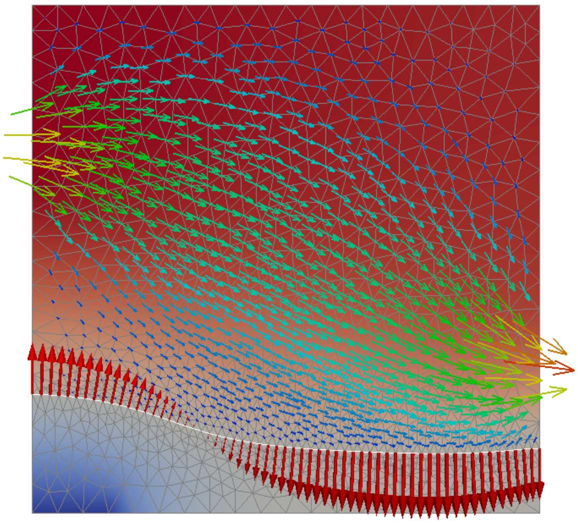
OPTIMAL CONTROL OF A STEFAN PROBLEM WITH GRADIENT-BASED METHODS IN FENICS



Master Thesis

submitted to the
Faculty of Mathematics
at
Otto-von-Guericke-Universität Magdeburg
in accordance with the requirements for the degree
Master of Science

authored by BJÖRN BARAN
born at December 08, 1989 in Flensburg,
university course Mathematics.

February 4, 2016

Supervised at Max Planck Institute
for Dynamics of Complex Technical Systems by
DR. JAN HEILAND

Contents

Notation V

Symbols VI

Abbreviations X

1. Introduction 2
- 1.1. Motivation 2
- 1.2. Prior Work 2
- 1.3. Research Goals 4
- 1.4. Outline 4

2. Two-Phase Stefan Problem 6
- 2.1. Domain 6
- 2.2. Heat Equation 7
- 2.3. Mesh Movement 9
- 2.4. Navier–Stokes Equations 9

3. Optimization 11
- 3.1. Optimal Control Problem 11
  - 3.1.1. Forward System 13
  - 3.1.2. Cost Functional 14
- 3.2. Derivation of the Adjoint System 14
  - 3.2.1. Lagrange Functional 15
  - 3.2.2. Adjoint Equations 16
  - 3.2.3. Backward System 33
- 3.3. Gradient Method 34
  - 3.3.1. Projected Gradient Method 35
  - 3.3.2. Line Minimization Algorithm 37
  - 3.3.3. Treatment of Control Constraints 38

4. Implementation and Discretization 39
- 4.1. Spatial Discretization 40
- 4.2. Weak Formulations 40

5. Numerical Examples 49
- 5.1. General Setting 49
- 5.2. Stabilizing to a Flat Position 50
- 5.3. Stabilizing to a Reachable Flat Position 59
- 5.4. Moving to a Reachable Position 64

6. Conclusions 69

7. Perspectives 70

A. Appendix 71
  A.1. Interface Normal Representation 71
  A.2. Stefan Condition Reformulation 72
  A.3. Results for the Example from Section 5.2 73
  A.4. Results for the Example from Section 5.3 74
  A.5. Results for the Example from Section 5.4 75

Bibliography 79

List of Figures

2.1. The domain $\Omega \in \mathbb{R}^2$ for the Stefan problem. 7

4.1. Triangulation of the domain $\Omega(t_0)$ respecting the interface position (−). 39
4.2. Triangulation of the domain $\Omega(t_N)$ respecting the moved interface position. 40

5.1. Numerical solution of the forward problem at $t = 1$. 50
5.2. Interface graphs with $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$. 51
5.3. Computed control for $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$. 52
5.4. Interface graphs with $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$ at various t. 53
5.5. Cost functional for $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$. 54
5.6. Interface distance $\mathfrak{d}$ at $t = E$ for different parameter sets. 55
5.7. Interface distance $\mathfrak{d}_{all}$ for the whole time interval for different parameter sets. 55
5.8. Cost functional for different initial guesses. 56
5.9. Computed controls c_1, c_5, c_6 with different initial guesses. 57
5.10. Interface graphs for the controls c_1, c_5, c_6. 58
5.11. Interface graphs with a reachable interface position $\tilde{c}_d$. 59
5.12. Interface graphs with a reachable interface position $\tilde{c}_d$ and control $\tilde{c}_1$ for various t. 60
5.13. Interface distance $\mathfrak{d}$ at $t = E$ for different parameter sets. 61
5.14. Interface distance $\mathfrak{d}_{all}$ the whole time interval for different parameter sets. 62
5.15. Interface distance $\mathfrak{d}$ at $t = E$ for different initial guesses and parameter sets. 63
5.16. Interface distance $\mathfrak{d}$ at $t = E$ for different values of λ. 63
5.17. Interface graphs with a reachable interface position $\hat{c}_d$. 64
5.18. Interface graphs with a reachable interface position $\hat{c}_d$ for various t. . . 65
5.19. Interface distance $\mathfrak{d}$ at $t = E$ for different parameter sets. 67
5.20. Interface distance $\mathfrak{d}_{all}$ the whole time interval for different parameter sets. 67
5.21. Computed controls $\hat{c}_1$ to $\hat{c}_5$ with different parameter sets. 68
5.22. Interface distance $\mathfrak{d}$ at $t = E$ for different controls $\hat{c}_1$, $\hat{c}_6$, $\hat{c}_7$, $\hat{c}_8$ with different initial guesses. 68

Notation

∂S	for $S \subset \mathbb{R}^n, n \in \{1,2\}$: $(n-1)$-dimensional boundary of S
$\circ$	concatenation of two functions: $(f \circ z)(x) = f(z(x))$
$\partial_{\boldsymbol{\mu}} f$	gradient of f in outer normal direction $\partial_{\boldsymbol{\mu}} f = \nabla f \cdot \boldsymbol{\mu}$
∇	gradient operator $\nabla \cdot = \left[\frac{\partial \cdot}{\partial x}, \frac{\partial \cdot}{\partial y}\right]^{\mathsf{T}}$
$\partial_t f$	derivative of f with respect to time
$\partial_x f$	derivative of f in horizontal direction $\partial_x f = \nabla f \cdot [1, 0]^{\mathsf{T}}$
$\partial_y f$	derivative of f in vertical direction $\partial_y f = \nabla f \cdot [0, 1]^{\mathsf{T}}$
∇f	gradient of $f \colon \mathbb{R}^2 \to \mathbb{R}$, $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]^{\mathsf{T}}$
$(\nabla f)_l$	gradient of f in normal direction pointing from the liquid to the solid phase $(\nabla f)_l = \partial_{-\boldsymbol{\nu}} f\vert_{\Omega_l}$
$[k(\nabla f)]_l^s$	jump of the gradient of f in normal direction across the interface $[k(\nabla f)]_l^s = [k_s(\nabla f)_s - k_l(\nabla f)_l]$
$(\nabla f)_s$	gradient of f in normal direction pointing from the solid to the liquid phase $(\nabla f)_s = \partial_{\boldsymbol{\nu}} f\vert_{\Omega_s}$
Δ	Laplace operator $\Delta \cdot = \frac{\partial^2 \cdot}{\partial x^2} + \frac{\partial^2 \cdot}{\partial y^2}$
$D_w f$	total derivative of f with respect to w
∇v	gradient of $v = [v_x, v_y]^{\mathsf{T}} : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$, $\nabla v = \begin{bmatrix} \frac{\partial v_x}{\partial x} & \frac{\partial v_x}{\partial y} \\ \frac{\partial v_y}{\partial x} & \frac{\partial v_y}{\partial y} \end{bmatrix}$

Symbols

C^0	space of continuous functions
$\mathcal{C}_{ad}$	set of admissible controls $\mathcal{C}_{ad} \subset \mathcal{C}$
$\mathcal{C}$	control space
$\mathcal{D}$	finite element space for the adjoint state on the interface
E	end time
Γ_C	cooling boundary for the heat equation
Γ_H	inflow boundary for the Navier–Stokes equations and the heating boundary for the heat equation
Γ_I	interface between the solid and liquid phases
$\Gamma_{\tilde{N}}$	zero boundary for the mesh movement at the top of the domain, $\Gamma_{\tilde{N}} \subset \Gamma_N$
Γ_N	no-slip boundary for the Navier–Stokes equations and do-nothing boundary for the heat equation
Γ_{out}	outflow boundary for the Navier–Stokes equations
$\mathcal{G}$	finite element space for the adjoint velocity in the liquid phase
$\hat{\mathcal{H}}$	finite element space for the derivative of the interface graph
$\mathcal{H}$	finite element space for the interface graph
$\mathcal{J}$	finite element space for the adjoint pressure in the liquid phase
J	cost functional
$\mathcal{K}$	reformulated cost functional which depends only on the control
L_T	linear form for the weak formulation of the heat equation
L_γ	bilinear form for the weak formulation of the adjoint Navier–Stokes equations
L_ω	linear form for the weak formulation of the adjoint heat equation
L_ψ	linear form for the weak formulation of the adjoint equation on the interface
L_u	bilinear form for the weak formulation of the Navier–Stokes equations
$\mathcal{L}$	Lagrange functional
$\bar{\Lambda}$	weight factor of the interface distance term over the whole time interval $\mathfrak{d}_{all}$ in the cost functional

Λ	weight factor of the interface distance term at $t = E$, $\mathfrak{d}$, in the cost functional
L	latent heat coefficient
Ω_l	liquid phase
Ω_s	solid phase
Ω	hold all domain
$\mathcal{O}$	finite element space for the adjoint temperature
$\mathbb{P}_k$	space of polynomials with degree less than or equal to k
$\mathcal{P}$	finite element space for the pressure in the liquid phase
$\mathcal{Q}$	finite element space for the temperature
$\mathbb{R}$	real numbers
Σ_C	spatial discretization of Γ_C
Σ_I	spatial discretization of Γ_I
Σ_l	triangulation of Ω_l
Σ_s	triangulation of Ω_s
Σ	triangulation of Ω
T_0	initial temperature distribution
T_C	cooling temperature at Γ_C
T_H	heating temperature at Γ_H
T_m	characteristic melting temperature
T	temperature $T\colon [0, E] \times \Omega \to \mathbb{R}$
$\mathcal{U}$	finite element space for the velocity in the liquid phase
$\mathcal{V}_{\Gamma_I}$	finite element space for the interface velocity
V_{Γ_I}	velocity of the interface Γ_I in direction $\boldsymbol{\nu}$, $V_{\Gamma_I}\colon\ [0,\ E]\ \times\ \Gamma_I\ \to\ \mathbb{R}$
V_{all}	mesh movement function which is extending V_{Γ_I} to the whole domain Ω, $V_{all}\colon\ [0,\ E]\ \times\ \Omega\ \to\ \mathbb{R}^2$
$\mathcal{V}$	finite element space for the mesh movement
$\mathcal{W}$	state space
a_T	bilinear form for the weak formulation of the heat equation
a_γ	multilinear form for the weak formulation of the adjoint Navier–Stokes equations
a_ω	bilinear form for the weak formulation of the adjoint heat equation
a_ψ	bilinear form for the weak formulation of the adjoint equation on the interface
a_u	multilinear form for the weak formulation of the Navier–Stokes equations
α	heat conductivity in the solid and liquid phases
$\underline{c}, \overline{c}$	pointwise lower and upper bound to admissible controls
c	control $c \in \mathcal{C}$

