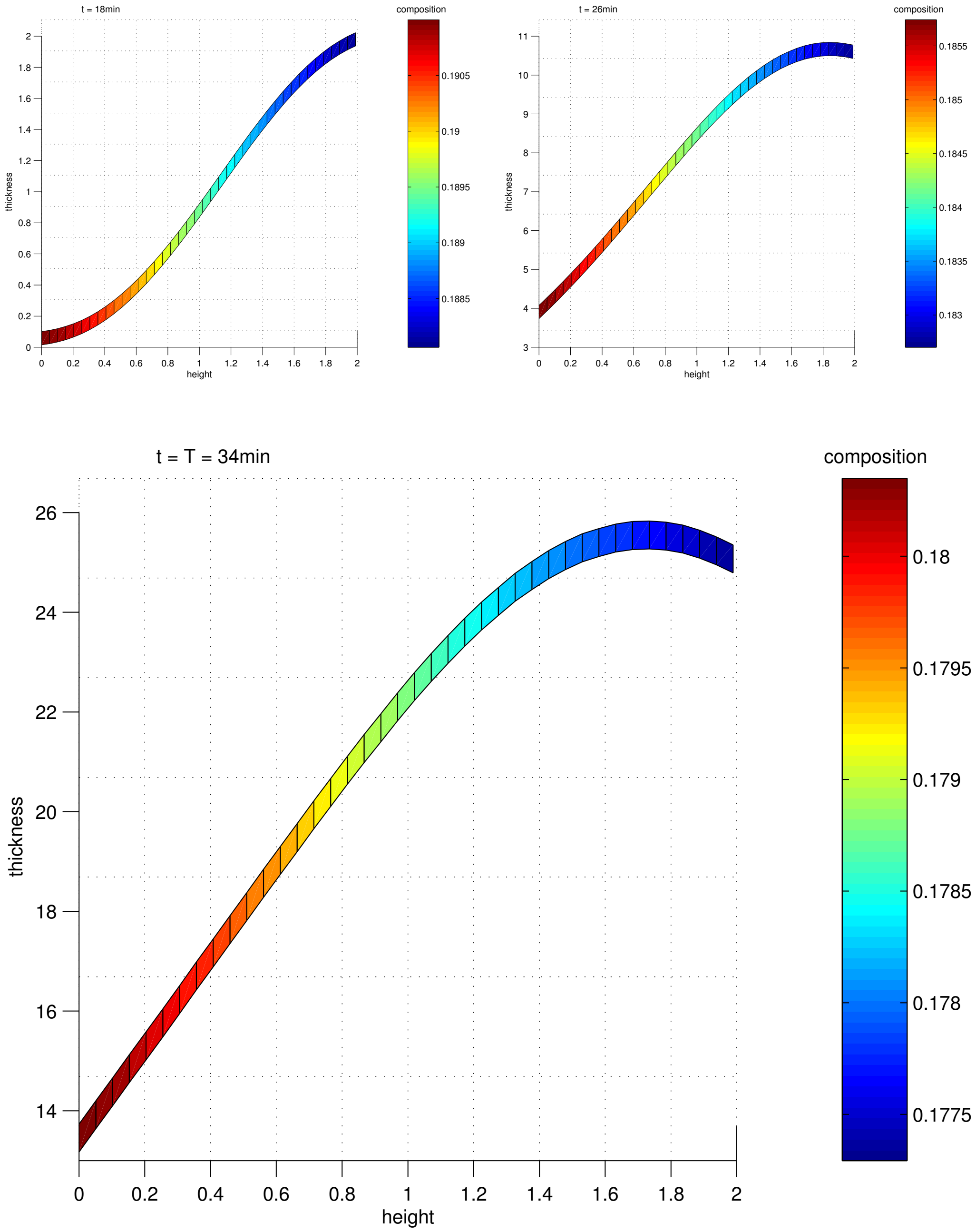


Figure 6.19: Evolution for the homogeneity of the layer at time $t = 18\,min$, $t = 26\,min$, and $t = 34\,min$ in the simulation of Experiment III.