$\mathfrak{d}_{all}$	interface distance term over the whole time interval in the cost functional
δ_1, δ_2	tolerances for the projected gradient algorithm
d^k	descent direction in kth iteration step of the projected gradient algorithm
$\mathfrak{d}$	interface distance term at $t = E$ in the cost functional
$\boldsymbol{e_2}$	unit vector $\boldsymbol{e_2} = [0, 1]^\mathsf{T}$
$\epsilon_1, \epsilon_2, \epsilon_3$	tolerances for the quadratic line minimization algorithm
η	kinematic viscosity
γ^{out}	adjoint state $\gamma^{out}\colon\ [0, E] \times (\Gamma_{out} \cap \partial\Omega_l) \to \mathbb{R}^2$
γ	adjoint state $\gamma\colon\ [0, E] \times \Omega_l \to \mathbb{R}^2$ (adjoint velocity)
h_0	initial graph representing the interface position at $t = 0$, $h_0\colon\ \Gamma_C \to \mathbb{R}$
h_d	graph representing the desired interface position, $h_d\colon\ [0, E] \times \Gamma_C \to \mathbb{R}$
h_t	derivative of h with respect to time, $h_t = \partial_t h$
h_x	derivative of h with respect to space, $h_x = \frac{dh}{dx}$
h	graph representing the interface position, $h\colon\ [0, E] \times \Gamma_C \to \mathbb{R}$
i_{max}	maximum iteration number of the quadratic line minimization algorithm
$\boldsymbol{\nu}$	unit normal along Γ_I pointing from the solid to the liquid phase
i	iteration index in the quadratic line minimization algorithm
k_{max}	maximum iteration number of the projected gradient algorithm
k_j	value of the cost functional at sampling point $s_j, j = 0, 1, 2$ in the quadratic line minimization algorithm $k_j = \mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k\,-\,1} - s_j \cdot d^k))$
k_l	heat conductivity in the liquid phase
k_s	heat conductivity in the solid phase
k	iteration index in the projected gradient algorithm
λ	weight factor of the control cost term $\mathfrak{p}$ in the cost functional
$\boldsymbol{\mu}$	outer unit normal of the domain
ω^I	adjoint state $\omega^I\colon\ [0, E] \times \Gamma_I \to \mathbb{R}$
ω	adjoint state $\omega\colon\ [0, E] \times \Omega \to \mathbb{R}$ (adjoint temperature)
$\mathfrak{p}$	control cost term in the cost functional
p_{in}	pressure at inflow boundary $p_{in}\colon\ [0,\ E] \times \Gamma_H \to \mathbb{R}$

Φ	function mapping from Γ_C to Γ_I
π	adjoint state $\pi\colon\ [0,E] \times \Omega_l \to \mathbb{R}$ (adjoint pressure)
$\mathbf{P}_{[\underline{c},\ \overline{c}]}$	projection $\mathbf{P}_{[\underline{c},\ \overline{c}]}\colon \mathcal{C} \to \mathcal{C}_{ad}$
ψ^C	adjoint state $\psi^C\colon\ [0,E] \times \Gamma_C \to \mathbb{R}$
ψ^I_{all}	adjoint state $\psi^I_{all}\colon\ [0,E] \times \Gamma_I \to \mathbb{R}^2$
ψ_{all}	adjoint state $\psi_{all}\colon\ [0,E] \times \Omega \to \mathbb{R}^2$
ψ	adjoint state $\psi\colon\ [0,E] \times \Gamma_C \to \mathbb{R}$
p	pressure $p\colon\ [0,\ E] \times \Omega_l \to \mathbb{R}$
q	quadratic polynomial $q\colon [s_0, s_2] \to \mathbb{R}$ for the quadratic line minimization algorithm
s	sampling point in the quadratic line minimization algorithm
s^k	step size in kth iteration step of the projected gradient algorithm
τ	time step size
u	velocity $u\colon\ [0,\ E] \times \Omega_l \to \mathbb{R}^2$
φ	adjoint state $\varphi\colon\ [0,E] \times \Gamma_H \to \mathbb{R}^2$
v, q	test functions for weak formulations

Abbreviations

FEM	Finite Element Method
PDE	Partial Differential Equation
X-FEM	Extended Finite Element Method
ALE	Arbitrary Lagrangian-Eulerian
FEniCS	Finite Element Computational Software, “ni sits nicely in the middle”, [Anders Logg]
SQP	Sequential Quadratic Programming
s.t.	subject to

Acknowledgment

First of all, I want to thank my supervisor Jan Heiland for guiding me throughout this work. Without his hints, I would have been stuck several times and his comments helped me a lot while writing down this thesis.

I also want to thank Peter Benner, not only for the possibility to write my thesis in this interesting field of research, but also for letting me work as a student in his group at the Max Planck Institute in Magdeburg. There I found a nice and faithful working environment, which is also due to the colleagues at the MPI.

In this context, I mention Jens Saak in particular. He has been my mentor in the last few years and opened the door to all of this. In his lectures and while working with him as a student worker, I learned more than in the rest of my study. As the supervisor of my scientific project, which preceded this thesis, he laid the foundation of this work. Together with my office colleague Heiko Weichelt, he encouraged me to remain in the science and continue as a PhD student.

Meine Familie hat mich von Anfang bis Ende meines Studiums in jeder erdenklichen Weise unterstützt. Auch wenn sie mich in letzter Zeit viel zu selten zu Gesicht bekommen haben, haben sie immer ein offenes Ohr für jegliche Probleme und haben mein ganzes Studium überhaupt erst möglich gemacht.

Special heartfelt thanks go to Kirsten. She read every single chapter more often than anybody else and provided me with motivation and moral support all the time.

1. Introduction

1.1. Motivation

Free boundary and moving boundary problems that can be used to model crystal growth or the solidification and melting of pure materials receive growing attention in science and technology [8, 40]. The optimal control of these problems appears even more interesting since certain desired shapes of the boundaries improve, for example, the material quality. In the case of crystal growth, a flat or at least convex interface increases the quality of the produced crystal. Besides the quality of the resulting product, the amount of energy consumed or the length of a production cycle can depend on the interface shape [8].
Problems with free boundaries include a strong coupling between unknown geometric quantities, in this case the interface position, and unknown physical quantities like the temperature distribution and the velocity of the fluid in the liquid phase. This results in a non-linear problem, for which the numerical solution and especially the optimal control are challenging tasks.

A model from the class of moving boundary problems is the two-phase Stefan problem, where the domain consists of two areas, the solid and the liquid phase. In both phases, the heat distribution is characterized by the heat equation and they are separated by a moving inner boundary. The motion of this interface is coupled to the temperature through the Stefan condition. This condition connects the normal velocity of the interface to the jump of the temperature gradient across the interface. Further, the model can be coupled with Navier–Stokes equations to describe the fluid flow in the liquid phase. The optimal control of such a non-linear problem in two spatial dimensions still comes with open tasks and is subject of this work.

1.2. Prior Work

This thesis addresses several mathematical fields. Namely, the modeling and numerical simulation of moving boundary problems, more precisely of Stefan problems. This includes mesh movement techniques combined with finite element methods (FEMs). Additionally, the modeling and numerical methods of Navier–Stokes equations are treated here. The thesis also expands on optimal control and PDE-constrained optimization.

A theoretical approach to Stefan-type problems can be found in several textbooks, e.g., [25, 30, 35]. More recent examples are [12, 14], which are closer to the current state of research. Previous work on the forward simulation of the Stefan problem mainly focused on one-dimensional or one-phase Stefan problems. There exist also weak formulations that completely avoid the explicit representation of the interface.

Among others, White published an enthalpy formulation of the Stefan problem [37] and discussed its numerical solution [36]. The enthalpy formulation works with a mushy region of material. Neither the tracking nor the explicit representation of the interface is required for this approach, which makes the implementation relatively simple. In [23, p. 219], a quite short FEniCS [1] implementation of such a formulation can be found. However, this approach is not suitable for the optimal control with interface tracking where the sharp interface representation is crucial. A historical overview of free boundary and solidification models is given in [39]. One possibility to treat the moving boundary explicitly is using extended finite elements (X-FEM) and a level set function as it is done by Zabaras et al. [38]. Here the interface is represented as the zero level set of a time dependent implicit function. This function is approximated together with the state equations on a fixed grid with finite elements. The weak discontinuity of the temperature at the interface is treated by X-FEM, which modify the FEM functions spaces in a narrow band around the interface. A similar idea was already used earlier by Nochetto et al. [26, 27]. Bernauer adapts the same approach in his PhD thesis [8] and the corresponding papers like [9]. Bänsch et al. are representing the interface in a different, more explicit way. They handle the interface with finite elements combined with a moving mesh, which respects the interface in every time step [5, 6].

Mesh movement methods were originally developed for finite differences. They have been used for one of the first times together with finite elements in [18]. This technique is also often found under the name of arbitrary Lagrangian-Eulerian (ALE) methods. For a historical overview look at [11]. Besides [6], also Beckett and his collaborators apply a moving mesh to solve the two-dimensional Stefan problem [7].

The system in this thesis is coupled with the Navier–Stokes equations, as is already mentioned in [6] and [38]. For the derivation of these equations look at the textbooks [10, 13], which further contain numerical methods and finite element techniques. In [6], $\mathbb{P}_2/\mathbb{P}_1$ Taylor–Hood finite elements [32] are used to approximate the velocity and pressure. Ziegenbalg solves a Stefan problem coupled with Navier–Stokes equations in his PhD thesis [40], too. In contrast to what we consider here, he uses finite differences combined with a moving mesh.

Well-known textbooks about optimal control and PDE-constrained optimization are [20, 34]. They suggest possible choices for a cost functional which has the purpose to steer the system to a desired state. In the case of the Stefan problem, the desired state is a particular interface position. An important method for the derivation of first order necessary optimality conditions is the Lagrange formalism. It results in an adjoint-based optimal control approach. By the help of a Lagrange functional, the adjoint system is derived and thereby the gradient of the cost functional with respect to the control variable. A detailed application of this technique for the Stefan problem in level set formulation can be found in [8,9]. However, the convection terms in the liquid phase are omitted. This is also the case in [22], where a simplified optimization approach for free boundary problems is presented and utilized for a Stefan-type problem. In contrast to these, in [17, 40], the Stefan problem is coupled with the Navier–Stokes equations and the convection is not driven by inflow and outflow boundaries but by Lorentz forces. A graph represents the inner boundary. The temperature, velocity, and pressure are approximated with finite differences. In [8, 40], the control acts directly on the temperature. Furthermore, they apply first order optimality conditions and gradient-based methods to approximate a control. Second order optimality conditions are proposed in [2, 3] for a more general free boundary problem. The latter two papers are mainly focused on the theoretical derivation of second order sufficient optimality conditions.

1.3. Research Goals

The aim of this thesis is to develop an optimal control approach for a two-dimensional two-phase Stefan problem fully coupled with Navier–Stokes equations. In both phases, the heat distribution is characterized by the heat equation. The fluid flow in the liquid phase is described by Navier–Stokes equations. The model considered in this work includes in- and outflow conditions which is an extension to the settings in the existing research.

In [6], finite elements and an unstructured triangular grid, which respects the interface, are used to approximate the solution of the Stefan problem. The Stefan condition is solved in a variational form for the interface velocity in normal direction. By solving a Laplace equation, this velocity is extended in a smooth way onto the whole mesh to prevent it from extreme deformation. These techniques are adapted in this thesis and combined with the adjoint-based optimal control approaches from [8, 40]. In order to have a sharp interface model, the moving boundary is represented as a graph like in [40]. A main contribution of the present work is the formulation of the adjoint system and first-order necessary optimality conditions following the "optimize-then-discretize" paradigm. In contrast to the given references, the control acts on the pressure at the inlet and not on the temperature. Thus, it affects the interface position in a more indirect way. Further, the mesh movement is directly included in the PDE system and the temperature and velocity are augmented by an additional mesh movement advection term as proposed in [6]. The quadratic tracking-type cost functional, which penalizes the deviation of the interface from the desired state as well as the control costs, is minimized with a projected gradient algorithm from [40]. A modified version of the quadratic line minimization algorithm [40] is used to compute the step size in the gradient method.
The forward and adjoint PDE systems are numerically solved using the finite element framework FEniCS [1]. With several numerical examples the performance of the proposed approach is illustrated and analyzed.

1.4. Outline

The thesis is structured as follows: In chapter 2, the equations forming the forward PDE system are described in detail. The domain is partitioned in a liquid and a solid phase separated by the moving interface. The system consists of the heat equation, the Navier–Stokes equations, and the equations for the mesh movement. The heat equation is coupled with the interface velocity through the Stefan condition.

This condition connects the normal velocity of the interface to the jump of the temperature gradient across the interface. Further, the mesh movement is embedded into the system and characterized by a Laplace equation. This is again coupled with the heat equation and the Navier–Stokes equations. The control acts at the inlet through steering the inflow pressure.

The optimal control problem is defined in chapter 3. In the beginning, the adjoint-based optimal control approach is formulated in an abstract way. By describing a cost functional and a Lagrange functional, the approach is applied to the concrete problem. The adjoint system is obtained through determining the derivatives of the Lagrange functional with respect to the single states defined in the forward system. With the backward system at hand, the gradient of the cost functional can be formulated and plugged into a projected gradient algorithm, equipped with a quadratic line minimization algorithm.

To be able to solve the forward and backward systems numerically, the equations and domain are discretized in chapter 4. The spatial discretization is done by a triangulation of the domain. The resulting mesh respects the interface position and is moved in every time step along the interface movement. To discretize the PDE systems in the forward and backward problems, their weak formulations are developed here.

Chapter 5 contains three numerical examples to illustrate the performance of the adjoint approach and the gradient algorithm. In the first example, the interface position is stabilized to a straight line, which, however, is not reachable by the system due to the asymmetric setup. In a second example, the interface is stabilized to a flat, not perfectly straight although reachable position. In the third example, the task is to move the interface upwards. For all three examples, the influence of different weight parameters in the cost functional and initial guesses is analyzed.

2. Two-Phase Stefan Problem

The considered model of the Stefan problem consists of the heat equation, including the Stefan condition and the graph representing the interface. Additionally, the Navier–Stokes equations and the mesh movement are integrated and fully coupled with the model. In the following section 2.1, the arrangement of the domain and its boundary regions are marked out. Moreover, the graph representation of the interface is defined in this section. The equations characterizing the temperature and the interface movement are described in detail in section 2.2 together with the corresponding boundary conditions. Analogously, the same is done for the mesh movement in section 2.3 and for the velocity and pressure in the liquid phase in section 2.4.

2.1. Domain

The domain $\Omega(t) \in \mathbb{R}^2$ is split into the solid phase $\Omega_s(t)$ and the liquid phase $\Omega_l(t)$, which are separated by the interface $\Gamma_I(t)$. The inflow and the heating with the temperature $T_H(t)$ are located at $\Gamma_H(t)$, the outflow at $\Gamma_{out}(t)$. At the bottom, there is the cooling boundary $\Gamma_C(t) = [a,b] \subset \mathbb{R}$ with the cooling temperature $T_C(t)$. The remaining parts of the outer boundary are denoted $\Gamma_N(t)$ and $\Gamma_{\tilde{N}}(t) \subset \Gamma_N(t)$ is at the top. The inner boundary $\Gamma_I(t)$ moves so that its position is time-dependent. Thus, the solid and liquid phases are time-dependent as are their boundaries. For the sake of brevity, the time-dependence "(t)" is dropped in most places throughout this thesis.

As in [40], it is assumed that the interface can be represented as a graph

$$\Gamma_I(t) = \left\{ \begin{bmatrix} x \\ h(t,x) \end{bmatrix} : x \in \Gamma_C \right\}, \qquad \text{with } h\colon [0,E] \times \Gamma_C \to \mathbb{R},$$

where $[0,E], E > 0$ is the time interval. The derivatives of h are abbreviated with

$$h_x := \frac{dh}{dx}, \qquad h_t := \partial_t h.$$

To map from Γ_C to the interface Γ_I, the function $\Phi\colon [0,E] \times \Gamma_C \to [0,E] \times \Gamma_I$ is used. It is defined as

$$\Phi(t,x) := \left(t, \begin{bmatrix} x \\ h(t,x) \end{bmatrix} \right).$$

The unit normal vector $\boldsymbol{\nu}$ along the interface Γ_I pointing from the solid to the liquid phase can be expressed as

$$\boldsymbol{\nu}(t,x) = \frac{1}{\sqrt{1 + h_x(t,x)^2}} \begin{bmatrix} -h_x(t,x) \\ 1 \end{bmatrix}. \tag{2.1}$$

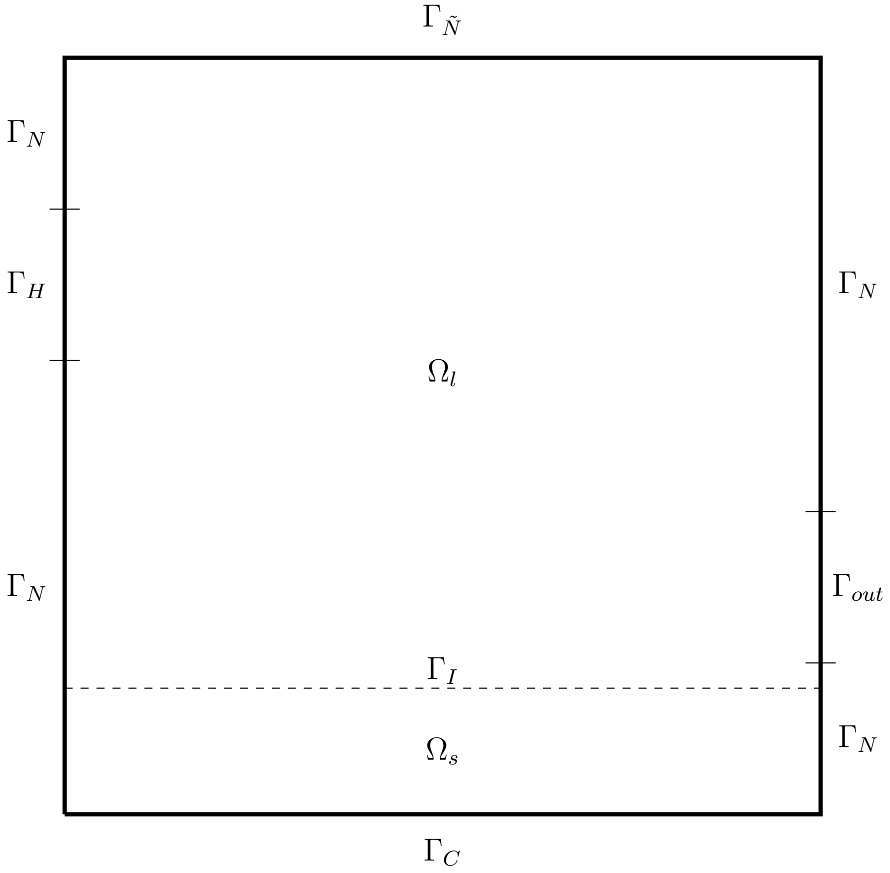


Figure 2.1.: The domain $\Omega \in \mathbb{R}^2$ for the Stefan problem.

2.2. Heat Equation

The temperature T in the solid and liquid phases is modeled with the heat equation:

$$\partial_t T + (u - V_{all}) \cdot \nabla T - \alpha \Delta T = 0, \qquad (0, E] \times \Omega,$$

where u is the velocity of the fluid and V_{all} the mesh movement. Additionally, the Stefan condition at the moving interface Γ_I couples the temperature with the velocity of the interface in normal direction:

$$[k_s(\nabla T)_s - k_l(\nabla T)_l] =: [k(\nabla T)]_l^s = L \cdot V_{\Gamma_I}, \qquad \text{on } \Gamma_I, \tag{2.2}$$

with k_s and k_l denoting the heat conductivities in the solid and liquid phases and L denoting the latent heat, and with $(\nabla T)_s := \partial_{\boldsymbol{\nu}} T\big|_{\Omega_s}$, $(\nabla T)_l := \partial_{-\boldsymbol{\nu}} T\big|_{\Omega_l}$. This equation can be used to determine V_{Γ_I} if T is already known.

With (2.1) the velocity of the interface Γ_I in normal direction V_{Γ_I} is

$$\begin{aligned} V_{\Gamma_I}(t,x) &= \partial_t \begin{bmatrix} x \\ h(t,x) \end{bmatrix} \cdot \boldsymbol{\nu}(t,x) = \begin{bmatrix} 0 \\ h_t(t,x) \end{bmatrix} \cdot \boldsymbol{\nu}(t,x) \\ &= \frac{h_t(t,x)}{\sqrt{1+h_x(t,x)^2}} = h_t(t,x)\boldsymbol{\nu}(t,x) \cdot \boldsymbol{e_2}, \end{aligned} \tag{2.3}$$

with $\boldsymbol{e_2} = [0,1]^\mathsf{T}$ the unit vector in vertical direction. For more details see (A.1). Using equation (2.3), the Stefan condition (2.2) can be reformulated to

$$\sqrt{1+h_x^2} \cdot [k(\nabla T)]_l^s \circ \Phi = L \cdot h_t, \qquad \text{on } \Gamma_C.$$

This reformulation will be needed to couple the whole system with the cost functional. The system now reads

$$\begin{aligned}
\partial_t T + (u - V_{all}) \cdot \nabla T - \alpha \Delta T &= 0, && \text{on } (0,E] \times \Omega, && \text{(2.4a)} \\
\sqrt{1+h_x^2} \cdot [k(\nabla T)]_l^s \circ \Phi - L \cdot h_t &= 0, && \text{on } (0,E] \times \Gamma_C, && \text{(2.4b)} \\
\sqrt{1+h_x^2} \cdot V_{\Gamma_I} \circ \Phi - h_t &= 0, && \text{on } (0,E] \times \Gamma_C, && \text{(2.4c)} \\
T &= T_H, && \text{on } (0,E] \times \Gamma_H, && \text{(2.4d)} \\
T &= T_C, && \text{on } (0,E] \times \Gamma_C, && \text{(2.4e)} \\
T &= T_m, && \text{on } (0,E] \times \Gamma_I, && \text{(2.4f)} \\
\partial_{\boldsymbol{\mu}} T &= 0, && \text{on } (0,E] \times (\Gamma_N \cup \Gamma_{out}), && \text{(2.4g)} \\
T(0) &= T_0, && \text{on } \Omega, && \text{(2.4h)} \\
V_{\Gamma_I}(0) &= 0, && \text{on } \Gamma_I, && \text{(2.4i)} \\
h(0) &= h_0, && \text{on } \Gamma_C. && \text{(2.4j)}
\end{aligned}$$

Equation (2.4c) links the interface graph h and the interface velocity V_{Γ_I}, (2.4d) and (2.4e) are the heating and cooling Dirichlet boundary conditions, and (2.4f) enforces T to have the characteristic melt temperature on the interface. Equation (2.4g) is the *do-nothing* condition [12] for the remaining boundary parts and (2.4h) - (2.4j) are the initial value conditions. T_0 and h_0 are the known initial values for T and h,

$$\alpha := \begin{cases} k_l, & \text{in } \Omega_l, \\ k_s, & \text{in } \Omega_s. \end{cases}$$

The mesh movement V_{all} and the velocity of the liquid u will be discussed in the following sections.

2.3. Mesh Movement

In the initial partition, the edges of the mesh are aligned with the interface Γ_I. To keep this for the next time step, the vertices on the interface must move with V_{Γ_I} in normal direction. In order to prevent the mesh from extreme deformation, V_{Γ_I} is smoothly extended to V_{all} on the whole domain Ω. For this, the following Laplace equation is solved:

$$\begin{aligned}
\Delta V_{all} &= 0, && \text{on } (0,E] \times \Omega, && (2.5a)\\
V_{all} - V_{\Gamma_I} \cdot \boldsymbol{\nu} &= 0, && \text{on } (0,E] \times \Gamma_I, && (2.5b)\\
V_{all} &= 0, && \text{on } (0,E] \times (\Gamma_C \cup \Gamma_{\tilde{N}}), && (2.5c)\\
V_{all} \cdot \boldsymbol{\mu} &= 0, && \text{on } (0,E] \times \partial\Omega, && (2.5d)\\
V_{all}(0) &= 0, && \text{on } \Omega. && (2.5e)
\end{aligned}$$

The second equation (2.5b) is a Dirichlet condition on the inner boundary Γ_I which ensures $V_{all} = V_{\Gamma_I} \cdot \boldsymbol{\nu}$ on Γ_I. Equations (2.5c) – (2.5e) assure that vertices on the outer boundary $\partial\Omega$ stay on the boundary. These conditions preserve the shape of the domain.

The interface Γ_I is a non-material surface. The movement V_{Γ_I} of the interface and the mesh movement V_{all} are not related to the movement of any physical material points. As already pointed out in [6], the non-material movement V_{all} needs to be removed from the material movement in T and u with advection terms

$$-V_{all} \cdot \nabla T$$

for the heat equation (2.4a) and

$$-(V_{all} \cdot \nabla) u$$

for the Navier–Stokes equations, which is denoted in the next section.