Results of the simulation

The results of the simulation are shown in Figure 6.17, 6.18, and 6.19. As already mentioned the flow is very strong in this experiment: $\boldsymbol{u}_{max} \approx 0.63\frac{cm}{s}$ (cf. Figure 6.17). Thus, we get a good mixing effect. This can also be seen for the concentrations in Figure 6.18. Looking at Figure 6.19 we recognize that the homogeneity of the layer is very bad. The tolerance for the thickness of $\pm 2\mu m$ per square centimeter is not reached at all. Similarly, the tolerance for the homogeneity of the composition of ± 0.001 is not reached. Thus, simulation results do not agree at all with results of real experiments as exemplarily shown in Figure 6.7.

To find a reason for these curious results we perform two more simulations under modified conditions: Looking at the external temperature we recognize that at every time $t \in [0, T]$ there is a maximal difference of $8\,℃$ between bottom and top of the melting pot. Now we reduce this difference to $4\,℃$ yielding

$$\begin{aligned} \theta_{top}(r) &= \theta_0 - t\Delta\theta - \tfrac{1}{2}(2.4 + 0.625r^2 - 0.85r^3 + 1.825r^4)\,℃, \\ \theta_{sw}(r) &= \theta_0 - t\Delta\theta + \tfrac{1}{2}(4.0 - 12.0h + 3.0h^2 + h^3)\,℃, \\ \theta_{bot}(r) &= \theta_0 - t\Delta\theta + \tfrac{1}{2}(2.9 + 0.0625r^2 + 0.275r^3 + 0.7625r^4)\,℃ \end{aligned} \tag{6.8'}$$

as external temperature. Note, that this modification has only small influence on the mean value of the temperature in the melting pot since the mean values of former and modified external temperature are nearly the same. Performing a new simulation with this manipulated external temperature we observe (cf. Figure 6.20) that the homogeneity of the layer is already improved. The tolerance for the thickness and the composition are nearly reached.
Reducing the difference once more to $2\,℃$ yielding

$$\begin{aligned} \theta_{top}(r) &= \theta_0 - t\Delta\theta - \tfrac{1}{4}(2.4 + 0.625r^2 - 0.85r^3 + 1.825r^4)\,℃, \\ \theta_{sw}(r) &= \theta_0 - t\Delta\theta + \tfrac{1}{4}(4.0 - 12.0h + 3.0h^2 + h^3)\,℃, \\ \theta_{bot}(r) &= \theta_0 - t\Delta\theta + \tfrac{1}{4}(2.9 + 0.0625r^2 + 0.275r^3 + 0.7625r^4)\,℃, \end{aligned} \tag{6.8''}$$

as external temperature the variations of thickness and composition of the layer are absolutely acceptable (cf. Figure 6.21) and comparable to the experimental results. Note, that the growth process with reduced temperature difference takes longer since the arising flow is not so strong and the equalization process of the concentrations is slower.

In summary, we observe that by reducing temperature differences in the melting pot (achieved by reducing temperature differences of the external temperature) we obtain decisively better results for the homogeneity of the layer. Together with the results of the physical experiments this gives us the hint that the modeling of the temperature dependency in the epitaxial growth process is not correct. Recalling the argumentation for the derivation of the melting temperature in Section 2.5.3 we recognize that local thermodynamical equilibrium was assumed to hold on the interface Γ. The conclusion of this last experiment is that with large temperature differences this assumption may not hold. Thus, in equation (2.24) the term $\theta_{n.e.}$ concerning non equilibrium influences on the melting temperature must also be taken into account.