2.4. Navier–Stokes Equations

The velocity in the liquid phase u and the pressure p are described with the incompressible Navier–Stokes equations for Newtonian fluids [13]:

$$\begin{aligned}
\partial_t u + ((u - V_{all}) \cdot \nabla) u & \\
-\eta \Delta u + \nabla p &= 0, && \text{on } (0,E] \times \Omega_l, && (2.6a)\\
\nabla \cdot u &= 0, && \text{on } (0,E] \times \Omega_l, && (2.6b)\\
u &= 0, && \text{on } (0,E] \times \Gamma_I, && (2.6c)\\
u &= 0, && \text{on } (0,E] \times (\Gamma_N \cap \partial\Omega_l), && (2.6d)\\
p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u - p_{in} \cdot \boldsymbol{\mu} &= 0, && \text{on } (0,E] \times \Gamma_H, && (2.6e)\\
p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u &= 0, && \text{on } (0,E] \times (\Gamma_{out} \cap \partial\Omega_l), && (2.6f)\\
u(0) &= 0, && \text{on } \Omega_l, && (2.6g)\\
p(0) &= 0, && \text{on } \Omega_l. && (2.6h)
\end{aligned}$$

The constant η is the kinematic viscosity. Additionally to momentum and mass balance equations (2.6a) – (2.6b), equation (2.6f) defines a outflow boundary condition on $\Gamma_{out} \cap \partial\Omega_l$. The inflow boundary condition (2.6e) is influenced by the control variable p_{in}. Equations (2.6c) – (2.6d) are no-slip boundary conditions and (2.6g) – (2.6h) initial value conditions for the velocity u and the pressure p.

Notice that if u is used over the whole domain Ω, it is extended with 0 on Ω_s. The pressure at the inflow boundary p_{in} is the control variable. It is used to steer the system to a desired state with the optimal control approach explained in the following chapter.

3. Optimization

In the beginning of this chapter, the control problem and Lagrange formalism are introduced in a general setting. Afterwards, the derived formulation is adapted to the concrete problem in detail. What follows is closely orientated towards [8, 40]. More details on optimal control of partial differential equations can be found in [34].
The concrete control problem is defined in terms of the underlying PDE system in section 3.1.1 and a cost functional in section 3.1.2. The adjoint system is derived in section 3.2. For this, a Lagrange functional is required (section 3.2.1). The resulting adjoint system is summarized in section 3.2.3. To compute a control, which steers the interface to a desired position, a projected gradient algorithm combined with a quadratic line minimization algorithm are formulated in section 3.3.

3.1. Optimal Control Problem

For the state w from the state space $\mathcal{W}$ and the control c which is an element of the control space $\mathcal{C}$, the optimal control problem is defined as

$$\begin{aligned} &\min_{w\in\mathcal{W},c\in\mathcal{C}} J(w,c) \\ \text{subject to (s.t.)}& \\ &e(w,c)=0, \\ &c\in\mathcal{C}_{ad}\subset\mathcal{C}. \end{aligned} \tag{3.1}$$

The cost functional $J : \mathcal{W}\times\mathcal{C}\to\mathbb{R}$ can be used to steer the system to the desired state w_d. The state equation $e(w,c) = 0$ connects the state and the control. Further, the control constraint $c\in\mathcal{C}_{ad}$ defines restrictions to the control. The set of admissible controls $\mathcal{C}_{ad}$ is usually a convex subset of $\mathcal{C}$. In case of $\mathcal{C}_{ad} = \mathcal{C}$, the problem is unrestricted.
Assuming the existence of a unique $w(c)\in\mathcal{W}$ for every $c\in\mathcal{C}_{ad}$, that solves the state equation $e(w(u), c) = 0$, the *reduced* cost functional $\mathcal{K}(c) := J(w(c), c)$ together with an equivalent optimal control problem can be defined:

$$\begin{aligned} &\min_{c\in\mathcal{C}} \mathcal{K}(c) \\ \text{s.t.}& \\ &e(w(c),c)=0, \\ &c\in\mathcal{C}_{ad}\subset\mathcal{C}. \end{aligned} \tag{3.2}$$

In order to find a solution c^* for (3.2) first-order necessary optimality conditions are required. These can be derived formally by applying the Lagrange formalism.

For this, the Lagrange functional $\mathcal{L} : \mathcal{W} \times \mathcal{C} \times \mathcal{B} \to \mathbb{R}$ is defined as

$$\mathcal{L}(w, c, \zeta) := J(w, c) - e(w, c) \cdot \zeta.$$

The Lagrange multiplier $\zeta \in \mathcal{B}$ is also called adjoint state and $e : \mathcal{W} \times \mathcal{C} \to \mathcal{B}^*$, where $\mathcal{B}^*$ is usually the dual of a Banach space $\mathcal{B}$. From

$$\mathcal{L}_w(w, c, \zeta)\delta w = 0,$$

for all admissible directions of variation $\delta w \in \mathcal{W}$, the so called adjoint equation

$$e^*(w, c, \zeta) := J_w(w, c)\delta w - e_w(w, c)\delta w \cdot \zeta = 0 \tag{3.3}$$

follows. Since (3.3) holds for all δw, it also holds for all $w_c(c)\delta c$ with $\delta c \in \mathcal{C}$

$$J_w(w, c)w_c(c)\delta c = e_w(w, c)w_c(c)\delta c \cdot \zeta. \tag{3.4}$$

Further, for all $c \in \mathcal{C}_{ad}$ the state equation gives

$$e(w(c), c) = 0,$$

which leads to

$$0 = D_c e(w, c)\delta c = e_w(w, c)w_c(c)\delta c + e_c(w, c)\delta c$$

and

$$e_w(w, c)w_c(c)\delta c = -e_c(w, c)\delta c. \tag{3.5}$$

The insertion of (3.5) into (3.4) leads to

$$J_w(w, c)w_c(c)\delta c = -e_c(w, c)\delta c \cdot \zeta.$$

One finds, that the gradient of the cost functional $\mathcal{K}$ with respect to the control c can be expressed in terms of the Lagrange functional:

$$\begin{aligned}\mathcal{K}_c(c)\delta c =& J_c(w, c)\delta c + J_w(w, c)w_c(c)\delta c\\ =& J_c(w, c)\delta c - e_c(w, c)\delta c \cdot \zeta\\ =& \mathcal{L}_c(w, c, \zeta)\delta c.\end{aligned}$$

For a control c^*, a state w^*, and an adjoint state ζ^*, the *gradient condition* reads

$$\begin{aligned}&\langle \mathcal{L}_c(w^*, c^*, \zeta^*), v - c^* \rangle\\ &= \langle J_c(w^*, c^*) - e_c(w^*, c^*) \cdot \zeta^*, v - c^* \rangle \geq 0, \qquad \text{for all } v \in \mathcal{C}_{ad}.\end{aligned} \tag{3.6}$$

In the unrestricted case, where $\mathcal{C}_{ad} = \mathcal{C}$, the inequality (3.6) simplifies to the so called *gradient equation*

$$\mathcal{L}_c(w^*, c^*, \zeta^*) = J_c(w^*, c^*) - e_c(w^*, c^*) \cdot \zeta^* = 0. \tag{3.7}$$

Accordingly, the *first order optimality conditions* read

$$\begin{aligned} e(w,c) &= 0, && \text{with } w = w(c),\\ e^*(w,c,\zeta) &= 0, && \text{with } \zeta = \zeta(c),\\ \langle \mathcal{L}_c(w,c,\zeta), v-c\rangle &\geq 0, && \text{for all } v \in \mathcal{C}_{ad},\\ c \in \mathcal{C}_{ad} &\subset \mathcal{C}. \end{aligned} \tag{3.8}$$

These are necessary optimality conditions. A control that fulfills (3.8) could be a maximum of the cost functional $\mathcal{K}$ instead of a minimum. If a control fulfills these conditions and minimizes $\mathcal{K}$ locally, (3.8) does not state any information about globality of the minimum. Nevertheless, (3.8) can be formulated for the present Stefan problem together with the aim to steer the interface to a desired position. The inflow pressure $c = p_{in}$ is the control variable and $w = [h, T, V_{\Gamma_I}, V_{all}, u, p]$ are the states. Concrete formulations of the state equation, the cost functional and the adjoint equation follow in the next sections.

3.1.1. Forward System

In case of the Stefan Problem, the state equation in (3.2) is a system of partial differential equations (PDEs) described in chapter 2. The whole system reads as follows. For a given control p_{in} find functions T (temperature), h (interface graph), V_{Γ_I} (interface velocity), V_{all} (mesh movement), u (velocity) and p (pressure) such that

$$\begin{array}{lrl}
\mathrm{HEAT}(T,u,V_{all}) := & \partial_t T + (u - V_{all})\cdot\nabla T - \alpha\Delta T = 0, & \text{on } (0,E]\times\Omega,\\
\mathrm{INT}(T,h) := & \sqrt{1+h_x^2}\cdot[k(\nabla T)]_l^s \circ \Phi - L\cdot h_t = 0, & \text{on } (0,E]\times\Gamma_C,\\
\mathrm{VEL}(h,V_{\Gamma_I}) := & \sqrt{1+h_x^2}\cdot V_{\Gamma_I}\circ\Phi - h_t = 0, & \text{on } (0,E]\times\Gamma_C,\\
 & T = T_H, & \text{on } (0,E]\times\Gamma_H,\\
 & T = T_C, & \text{on } (0,E]\times\Gamma_C,\\
\mathrm{TEMP}(T) := & T - T_m = 0, & \text{on } (0,E]\times\Gamma_I,\\
 & \partial_{\boldsymbol{\mu}} T = 0, & \text{on } (0,E]\times(\Gamma_N\cup\Gamma_{out}),\\
 & T(0) = T_0, & \text{on } \Omega,\\
 & V_{\Gamma_I}(0) = 0, & \text{on } \Gamma_I,\\
 & h(0) = h_0, & \text{on } \Gamma_C,\\
\mathrm{MESH}(V_{all}) := & \Delta V_{all} = 0, & \text{on } (0,E]\times\Omega,\\
\mathrm{MOVE}(V_{all},V_{\Gamma_I}) := & V_{all} - V_{\Gamma_I}\cdot\boldsymbol{\nu} = 0, & \text{on } (0,E]\times\Gamma_I,\\
 & V_{all} = 0, & \text{on } (0,E]\times(\Gamma_C\cup\Gamma_{\tilde{N}}),\\
 & V_{all}\cdot\boldsymbol{\mu} = 0, & \text{on } (0,E]\times\partial\Omega,\\
 & V_{all}(0) = 0, & \text{on } \Omega, \qquad\qquad (3.9)\\
\mathrm{NSE}(u,p,V_{all}) := & \partial_t u + ((u - V_{all})\cdot\nabla)u & \\
 & -\eta\Delta u + \nabla p = 0, & \text{on } (0,E]\times\Omega_l,\\
\mathrm{DIV}(u) := & \nabla\cdot u = 0, & \text{on } (0,E]\times\Omega_l,\\
 & u = 0, & \text{on } (0,E]\times\Gamma_I,
\end{array}$$

$$\begin{aligned}
& & u &= 0, && \text{on } (0,E] \times (\Gamma_N \cap \partial\Omega_l),\\
&\mathrm{CONT}(p,u,p_{in}) := & p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u - p_{in} \cdot \boldsymbol{\mu} &= 0, && \text{on } (0,E] \times \Gamma_H,\\
&\mathrm{OUT}(p,u) := & p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u &= 0, && \text{on } (0,E] \times (\Gamma_{out} \cap \partial\Omega_l),\\
& & u(0) &= 0, && \text{on } \Omega_l,\\
& & p(0) &= 0, && \text{on } \Omega_l.
\end{aligned}$$

Throughout this thesis the latter system (3.9) is named the forward system. For later reference, we introduce the terms HEAT, INT, ..., OUT.