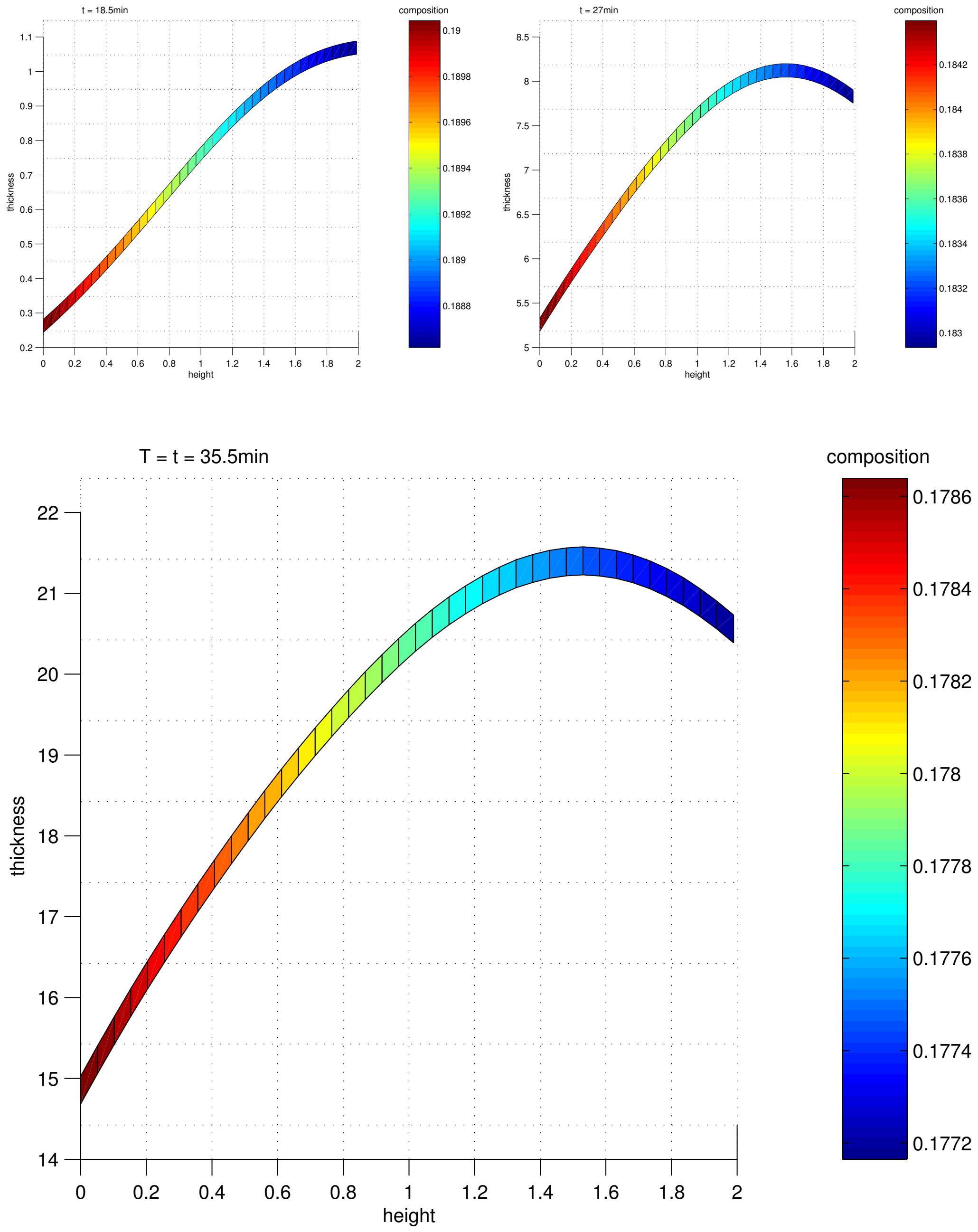


Figure 6.20: Evolution for the homogeneity of the layer at time $t = 18.5\,min$, $t = 27\,min$, and $t = 35.5\,min$ in the simulation with halved temperature ramp in Experiment III.