3.1.2. Cost Functional

In this section, a quadratic tracking-type cost functional is described which expresses the aim to steer the position of the interface to a desired one. The graph h_d describes the desired position of the interface Γ_I. The scalars Λ, $\bar{\Lambda}$, and λ are weight factors for the cost functional J

$$\begin{aligned}
J(h,p_{in}) :=& \frac{\Lambda}{2} \int_{\Gamma_C} (h(E,x) - h_d(E,x))^2 \, dx + \frac{\lambda}{2} \int_0^E \int_{\Gamma_H} (p_{in}(t))^2 \, dxdt \\
&+ \frac{\bar{\Lambda}}{2} \int_0^E \int_{\Gamma_C} (h(t,x) - h_d(t,x))^2 \, dxdt.
\end{aligned}$$

The first term aims to steer the interface position to the desired position at terminal time E, while the third term monitors the interface movement over the complete time horizon $(0,E]$. The second term models control costs and has a regularizing effect. By adjusting λ, constraints on the control p_{in} can become unnecessary. This implies, that restrictions can be dropped and $\mathcal{C}_{ad} = \mathcal{C}$.

So far, no proof of the existence and uniqueness of a solution to the forward system (3.9) is known [8,40]. Thus, the existence of an optimal control can not be guaranteed. Therefore, the optimal control techniques applied in this thesis are only formal. It is assumed that for every $p_{in} \in \mathcal{C}_{ad}$, unique states $T(p_{in})$, $h(p_{in})$, $V_{\Gamma_I}(p_{in})$, $V_{all}(p_{in})$, $u(p_{in})$, and $p(p_{in})$ exist. As a consequence, the cost functional can be reformulated as in section 3.1

$$\mathcal{K}(p_{in}) := J(h(p_{in}), p_{in}). \tag{3.10}$$

The only missing elements to formulate an optimal control problem in the terminology of (3.8) are the Lagrange functional and the adjoint equation.

3.2. Derivation of the Adjoint System

The Lagrange functional is defined in this section. It is only required as a tool to derive the adjoint equations. For the sake of brevity dx, ds, and dt are omitted in what follows.

3.2.1. Lagrange Functional

As in the formal derivation of (3.8) in section 3.1, the Lagrange functional consists of the cost functional and the term $-e(w,c)\cdot\zeta$, which represents the PDE-constraints multiplied with a Lagrange multiplier ζ. In the considered case, $e(w,c)$ is the sum of particular terms from (3.9). Those equations from (3.9) that are not included are treated explicitly as conditions to the directions of variation (3.14). Let $w = [h, T, V_{\Gamma_I}, V_{all}, u, p]$ be the tuple of states and let $\zeta = [\omega, \omega^I, \psi, \psi^C, \psi_{all}, \psi^I_{all}, \gamma,\ \pi, \varphi, \gamma^{out}]$ be the tuple of adjoint states. Then we define, with the terms introduced in (3.9), the Lagrange functional:

$$
\begin{aligned}
\mathcal{L}(h, T, &V_{\Gamma_I}, V_{all}, u, p, p_{in}, \omega, \omega^I, \psi, \psi^C, \psi_{all}, \psi^I_{all}, \gamma, \pi, \varphi, \gamma^{out}) := \\
&\frac{\Lambda}{2}\int_{\Gamma_C} (h(E,x) - h_d(E,x))^2 + \frac{\lambda}{2}\int_0^E\int_{\Gamma_H} (p_{in}(t))^2 \\
&+ \frac{\bar{\Lambda}}{2}\int_0^E\int_{\Gamma_C} (h(t,x) - h_d(t,x))^2 \\
&- \int_0^E\int_{\Omega} \mathrm{HEAT}(T, u, V_{all})\cdot\omega - \int_0^E\int_{\Gamma_C} \mathrm{INT}(T,h)\cdot\psi \\
&- \int_0^E\int_{\Gamma_C} \mathrm{VEL}(h, V_{\Gamma_I})\cdot\psi^C - \int_0^E\int_{\Gamma_I} \mathrm{TEMP}(T)\cdot\omega^I \\
&- \int_0^E\int_{\Omega} \mathrm{MESH}(V_{all})\cdot\psi_{all} - \int_0^E\int_{\Gamma_I} \mathrm{MOVE}(V_{all}, V_{\Gamma_I})\cdot\psi^I_{all} \\
&- \int_0^E\int_{\Omega_l} \mathrm{NSE}(u, p, V_{all})\cdot\gamma - \int_0^E\int_{\Omega_l} \mathrm{DIV}(u)\cdot\pi \\
&- \int_0^E\int_{\Gamma_H} \mathrm{CONT}(p, u, p_{in})\cdot\varphi - \int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l} \mathrm{OUT}(p,u)\cdot\gamma^{out}.
\end{aligned}
\tag{3.11}
$$

As shown in section 3.1, the derivatives of $\mathcal{L}$ with respect to the states $w = [h, T, V_{\Gamma_I}, V_{all}, u, p]$ can be used to derive the adjoint system.

3.2.2. Adjoint Equations

The normal derivative on the interface Γ_I,

$$(\partial_{\boldsymbol{\nu}} T)_{s/l} = (\nabla T)_{s/l},$$

mapped onto Γ_C, using (A.2) and following [40, p. 163] is

$$(\nabla T)_{s/l} \circ \Phi = \sqrt{1 + h_x^2} (\partial_y T)_{s/l} \circ \Phi, \qquad \text{on } \Gamma_C. \tag{3.12}$$

With (3.12), $\mathrm{INT}(T, h)$ becomes

$$\begin{aligned} \mathrm{INT}(T, h) &= \sqrt{1 + h_x^2} \cdot [k_s (\nabla T)_s - k_l (\nabla T)_l] \circ \Phi - L \cdot h_t \\ &= (1 + h_x^2) \cdot [k_s (\partial_y T)_s - k_l (\partial_y T)_l] \circ \Phi - L \cdot h_t. \end{aligned} \tag{3.13}$$

As explained in (3.3), the derivative of the Lagrange functional with respect to the states is set equal to zero

$$D_{[h,T,V_{\Gamma_I},V_{all},u,p]} \mathcal{L}[\delta h, \delta T, \delta V_{\Gamma_I}, \delta V_{all}, \delta u, \delta p] = 0,$$

where $\delta h, \ldots, \delta p$ denotes the directions of variation. All equations from (3.9), which do not occur as terms in the Lagrange functional (3.11), are treated as explicit conditions on the directions of variation:

$$\begin{aligned} \delta T &= 0, && \text{on } (0, E] \times (\Gamma_C \cup \Gamma_H), \\ \partial_{\boldsymbol{\mu}} \delta T &= 0, && \text{on } (0, E] \times (\Gamma_N \cup \Gamma_{out}), \\ \delta T(0) &= 0, && \text{on } \Omega, \\ \delta V_{\Gamma_I}(0) &= 0, && \text{on } \Gamma_I, \\ \delta h(0) &= 0, && \text{on } \Gamma_C, \\ \delta V_{all} &= 0, && \text{on } (0, E] \times (\Gamma_C \cup \Gamma_{\tilde{N}}), \\ \delta V_{all}(0) &= 0, && \text{on } \Omega, \\ \delta V_{all} \cdot \boldsymbol{\mu} &= 0, && \text{on } (0, E] \times \partial\Omega, \\ \delta u &= 0, && \text{on } (0, E] \times (\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)), \\ \delta u(0) &= 0, && \text{on } \Omega_l, \\ \delta p(0) &= 0, && \text{on } \Omega_l. \end{aligned} \tag{3.14}$$

These conditions must be ensured when computing the derivatives of $\mathcal{L}$ with respect to the particular states in the following sections.

The Derivative with Respect to the Temperature T

We apply integration by parts as well as (3.14) to the variation of the third integral in the Lagrange functional (3.11) with respect to the temperature. This leads to

$$\begin{aligned}
&D_T\Bigg(-\int_0^E\int_\Omega \mathrm{HEAT}(T,u,V_{all})\cdot\omega\Bigg)\cdot\delta T\\
=&D_T\Bigg(-\int_0^E\int_\Omega (\partial_t T+(u-V_{all})\cdot\nabla T-\alpha\Delta T)\cdot\omega\Bigg)\cdot\delta T\\
=&-\int_0^E\int_\Omega \partial_t\delta T\cdot\omega-\int_0^E\int_\Omega (u-V_{all})\cdot\nabla\delta T\cdot\omega\\
&+\int_0^E\int_\Omega \alpha\Delta\delta T\cdot\omega.
\end{aligned}$$

For better readability, this equation is split into the first integral

$$\begin{aligned}
-\int_0^E\int_\Omega \partial_t\delta T\cdot\omega =& -\int_0^E\int_{\Omega_s}\partial_t\delta T\cdot\omega-\int_0^E\int_{\Omega_l}\partial_t\delta T\cdot\omega\\
=&-\int_{\Omega_s}\omega(E)\delta T(E)+\int_{\Omega_s}\omega(0)\underbrace{\delta T(0)}_{\overset{(3.14)}{=}0}+\int_0^E\int_{\Omega_s}\partial_t\omega\cdot\delta T\\
&-\int_{\Omega_l}\omega(E)\delta T(E)+\int_{\Omega_l}\omega(0)\underbrace{\delta T(0)}_{\overset{(3.14)}{=}0}+\int_0^E\int_{\Omega_l}\partial_t\omega\cdot\delta T\\
=&-\int_{\Omega}\omega(E)\delta T(E)+\int_0^E\int_{\Omega}\partial_t\omega\cdot\delta T,
\end{aligned}$$

the second integral

$$\begin{aligned}
&-\int_0^E\int_\Omega (u-V_{all})\cdot\nabla\delta T\cdot\omega\\
=&-\int_0^E\int_{\Omega_s}(u-V_{all})\cdot\nabla\delta T\cdot\omega-\int_0^E\int_{\Omega_l}(u-V_{all})\cdot\nabla\delta T\cdot\omega\\
=&-\int_0^E\int_{\partial\Omega_s}(u-V_{all})\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T+\int_0^E\int_{\Omega_s}(u-V_{all})\cdot\nabla\omega\cdot\delta T
\end{aligned}$$

$$
\begin{aligned}
&-\int_0^E\int_{\partial\Omega_l}(u-V_{all})\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T+\int_0^E\int_{\Omega_l}(u-V_{all})\cdot\nabla\omega\cdot\delta T\\
={}&\int_0^E\int_{\Omega}(u-V_{all})\cdot\nabla\omega\cdot\delta T-\int_0^E\int_{\Gamma_I}(u-V_{all})\cdot(\omega\cdot\underbrace{\boldsymbol{\mu}}_{=\,\boldsymbol{\nu}})\cdot\delta T\Big|_{\Omega_s}\\
&-\int_0^E\int_{\Gamma_I}(u-V_{all})\cdot(\omega\cdot\underbrace{\boldsymbol{\mu}}_{=\,-\boldsymbol{\nu}})\cdot\delta T\Big|_{\Omega_l}+\int_0^E\int_{\partial\Omega_s\setminus\Gamma_I}\omega\underbrace{(V_{all}\cdot\boldsymbol{\mu})}_{\overset{(2.5)}{=}\,0}\cdot\delta T\\
&+\int_0^E\int_{\partial\Omega_l\setminus\Gamma_I}\omega\underbrace{(V_{all}\cdot\boldsymbol{\mu})}_{\overset{(2.5)}{=}\,0}\cdot\delta T-\int_0^E\int_{\partial\Omega_s\setminus\Gamma_I}u\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T\\
&-\int_0^E\int_{\partial\Omega_l\setminus\Gamma_I}u\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T\\
={}&\int_0^E\int_{\Omega}(u-V_{all})\cdot\nabla\omega\cdot\delta T-\int_0^E\int_{\Gamma_I}(u-V_{all})\cdot(\omega\cdot\boldsymbol{\nu})\cdot\delta T\\
&+\int_0^E\int_{\Gamma_I}(u-V_{all})\cdot(\omega\cdot\boldsymbol{\nu})\cdot\delta T-\int_0^E\int_{\Gamma_C\cup\Gamma_N\cup(\Gamma_{out}\cap\partial\Omega_s)}\underbrace{u}_{\overset{(2.6a)}{=}\,0}\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T\\
&-\int_0^E\int_{\Gamma_H}u\cdot(\omega\cdot\boldsymbol{\mu})\cdot\underbrace{\delta T}_{\overset{(3.14)}{=}\,0}-\int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l}u\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T\\
={}&\int_0^E\int_{\Omega}(u-V_{all})\cdot\nabla\omega\cdot\delta T-\int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l}u\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T,
\end{aligned}
$$

and the third integral

$$
\begin{aligned}
\int_0^E\int_{\Omega}\alpha\Delta\delta T\cdot\omega={}&\int_0^E\int_{\Omega_s}k_s\Delta\delta T\cdot\omega+\int_0^E\int_{\Omega_l}k_l\Delta\delta T\cdot\omega\\
={}&\int_0^E\int_{\partial\Omega_s}k_s\omega\partial_{\boldsymbol{\mu}}\delta T-\int_0^E\int_{\Omega_s}k_s\nabla\omega\cdot\nabla\delta T\\
&+\int_0^E\int_{\partial\Omega_l}k_l\omega\partial_{\boldsymbol{\mu}}\delta T-\int_0^E\int_{\Omega_l}k_l\nabla\omega\cdot\nabla\delta T
\end{aligned}
$$

$$
\begin{aligned}
&= \int_0^E \int_{\Gamma_I} k_s \omega (\partial_{\boldsymbol{\mu}} \delta T)_s + \int_0^E \int_{\Gamma_I} k_l \omega (\partial_{\boldsymbol{\mu}} \delta T)_l \\
&\quad + \int_0^E \int_{\Gamma_N \cup \Gamma_{out}} \alpha \omega \underbrace{\partial_{\boldsymbol{\mu}} \delta T}_{\overset{(3.14)}{=} 0} + \int_0^E \int_{\Gamma_H \cup \Gamma_C} \alpha \omega \partial_{\boldsymbol{\mu}} \delta T \\
&\quad - \int_0^E \int_{\partial \Omega_s} k_s \partial_{\boldsymbol{\mu}} \omega \delta T + \int_0^E \int_{\Omega_s} k_s \Delta \omega \delta T \\
&\quad - \int_0^E \int_{\partial \Omega_l} k_l \partial_{\boldsymbol{\mu}} \omega \delta T + \int_0^E \int_{\Omega_l} k_l \Delta \omega \delta T \\
&= \int_0^E \int_{\Gamma_I} \omega k_s (\partial_{\boldsymbol{\nu}} \delta T)_s - \int_0^E \int_{\Gamma_I} \omega k_l (\partial_{\boldsymbol{\nu}} \delta T)_l \\
&\quad - \int_0^E \int_{\Gamma_I} k_s (\partial_{\boldsymbol{\mu}} \omega)_s \delta T - \int_0^E \int_{\Gamma_I} k_l (\partial_{\boldsymbol{\mu}} \omega)_l \delta T \\
&\quad + \int_0^E \int_{\Omega} \alpha \Delta \omega \delta T - \int_0^E \int_{\Gamma_H \cup \Gamma_C} \alpha \partial_{\boldsymbol{\mu}} \omega \underbrace{\delta T}_{\overset{(3.14)}{=} 0} \\
&\quad - \int_0^E \int_{\Gamma_N \cup \Gamma_{out}} \alpha \partial_{\boldsymbol{\mu}} \omega \delta T + \int_0^E \int_{\Gamma_H \cup \Gamma_C} \alpha \omega \partial_{\boldsymbol{\mu}} \delta T \\
&= \int_0^E \int_{\Omega} \alpha \Delta \omega \delta T + \int_0^E \int_{\Gamma_I} \omega [k_s (\partial_{\boldsymbol{\nu}} \delta T)_s - k_l (\partial_{\boldsymbol{\nu}} \delta T)_l] \\
&\quad - \int_0^E \int_{\Gamma_I} [k_s (\partial_{\boldsymbol{\nu}} \omega)_s - k_l (\partial_{\boldsymbol{\nu}} \omega)_l] \delta T - \int_0^E \int_{\Gamma_N \cup \Gamma_{out}} \alpha \partial_{\boldsymbol{\mu}} \omega \delta T \\
&\quad + \int_0^E \int_{\Gamma_H \cup \Gamma_C} \alpha \omega \partial_{\boldsymbol{\mu}} \delta T.
\end{aligned}
$$

The latter three equations are combined again and now yield

$$
\begin{aligned}
& D_T \left(- \int_0^E \int_{\Omega} \mathrm{HEAT}(T, u, V_{all}) \cdot \omega \right) \cdot \delta T \\
&= - \int_{\Omega} \omega(E) \delta T(E) + \int_0^E \int_{\Omega} \partial_t \omega \cdot \delta T
\end{aligned}
$$

$$
\begin{aligned}
&+\int_0^E\int_\Omega (u-V_{all})\cdot\nabla\omega\cdot\delta T-\int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l} u\cdot(\omega\cdot\boldsymbol{\mu})\cdot\delta T\\
&+\int_0^E\int_\Omega \alpha\Delta\omega\delta T+\int_0^E\int_{\Gamma_I}\omega[k_s(\partial_{\boldsymbol{\nu}}\delta T)_s-k_l(\partial_{\boldsymbol{\nu}}\delta T)_l]\\
&-\int_0^E\int_{\Gamma_I}[k_s(\partial_{\boldsymbol{\nu}}\omega)_s-k_l(\partial_{\boldsymbol{\nu}}\omega)_l]\delta T-\int_0^E\int_{\Gamma_N\cup\Gamma_{out}}\alpha\partial_{\boldsymbol{\mu}}\omega\delta T\\
&+\int_0^E\int_{\Gamma_H\cup\Gamma_C}\alpha\omega\partial_{\boldsymbol{\mu}}\delta T\\
=&\int_0^E\int_\Omega(\partial_t\omega+(u-V_{all})\cdot\nabla\omega+\alpha\Delta\omega)\cdot\delta T-\int_\Omega\omega(E)\delta T(E)\\
&+\int_0^E\int_{\Gamma_I}\omega[k(\nabla\delta T)]_l^s-\int_0^E\int_{\Gamma_I}[k(\nabla\omega)]_l^s\delta T+\int_0^E\int_{\Gamma_H\cup\Gamma_C}\alpha\omega\partial_{\boldsymbol{\mu}}\delta T \qquad (3.15)\\
&-\int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l}(\alpha\partial_{\boldsymbol{\mu}}\omega+u\cdot(\omega\cdot\boldsymbol{\mu}))\cdot\delta T-\int_0^E\int_{\Gamma_N\cup(\Gamma_{out}\cap\partial\Omega_s)}\alpha\partial_{\boldsymbol{\mu}}\omega\delta T.
\end{aligned}
$$

Thus, the variation of the Lagrange functional with respect to T together with (3.15) reads

$$
\begin{aligned}
0=&D_T\mathcal{L}\delta T=D_T\Bigg(-\int_0^E\int_\Omega \mathrm{HEAT}(T,u,V_{all})\cdot\omega\Bigg)\cdot\delta T\\
&+D_T\Bigg(-\int_0^E\int_{\Gamma_C}\mathrm{INT}(T,h)\cdot\psi\Bigg)\cdot\delta T+D_T\Bigg(-\int_0^E\int_{\Gamma_I}\mathrm{TEMP}(T)\cdot\omega^I\Bigg)\cdot\delta T\\
=&\int_0^E\int_\Omega(\partial_t\omega+(u-V_{all})\cdot\nabla\omega+\alpha\Delta\omega)\cdot\delta T-\int_\Omega\omega(E)\delta T(E)\\
&+\int_0^E\int_{\Gamma_C}\omega\circ\Phi[k(\nabla\delta T)]_l^s\circ\Phi-\int_0^E\int_{\Gamma_I}[k(\nabla\omega)]_l^s\delta T+\int_0^E\int_{\Gamma_H\cup\Gamma_C}\alpha\omega\partial_{\boldsymbol{\mu}}\delta T\\
&-\int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l}(\alpha\partial_{\boldsymbol{\mu}}\omega+u\cdot(\omega\cdot\boldsymbol{\mu}))\cdot\delta T-\int_0^E\int_{\Gamma_N\cup(\Gamma_{out}\cap\partial\Omega_s)}\alpha\partial_{\boldsymbol{\mu}}\omega\delta T
\end{aligned}
$$

$$
\begin{aligned}
&- \int_0^E \int_{\Gamma_C} \sqrt{1+h_x^2} \cdot \psi \cdot [k(\nabla \delta T)]_l^s \circ \Phi - \int_0^E \int_{\Gamma_I} \omega^I \cdot \delta T \\
= &\int_0^E \int_{\Omega} (\partial_t \omega + (u - V_{all}) \cdot \nabla \omega + \alpha \Delta \omega) \cdot \delta T - \int_{\Omega} \omega(E) \delta T(E) \\
&- \int_0^E \int_{\Gamma_{out} \cap \partial \Omega_l} (\alpha \partial_{\boldsymbol{\mu}} \omega + u \cdot (\omega \cdot \boldsymbol{\mu})) \cdot \delta T - \int_0^E \int_{\Gamma_N \cup (\Gamma_{out} \cap \partial \Omega_s)} \alpha \partial_{\boldsymbol{\mu}} \omega \delta T \\
&+ \int_0^E \int_{\Gamma_C} (\omega \circ \Phi - \sqrt{1+h_x^2} \cdot \psi) \cdot [k(\nabla \delta T)]_l^s \circ \Phi \\
&- \int_0^E \int_{\Gamma_I} (\omega^I + [k(\nabla \omega)]_l^s) \cdot \delta T + \int_0^E \int_{\Gamma_H \cup \Gamma_C} \alpha \omega \partial_{\boldsymbol{\mu}} \delta T.
\end{aligned}
$$

By proper variation of δT, certain terms can be eliminated from the prior equation. Thereby, terms which are integrated over the same domain and have the same multiplier on the right can be consolidated into one equation so that the following adjoint equations arise

$$\partial_t \omega + (u - V_{all}) \cdot \nabla \omega + \alpha \Delta \omega = 0, \qquad \text{on } [0, E) \times \Omega \tag{3.16a}$$

$$\alpha \partial_{\boldsymbol{\mu}} \omega + u \cdot (\omega \cdot \boldsymbol{\mu}) = 0, \qquad \text{on } [0, E) \times (\Gamma_{out} \cap \partial \Omega_l) \tag{3.16b}$$

$$\partial_{\boldsymbol{\mu}} \omega = 0, \qquad \text{on } [0, E) \times (\Gamma_N \cup (\Gamma_{out} \cap \partial \Omega_s)) \tag{3.16c}$$

$$\omega = 0, \qquad \text{on } [0, E) \times (\Gamma_C \cup \Gamma_H) \tag{3.16d}$$

$$\omega \circ \Phi - \sqrt{1+h_x^2} \cdot \psi = 0, \qquad \text{on } [0, E) \times \Gamma_C \tag{3.16e}$$

$$\omega^I + [k(\nabla \omega)]_l^s = 0, \qquad \text{on } [0, E) \times \Gamma_I \tag{3.16f}$$

$$\omega(E) = 0, \qquad \text{on } \Omega. \tag{3.16g}$$

The latter equations are the adjoint system for the adjoint state ω which can be interpreted as the adjoint temperature variable. The first equation (3.16a) is similar to the heat equation, while the equation (3.16f) is analogue to the Stefan condition. The other equations can be understood as boundary conditions and the initial condition at time $t = E$. The sole source term in these equations is $\sqrt{1+h_x^2} \cdot \psi$ in equation (3.16e), which realizes the coupling to the adjoint state ψ and through this to the distance terms in the cost functional (3.10).