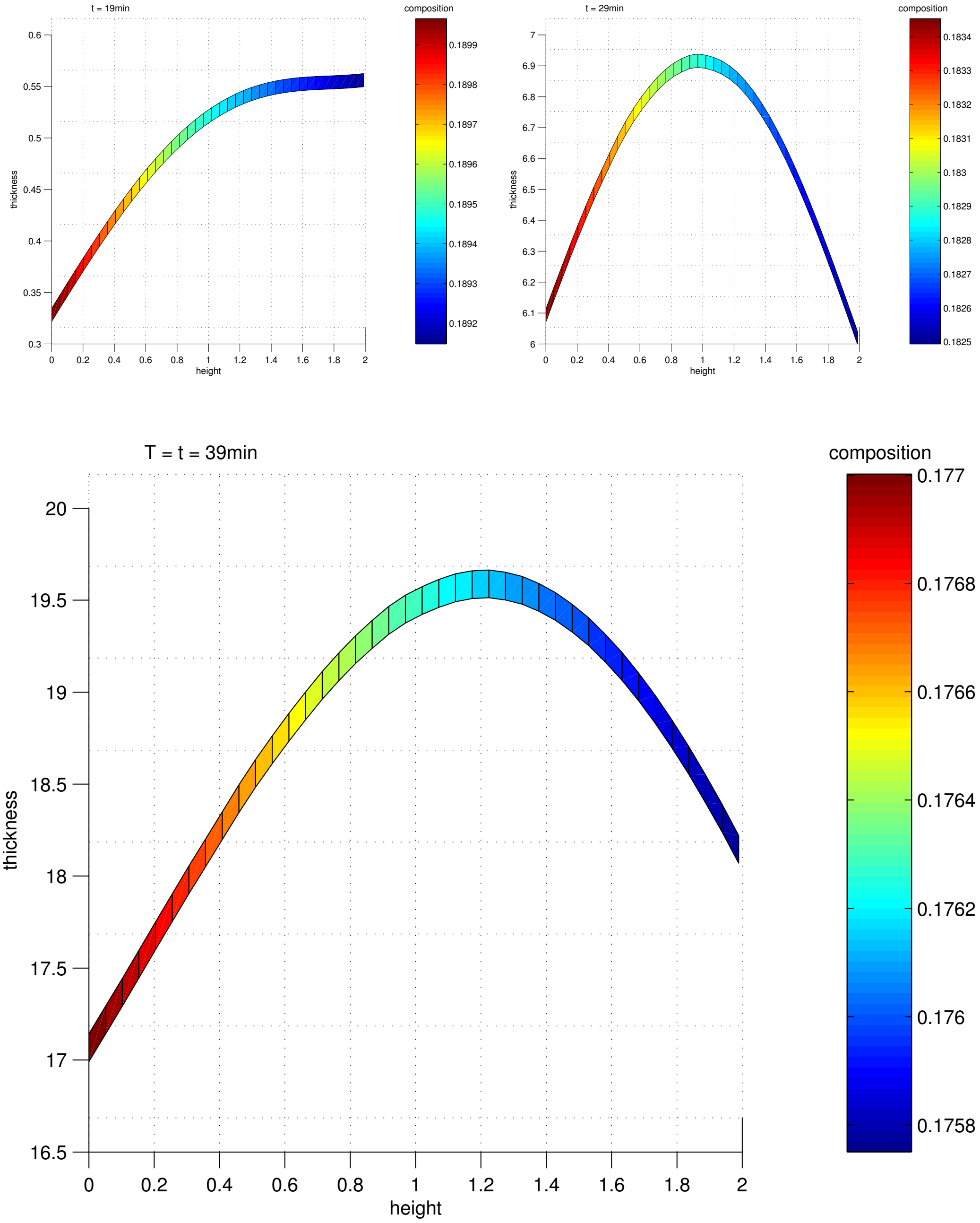


Figure 6.21: Evolution for the homogeneity of the layer at time $t = 19\,min$, $t = 29\,min$, and $t = 39\,min$ in the simulation with quartered temperature ramp in Experiment III.

6.3 Conclusions

This work presents an efficient method for the simulation of the liquid phase epitaxy as described by the physical model presented in Chapter 2 on page 27. Real-time simulations in two space dimensions of this complex problem with nonlinear coupling on the interface are almost possible. For faster 3-dimensional simulations parallelization of the code is necessary.

The introduction of the damping term (cf. Section 4.5.2) ensures the convergence of the algorithms (cf. Section 6.1) and is a powerful tool to reduce oscillations on coarse grids. In Section 6.1.3 we show that the numerical results produced by the moving and the fixed interface approach (cf. Algorithm 4.4 and 4.6) are comparable. Thus, the simplifications presented in the fixed interface approach (cf. Algorithm 4.6) allow an easier and more efficient simulation of the problem.

A good accordance for maximal flow velocity, temperature, concentrations, as well as interface velocity between 2- and 3-dimensional simulations is shown (cf. Section 6.2.2 and 6.2.3). Thus, 2-dimensional simulations for the geometry we use deliver good results with respect to quality and quantity. The effort incurred in running 3-dimensional simulations is not justified.

The model provides reasonable results for experiments with constant temperature outside the melting pot (cf. Section 6.2.2 and 6.2.3). If a temperature ramp along the side walls of the melting pot is applied, the numerical results differ drastically from the results achieved by physical experiments (cf. Section 6.2.4). As is shown by varying the external temperature in Experiment III (cf. Section 6.2.4) the quality of numerical results is strongly related to arising temperature differences. This leads to the conclusion that the modeling of temperature in the melting pot is not appropriate. Applying the phase diagram equations presumes local thermodynamical equilibrium. In our opinion this is not fulfilled in the liquid phase epitaxial process as addressed in this work since large temperature differences on the interface occur. Thus we claim that the influence of non equilibrium states for the melting temperature must be taken into account to obtain an applicable physical model and reasonable numerical results.

Bibliography

[1] R.A. Adams. *Sobolev spaces*, volume 65 of *Pure and Applied Mathematics*. Academic Press, New York - San Francisco - London, 1975.

[2] H.W. Alt. *Lineare Funktionalanalysis. Eine anwendungsorientierte Einführung*. Springer, Berlin, 1992.

[3] I. Babuška. The finite element method with Lagrangian multipliers. *Numer. Math.*, 20:179–192, 1973.

[4] M.J. Baines. *Moving Finite Elements*. Oxford University Press, New York, 1994.

[5] R. Bonnerot and P. Jamet. Numerical computation of the free boundary for the two-dimensional Stefan problem by space-time finite elements. *J. Comput. Phys.*, 25:163–181, 1977.

[6] St. Boschert, Th. Kaiser, A. Schmidt, K.G. Siebert, K.W. Benz, and G. Dziuk. Global simulation of (Cd,Zn)Te single crystal growth by the vertical Bridgman technique. In S.N. Atluri and P.E. O'Donoghue, editors, *Modelling and Simulation Based Engineering*, Palmdale, 1998. Tech Science Press.

[7] St. Boschert, A. Schmidt, and K.G. Siebert. Numerical Simulation of Crystal Growth by the Vertical Bridgman Method. In J.S. Szmyd and K. Suzuki, editors, *Modelling of Transport Phenomena in Crystal Growth*, volume 6 of *Developments in Heat Transfer*. WIT Press, 2000.

[8] St. Boschert, A. Schmidt, K.G. Siebert, E. Bänsch, K.W. Benz, G. Dziuk, and Th. Kaiser. Simulation of industrial crystal growth by the vertical

Bridgman method. In W. Jäger and H.-J. Krebs, editors, *Mathematics - Key Technology for the Future Joint Projects between Universities and Industry*, pages 315–330. Springer, 2003.

[9] D. Braess. *Finite Elemente. Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer, Berlin, 1997.

[10] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *RAIRO Anal. Numér.*, 8(R-2):129–151, 1974.

[11] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*, volume 15 of *Springer Series in Computational Mathematics*. Springer, New York, 1991.