The Derivative with Respect to the Interface Velocity V_{Γ_I}

The variation of the interface V_{Γ_I} with respect to the normal velocity is

$$
\begin{aligned}
0 =& D_{V_{\Gamma_I}}\mathcal{L}\delta V_{\Gamma_I} = D_{V_{\Gamma_I}}\Bigg(-\int_0^E\int_{\Gamma_C} \mathrm{VEL}(h, V_{\Gamma_I})\cdot\psi^C\Bigg)\delta V_{\Gamma_I} \\
&+ D_{V_{\Gamma_I}}\Bigg(-\int_0^E\int_{\Gamma_I} \mathrm{MOVE}(V_{all}, V_{\Gamma_I})\cdot\psi^I_{all}\Bigg)\delta V_{\Gamma_I} \\
=& -\int_0^E\int_{\Gamma_C} \sqrt{1+h_x^2}\cdot\delta V_{\Gamma_I}\circ\Phi\cdot\psi^C + \int_0^E\int_{\Gamma_I} \delta V_{\Gamma_I}\cdot\boldsymbol{\nu}\cdot\psi^I_{all} \\
=& -\int_0^E\int_{\Gamma_C} \sqrt{1+h_x^2}\cdot\psi^C\cdot\delta V_{\Gamma_I}\circ\Phi + \int_0^E\int_{\Gamma_C} (\boldsymbol{\nu}\cdot\psi^I_{all})\circ\Phi\cdot\delta V_{\Gamma_I}\circ\Phi \\
=& -\int_0^E\int_{\Gamma_C} (\sqrt{1+h_x^2}\cdot\psi^C - (\boldsymbol{\nu}\cdot\psi^I_{all})\circ\Phi)\cdot\delta V_{\Gamma_I}\circ\Phi
\end{aligned}
$$

from which the adjoint equation

$$
\sqrt{1+h_x^2}\cdot\psi^C = (\boldsymbol{\nu}\cdot\psi^I_{all})\circ\Phi, \qquad [0,E)\times\Gamma_C, \tag{3.17}
$$

results.

The Derivative with Respect to the Mesh Movement V_{all}

Applying integration by parts and (3.14) to the variation of the eighth integral of the Lagrange functional (3.11) leads to

$$
\begin{aligned}
& D_{V_{all}}\Bigg(-\int_0^E\int_{\Omega} \mathrm{MESH}(V_{all})\cdot\psi_{all}\Bigg)\cdot\delta V_{all} = -\int_0^E\int_{\Omega} \Delta\delta V_{all}\cdot\psi_{all} \\
=& -\int_0^E\int_{\partial\Omega} \psi_{all}\cdot\partial_{\boldsymbol{\mu}}\delta V_{all} + \int_0^E\int_{\Omega} \nabla\psi_{all}\cdot\nabla\delta V_{all} \\
=& -\int_0^E\int_{\partial\Omega} \psi_{all}\cdot\partial_{\boldsymbol{\mu}}\delta V_{all} + \int_0^E\int_{\partial\Omega} \partial_{\boldsymbol{\mu}}\psi_{all}\cdot\delta V_{all} \\
& -\int_0^E\int_{\Omega} \Delta\psi_{all}\cdot\delta V_{all}
\end{aligned}
$$

$$\begin{aligned}
&= -\int_0^E \int_\Omega \Delta\psi_{all} \cdot \delta V_{all} - \int_0^E \int_{\partial\Omega} \psi_{all} \cdot \partial_{\boldsymbol{\mu}} \delta V_{all} \\
&\quad + \int_0^E \int_{\Gamma_C \cup \Gamma_{\tilde{N}}} \partial_{\boldsymbol{\mu}} \psi_{all} \cdot \underbrace{\delta V_{all}}_{\overset{(3.14)}{=} 0} + \int_0^E \int_{\Gamma_H \cup \Gamma_{out} \cup (\Gamma_N \setminus \Gamma_{\tilde{N}})} \partial_{\boldsymbol{\mu}} \psi_{all} \cdot \delta V_{all}.
\end{aligned} \tag{3.18}$$

Analogously, the variation of the tenth integral in the Lagrange functional (3.11) is

$$\begin{aligned}
D_{V_{all}} \left(-\int_0^E \int_{\Omega_l} \mathrm{NSE}(u, p, V_{all}) \cdot \gamma \right) \cdot \delta V_{all} &= \int_0^E \int_{\Omega_l} (\delta V_{all} \cdot \nabla) u \cdot \gamma \\
&= \int_0^E \int_{\Omega_l} (\nabla u)^T \cdot \gamma \cdot \delta V_{all}.
\end{aligned} \tag{3.19}$$

Together with (3.18) and (3.19), the variation of the Lagrange functional (3.11) with respect to the mesh movement V_{all} is

$$\begin{aligned}
0 =& D_{V_{all}} \mathcal{L} \delta V_{all} = D_{V_{all}} \left(-\int_0^E \int_\Omega \mathrm{HEAT}(T, u, V_{all}) \cdot \omega \right) \cdot \delta V_{all} \\
&+ D_{V_{all}} \left(-\int_0^E \int_\Omega \mathrm{MESH}(V_{all}) \cdot \psi_{all} \right) \cdot \delta V_{all} \\
&+ D_{V_{all}} \left(-\int_0^E \int_{\Gamma_I} \mathrm{MOVE}(V_{all}, V_{\Gamma_I}) \cdot \psi_{all}^I \right) \cdot \delta V_{all} \\
&+ D_{V_{all}} \left(-\int_0^E \int_{\Omega_l} \mathrm{NSE}(u, p, V_{all}) \cdot \gamma \right) \cdot \delta V_{all} \\
=& \int_0^E \int_\Omega \omega \nabla T \cdot \delta V_{all} - \int_0^E \int_\Omega \Delta\psi_{all} \cdot \delta V_{all} \\
&- \int_0^E \int_{\partial\Omega} \psi_{all} \cdot \partial_{\boldsymbol{\mu}} \delta V_{all} + \int_0^E \int_{\Gamma_H \cup \Gamma_{out} \cup (\Gamma_N \setminus \Gamma_{\tilde{N}})} \partial_{\boldsymbol{\mu}} \psi_{all} \cdot \delta V_{all} \\
&- \int_0^E \int_{\Gamma_I} \psi_{all}^I \cdot \delta V_{all} + \int_0^E \int_{\Omega_l} (\nabla u)^T \cdot \gamma \cdot \delta V_{all} \\
=& \int_0^E \int_\Omega (-\Delta\psi_{all} + \omega \nabla T + (\nabla u)^T \cdot \gamma) \cdot \delta V_{all}
\end{aligned}$$

$$
\begin{aligned}
&-\int_0^E\int_{\partial\Omega}\psi_{all}\cdot\partial_{\boldsymbol{\mu}}\delta V_{all}-\int_0^E\int_{\Gamma_I}\psi_{all}^I\cdot\delta V_{all} \\
&+\int_0^E\int_{\Gamma_H\cup\Gamma_{out}\cup(\Gamma_N\setminus\Gamma_{\tilde{N}})}\partial_{\boldsymbol{\mu}}\psi_{all}\cdot\delta V_{all}.
\end{aligned}
\tag{3.20}
$$

The following adjoint equations emerge by proper variation with δV_{all} in (3.20)

$$
\begin{aligned}
-\Delta\psi_{all}+\omega\nabla T+(\nabla u)^T\cdot\gamma&=0, && \text{on } [0,E)\times\Omega,\\
\partial_{\boldsymbol{\mu}}\psi_{all}&=0, && \text{on } [0,E)\times(\Gamma_H\cup\Gamma_{out}\cup(\Gamma_N\setminus\Gamma_{\tilde{N}})),\\
\psi_{all}&=0, && \text{on } [0,E)\times\partial\Omega,\\
\psi_{all}^I&=0, && \text{on } [0,E)\times\Gamma_I.
\end{aligned}
\tag{3.21}
$$

The Derivative with Respect to the Velocity of the Fluid u

The variation of the tenth integral in the Lagrange functional (3.11) with respect to the velocity in the liquid phase u can be reformulated using integration by parts, the chain rule, and (3.14):

$$
\begin{aligned}
&D_u\Bigg(-\int_0^E\int_{\Omega_l}\mathrm{NSE}(u,p,V_{all})\cdot\gamma\Bigg)\delta u\\
&=-\int_0^E\int_{\Omega_l}(\partial_t\delta u+((u-V_{all})\cdot\nabla)\delta u+(\delta u\cdot\nabla)u-\eta\Delta\delta u)\cdot\gamma\\
&=-\int_0^E\int_{\Omega_l}\partial_t\delta u\cdot\gamma-\int_0^E\int_{\Omega_l}((u-V_{all})\cdot\nabla)\delta u\cdot\gamma-\int_0^E\int_{\Omega_l}(\delta u\cdot\nabla)u\cdot\gamma\\
&\quad+\int_0^E\int_{\Omega_l}\eta\Delta\delta u\cdot\gamma.
\end{aligned}
$$

For better readability, this equation is split into the first integral

$$
-\int_0^E\int_{\Omega_l}\partial_t\delta u\cdot\gamma=-\int_{\Omega_l}\gamma(E)\cdot\delta u(E)+\int_{\Omega_l}\gamma(0)\cdot\underbrace{\delta u(0)}_{\overset{(3.14)}{=}0}+\int_0^E\int_{\Omega_l}\partial_t\gamma\cdot\delta u,
$$

the second integral

$$
-\int_0^E\int_{\Omega_l}((u-V_{all})\cdot\nabla)\delta u\cdot\gamma
$$

$$
\begin{aligned}
&= -\int_0^E \int_{\partial\Omega_l} (\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) \cdot \delta u + \int_0^E \int_{\Omega_l} ((u - V_{all}) \cdot \nabla)\gamma \cdot \delta u \\
&= \int_0^E \int_{\Omega_l} ((u - V_{all}) \cdot \nabla)\gamma \cdot \delta u - \int_0^E \int_{\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)} (\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) \cdot \underbrace{\delta u(0)}_{\overset{(3.14)}{=} 0} \\
&\quad - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} (\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) \cdot \delta u,
\end{aligned}
$$

the third integral

$$
-\int_0^E \int_{\Omega_l} (\delta u \cdot \nabla) u \cdot \gamma = -\int_0^E \int_{\Omega_l} (\nabla u)^T \cdot \gamma \cdot \delta u,
$$

and the fourth integral

$$
\begin{aligned}
\int_0^E \int_{\Omega_l} \eta \Delta \delta u \cdot \gamma &= \int_0^E \int_{\partial\Omega_l} \eta\gamma \cdot \partial_{\boldsymbol{\mu}} \delta u - \int_0^E \int_{\Omega_l} \eta \nabla \cdot \gamma \cdot \nabla \cdot \delta u \\
&= \int_0^E \int_{\partial\Omega_l} \eta\gamma \cdot \partial_{\boldsymbol{\mu}} \delta u - \int_0^E \int_{\partial\Omega_l} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot \delta u + \int_0^E \int_{\Omega_l} \eta \Delta\gamma \cdot \delta u \\
&= \int_0^E \int_{\Omega_l} \eta \Delta\gamma \cdot \delta u + \int_0^E \int_{\partial\Omega_l} \eta\gamma \cdot \partial_{\boldsymbol{\mu}} \delta u \\
&\quad - \int_0^E \int_{\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot \underbrace{\delta u}_{\overset{(3.14)}{=} 0} - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot \delta u.
\end{aligned}
$$

The latter four equations combined yield

$$
\begin{aligned}
&D_u \left(-\int_0^E \int_{\Omega_l} \mathrm{NSE}(u, p, V_{all}) \cdot \gamma \right) \delta u \\
&= -\int_{\Omega_l} \gamma(E) \cdot \delta u(E) + \int_0^E \int_{\Omega_l} \partial_t \gamma \cdot \delta u + \int_0^E \int_{\Omega_l} ((u - V_{all}) \cdot \nabla)\gamma \cdot \delta u \\
&\quad - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} (\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) \cdot \delta u - \int_0^E \int_{\Omega_l} (\nabla u)^T \cdot \gamma \cdot \delta u
\end{aligned}
$$

$$
\begin{aligned}
&+\int_0^E\int_{\Omega_l}\eta\Delta\gamma\cdot\delta u+\int_0^E\int_{\partial\Omega_l}\eta\gamma\cdot\partial_{\boldsymbol{\mu}}\delta u-\int_0^E\int_{\Gamma_H\cup(\Gamma_{out}\cap\partial\Omega_l)}\eta\partial_{\boldsymbol{\mu}}\gamma\cdot\delta u\\
=&\int_0^E\int_{\Omega_l}\Big(\partial_t\gamma+((u-V_{all})\cdot\nabla)\gamma-(\nabla u)^T\cdot\gamma+\eta\Delta\gamma\Big)\cdot\delta u\\
&-\int_{\Omega_l}\gamma(E)\cdot\delta u(E)+\int_0^E\int_{\partial\Omega_l}\eta\gamma\cdot\partial_{\boldsymbol{\mu}}\delta u-\int_0^E\int_{\Gamma_H\cup(\Gamma_{out}\cap\partial\Omega_l)}\eta\partial_{\boldsymbol{\mu}}\gamma\cdot\delta u\\
&-\int_0^E\int_{\Gamma_H\cup(\Gamma_{out}\cap\partial\Omega_l)}(\gamma\cdot\boldsymbol{\mu})\cdot(u-V_{all})\cdot\delta u.
\end{aligned}
\tag{3.22}
$$