[12] M.O. Bristeau, R: Glowinski, and J. Periaux. Numerical methods for the navier-stokes equations. application to the simulation of compressible and incompressible flows. *Computer Physics Report*, 6:73–188, 1987.

[13] E. Burman and P. Hansbo. Edge stabilization for Galerkin approximations of convection-diffusion-reaction problems. *Comp. Methods Appl. Mech. Engrg.*, 193:1437–1453, 2004.

[14] P. Cappner, editor. *Properties of Narrow Gap Cadmium-based Compounds*. INSPEC, London, 1994.

[15] P. Deuflhard. *Newton methods for nonlinear problems. Affine invariance and adaptive algorithms*, volume 35 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2004.

[16] P. Deuflhard and F. Bornemann. *Numerische Mathematik 2: Gewöhnliche Differentialgleichungen*. de Gruyter, Berlin, 2002.

[17] J. Dolbow and R. Merle. Solving thermal and phase change problems with the eXtended finite element method. *Comput. Mech.*, 28(5):339–350, 2002.

[18] J. Douglas and T. Dupont. Interior penalty procedures for elliptic and parabolic Galerkin methods. In R. Glowinski and J.L. Lions, editors, *Computing Methods in Applied Sciences*, Berlin, 1976. Springer.

[19] G. Dziuk, K. W. Benz, K.-M. Lin, and O. Kriessl. Simulation und Optimierung der Flüssigphasenepitaxie bei der Herstellung von Infrarotdetektoren. BMBF-Abschlussbericht.

[20] R. W. Freund. A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems. *SIAM Journal on Scientific Computing*, 14(2):470–482, 1993.

[21] D. Gilbarg and N.S. Trudinger. *Elliptic partial differential equations of second order*. Springer, Berlin, 2001.

[22] V. Girault and P.-V. Raviart. *Finite element methods for Navier-Stokes equations. Theory and algorithms*, volume 5 of *Series in Computational Mathematics*. Springer, Berlin, 1986.

[23] V.M. Glazov, S.N. Chizhevskaya, and N.N. Glagoleva. *Liquid Semiconductors*. Plenum Press, New York, 1969.

[24] P. Grisvard. *Elliptic problems in nonsmooth domains*, volume 24 of *Monographs and Studies in Mathematics*. Pitman Advanced Publishing Program, Boston-London-Melbourne, 1985.

[25] Morton E. Gurtin. *An introduction to continuum mechanics*, volume 158 of *Mathematics in science and engineering*. Academic Press, New York, 1981.

[26] W. Hackbusch. *Theorie und Numerik elliptischer Differentialgleichungen*. B.G. Teubner, Stuttgart, 1986.

[27] T.C. Harman. Liquidus isotherms, solidus lines and LPE growth in the Te-rich corner of Hg-Cd-Te systems. *Journal of Electronic Materials*, 9(6):945–961, 1980.

[28] P. Hood and C. Taylor. A numerical solution of the navier-stokes equations using the finite element technique. *Comput. Fluids*, 1:73–100, 1973.

[29] Takamichi Iida and Roderick I.L. Guthrie. *The Physical Properties of Liquid Metals*. University Press, Oxford, 1993.

[30] H. Ji, D. Chopp, and J. Dolbow. A hybrid extended finite element/level set method for modeling phase transformations. *Int. J. Numer. Methods Eng.*, 54(8):1209–1233, 2002.

[31] D.H. Kim and R.A. Brown. Modelling of the dynamics of HgCdTe growth by the vertical Bridgman method. *Journal of Crystal Growth*, 114:411–434, 1993.

[32] M. Kimura, , N. Djilali, S. Dost, H. Kanai, A. Tanaka, and T. Sukegawa. Liquid phase epitaxy of silicon: an experimental and numerical parametric study. *J. Crystal Growth*, 167:516–524, 1996.

[33] M. Kimura, N. Djilali, and S. Dost. Convective transport and interface kinectics in liquid phase epitaxy. *J. Crystal Growth*, 143:334–348, 1994.

[34] M. Kimura, Z. Qin, and S. Dost. A solid-liquid diffusion model for growth and dissolution of ternary alloys by liquid phase epitaxy. *J. Crystal Growth*, 158:231–240, 1996.

[35] R. Kößler. *Messung von Diffusionskoeffizienten in Kristallzüchtungsschmelzen von Halbleiterverbindungen nach dem Scherzellenverfahren*. PhD thesis, Universität Karlsruhe, 1991.

[36] L. D. Landau and E. M. Lifschitz. *Hydrodynamik*, volume 6 of *Lehrbuch der theoretischen Physik*. Akademie-Verlag, Berlin, 1991.

[37] Landolt-Börnstein. *Numerical Data and Functional Relationships in Sciene and Technology*. Springer new series, Berlin.

[38] David R. Lide, editor. *Handbook of Chemistry and Physics*. CRC Press, London, 1999.

[39] K. Lin, S. Boschert, P. Dold, K.-W. Benz, O. Kriessl, A. Schmidt, K.G. Siebert, and G. Dziuk. Numerical methods for industrial vertical Bridgman growth of (Cd,Zn)Te. *J. Crystal Growth*, 237–239:1736–1740, 2002.

[40] J.L. Lions and E. Magenes. *Non-homogeneous boundary value problems and applications*, volume 1. Springer, Berlin, 1972.

[41] J.L. Lions and E. Magenes. *Non-homogeneous boundary value problems and applications*, volume 2. Springer, Berlin, 1972.

[42] Andreas Meister. *Numerical methods for linear systems of equations. An introduction to modern methods*. Vieweg, Wiesbaden, 2005.

[43] S. Müller-Urbaniak. Eine Analyse des Zwischenschritt-θ-Verfahrens zur Lösung der instationären Navier-Stokes-Gleichungen. Preprint 94-01, University of Heidelberg, 1994.

[44] F. Otto. Die Babuška-Brezzi Bedingung für das Taylor-Hood-Element. Master's thesis, University of Bonn, 1990.

[45] J. Pitkäranta. Boundary subspaces for the finite element method with Lagrange multipliers. *Numer. Math.*, 33:273–289, 1979.

[46] Z. Qin, M. Kimura, and S. Dost. Convection model for growth and dissolution of ternary alloys by liquid phase epitaxy. *J. Crystal Growth*, 167:74–86, 1996.

[47] P.A. Raviart and J.M. Thomas. Primal hybrid finite element methods for 2nd order elliptic equations. *Math. Comput.*, 31(138):391–413, 1977.

[48] Yousef Saad. *Iterative methods for sparse linear systems. 2nd ed.* SIAM Society for Industrial and Applied Mathematics, Philadelphia, 2003.

[49] A. Schmidt and K.G. Siebert. *Design of Adaptive Finite Element Software*. Springer, Berlin, 2005.

[50] P. Sonneveld. CGS, a fast Lanczos-type solver for nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 10(1):36–52, 1989.

[51] R. Temam. *Navier-Stokes equations. Theory and numerical analysis*. AMS, Providence, RI, 2001.

[52] Vidar Thomée. *Galerkin Finite Element Methods for Parabolic Problems*, volume 25 of *Springer Series in Computational Mathematics*. Springer, Berlin, 1997.

[53] Augusto Visintin. *Models of phase transitions*, volume 28 of *Progress in nonlinear differential equations and their applications*. Birkhäuser, Boston, 1996.

[54] C. Weichmann, R. Backofen, and A. Voigt. Time dependent 3D heat radiation calculation in high temperature furnaces. In B. Sunden, C.A. Brebbia, and A. Mendes, editors, *Advanced Comp. Methods in Heat Transfer*, pages 313–323. WIT Press, 2004.

[55] J. Wloka. *Partielle Differentialgleichungen*. B.G. Teubner, Stuttgart, 1982.

[56] P. Wriggers. *Nichtlineare Finite-Element-Methoden*. Springer, Berlin, 2001.