Analogously, the variation of the eleventh integral in (3.11) is

$$
\begin{aligned}
D_u\Big(-\int_0^E\int_{\Omega_l}\mathrm{DIV}(u)\cdot\pi\Big)\delta u=&-\int_0^E\int_{\Omega_l}\nabla\cdot\delta u\cdot\pi\\
=&-\int_0^E\int_{\partial\Omega_l}(\pi\cdot\boldsymbol{\mu})\cdot\delta u+\int_0^E\int_{\Omega_l}\nabla\pi\cdot\delta u\\
=&-\int_0^E\int_{\Gamma_I\cup(\Gamma_N\cap\partial\Omega_l)}(\pi\cdot\boldsymbol{\mu})\cdot\underbrace{\delta u}_{\overset{(3.14)}{=}0}-\int_0^E\int_{\Gamma_H\cup(\Gamma_{out}\cap\partial\Omega_l)}(\pi\cdot\boldsymbol{\mu})\cdot\delta u\\
&+\int_0^E\int_{\Omega_l}\nabla\pi\cdot\delta u.
\end{aligned}
\tag{3.23}
$$

Using (3.22) and (3.23), the variation of the Lagrange functional (3.11) with respect to u is

$$
\begin{aligned}
0=&D_u\mathcal{L}\delta u=D_u\Big(-\int_0^E\int_{\Omega_l}\mathrm{HEAT}(T,u,V_{all})\cdot\omega\Big)\delta u\\
&+D_u\Big(-\int_0^E\int_{\Omega_l}\mathrm{NSE}(u,p,V_{all})\cdot\gamma\Big)\delta u+D_u\Big(-\int_0^E\int_{\Omega_l}\mathrm{DIV}(u)\cdot\pi\Big)\delta u\\
&+D_u\Big(-\int_0^E\int_{\Gamma_H}\mathrm{CONT}(p,u,p_{in})\cdot\varphi\Big)\delta u\\
&+D_u\Big(-\int_0^E\int_{\Gamma_{out}\cap\partial\Omega_l}\mathrm{OUT}(p,u)\cdot\gamma^{out}\Big)\delta u
\end{aligned}
$$

$$
\begin{aligned}
&= -\int_0^E \int_{\Omega_l} \omega \nabla T \cdot \delta u \\
&\quad + \int_0^E \int_{\Omega_l} \Big(\partial_t \gamma + ((u - V_{all}) \cdot \nabla)\gamma - (\nabla u)^T \cdot \gamma + \eta \Delta \gamma \Big) \cdot \delta u \\
&\quad - \int_{\Omega_l} \gamma(E) \cdot \delta u(E) + \int_0^E \int_{\partial \Omega_l} \eta \gamma \cdot \partial_{\boldsymbol{\mu}} \delta u - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial \Omega_l)} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot \delta u \\
&\quad - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial \Omega_l)} (\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) \cdot \delta u \\
&\quad - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial \Omega_l)} (\pi \cdot \boldsymbol{\mu}) \cdot \delta u + \int_0^E \int_{\Omega_l} \nabla \pi \cdot \delta u \\
&\quad + \int_0^E \int_{\Gamma_H} \eta \varphi \cdot \partial_{\boldsymbol{\mu}} \delta u + \int_0^E \int_{\Gamma_{out} \cap \partial \Omega_l} \eta \gamma^{out} \cdot \partial_{\boldsymbol{\mu}} \delta u \\
&= \int_0^E \int_{\Omega_l} \Big(\partial_t \gamma + ((u - V_{all}) \cdot \nabla)\gamma - (\nabla u)^T \cdot \gamma \\
&\qquad\qquad + \eta \Delta \gamma + \nabla \pi - \omega \nabla T \Big) \cdot \delta u \\
&\quad - \int_0^E \int_{\Gamma_H \cup (\Gamma_{out} \cap \partial \Omega_l)} ((\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) + (\pi \cdot \boldsymbol{\mu}) + \eta \partial_{\boldsymbol{\mu}} \gamma) \cdot \delta u \\
&\quad + \int_0^E \int_{\Gamma_I \cup (\Gamma_N \cap \partial \Omega_l)} \eta \gamma \cdot \partial_{\boldsymbol{\mu}} \delta u + \int_0^E \int_{\Gamma_H} \eta (\varphi + \gamma) \cdot \partial_{\boldsymbol{\mu}} \delta u \\
&\quad + \int_0^E \int_{\Gamma_{out} \cap \partial \Omega_l} \eta (\gamma^{out} + \gamma) \cdot \partial_{\boldsymbol{\mu}} \delta u - \int_{\Omega_l} \gamma(E) \cdot \delta u(E).
\end{aligned}
$$

Proper variation with δu in the last equation leads to the following adjoint equations

$$
\begin{aligned}
\partial_t \gamma - (\nabla u)^T \cdot \gamma + ((u - V_{all}) \cdot \nabla)\gamma & \\
+ \eta \Delta \gamma + \nabla \pi &= \omega \nabla T, && \text{on } [0, E) \times \Omega_l, && (3.24\text{a}) \\
(\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) + \eta \partial_{\boldsymbol{\mu}} \gamma + \pi \cdot \boldsymbol{\mu} &= 0, && \text{on } [0, E) \times (\Gamma_H \cup (\Gamma_{out} \cap \partial \Omega_l)), && (3.24\text{b}) \\
\gamma &= 0, && \text{on } [0, E) \times (\Gamma_I \cup (\Gamma_N \cap \partial \Omega_l)), && (3.24\text{c}) \\
\varphi &= -\gamma, && \text{on } [0, E) \times \Gamma_H, && (3.24\text{d})
\end{aligned}
$$

$$\begin{aligned} \gamma^{out} &= -\gamma, && \text{on } [0,E) \times \Gamma_H, && (3.24e) \\ \gamma(E) &= 0, && \text{on } \Omega_l. && (3.24f) \end{aligned}$$

The latter system of equations can be interpreted as the system for the adjoint velocity γ and the adjoint pressure π. The first equation (3.24a) is similar to the momentum equation of the Navier–Stokes equations. The mass balance equation will arise in (3.26). The right hand side of (3.24a) couples with the adjoint state ω and through this indirectly with the distance terms in the cost functional (3.10). Equations (3.24b),(3.24c) and (3.24f) can be understood as boundary conditions and initial value condition at time $t = E$. The only occurrence of the adjoint state φ, which will appear in the gradient of the cost function, is in the equation $\varphi = -\gamma$. Details can be found in section 3.3.

The Derivative with Respect to the Pressure p

The variation of the tenth integral of the Lagrange functional (3.11) with respect to the pressure p can be reformulated by applying integration by parts

$$\begin{aligned} D_p\left(-\int_0^E \int_{\Omega_l} \mathrm{NSE}(u,p,V_{all}) \cdot \gamma \right) \delta p &= -\int_0^E \int_{\Omega_l} \nabla \delta p \cdot \gamma \\ = -\int_0^E \int_{\partial\Omega_l} (\gamma \cdot \boldsymbol{\mu}) \cdot \delta p &+ \int_0^E \int_{\Omega_l} \nabla \cdot \gamma \cdot \delta p. \end{aligned} \tag{3.25}$$

Taking the variation of the Lagrange functional (3.11) and using (3.25), leads to

$$\begin{aligned} 0 = & D_p \mathcal{L} \delta p = D_p\left(-\int_0^E \int_{\Omega_l} \mathrm{NSE}(u,p,V_{all}) \cdot \gamma \right) \delta p + D_p\left(-\int_0^E \int_{\Gamma_H} \mathrm{CONT}(p,p_{in}) \cdot \varphi \right) \delta p \\ & + D_p\left(-\int_0^E \int_{\Gamma_{out} \cap \partial\Omega_l} \mathrm{OUT}(p,u) \cdot \gamma^{out} \right) \delta p \\ = & -\int_0^E \int_{\partial\Omega_l} (\gamma \cdot \boldsymbol{\mu}) \cdot \delta p + \int_0^E \int_{\Omega_l} \nabla \cdot \gamma \cdot \delta p \\ & - \int_0^E \int_{\Gamma_H} \varphi \cdot \boldsymbol{\mu} \cdot \delta p - \int_0^E \int_{\Gamma_{out} \cap \partial\Omega_l} \gamma^{out} \cdot \boldsymbol{\mu} \cdot \delta p \\ = & \int_0^E \int_{\Omega_l} \nabla \cdot \gamma \cdot \delta p - \int_0^E \int_{\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)} (\gamma \cdot \boldsymbol{\mu}) \cdot \delta p \end{aligned}$$

$$- \int_0^E \int_{\Gamma_H} (\gamma \cdot \boldsymbol{\mu} + \varphi \cdot \boldsymbol{\mu}) \cdot \delta p - \int_0^E \int_{\Gamma_{out} \cap \partial\Omega_l} (\gamma \cdot \boldsymbol{\mu} + \gamma^{out} \cdot \boldsymbol{\mu}) \cdot \delta p.$$

From this equation the following adjoint equations arise

$$\nabla \cdot \gamma = 0, \qquad \text{on } [0,E) \times \Omega_l, \tag{3.26a}$$
$$\gamma \cdot \boldsymbol{\mu} = 0, \qquad \text{on } [0,E) \times (\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)), \tag{3.26b}$$
$$\varphi \cdot \boldsymbol{\mu} = -\gamma \cdot \boldsymbol{\mu}, \qquad \text{on } [0,E) \times \Gamma_H, \tag{3.26c}$$
$$\gamma^{out} \cdot \boldsymbol{\mu} = -\gamma \cdot \boldsymbol{\mu}, \qquad \text{on } [0,E) \times (\Gamma_{out} \cap \partial\Omega_l). \tag{3.26d}$$

The first equation (3.26a) fits perfectly to (3.24a) – (3.24f) to get an adjoint Navier–Stokes-like system for γ and π. Equations (3.26b) – (3.26d) appear also in (3.24a) – (3.24f) in a stronger version.

The Derivative with Respect to the Interface Graph h

We use the chain rule and integration by parts for the fifth integral together with (3.13). This leads to

$$\begin{aligned}
&D_h \left(- \int_0^E \int_{\Gamma_C} \mathrm{INT}(T,h) \cdot \psi \right) \cdot \delta h \\
&= - \int_0^E \int_{\Gamma_C} (1 + h_x^2) \cdot [k(\partial_y^2 T)]_l^s \circ \Phi \cdot \psi \cdot \delta h \\
&\quad - \int_0^E \int_{\Gamma_C} 2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot \psi \cdot \delta h_x + \int_0^E \int_{\Gamma_C} L \cdot \partial_t \delta h \cdot \psi.
\end{aligned}$$

For this equation, the application of integration by parts to the second integral

$$\begin{aligned}
&- \int_0^E \int_{\Gamma_C} 2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot \psi \cdot \delta h_x \\
&= - \int_0^E \int_{\partial\Gamma_C} 2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot (\psi \cdot \boldsymbol{\mu}) \cdot \delta h \\
&\quad + \int_0^E \int_{\Gamma_C} \partial_x (2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot \psi) \cdot \delta h,
\end{aligned}$$

and to the third one yields

$$\int_0^E\int_{\Gamma_C} L\cdot\partial_t\delta h\cdot\psi = \int_{\Gamma_C} L\cdot\psi(E)\cdot\delta h(E) - \int_{\Gamma_C} L\cdot\psi(0)\cdot\underbrace{\delta h(0)}_{\overset{(3.14)}{=}\,0} - \int_0^E\int_{\Gamma_C} L\cdot\partial_t\psi\cdot\delta h.$$

These equations consolidated again read

$$\begin{aligned}
&D_h\Bigg(-\int_0^E\int_{\Gamma_C}\mathrm{INT}(T,h)\cdot\psi\Bigg)\cdot\delta h\\
&=-\int_0^E\int_{\Gamma_C}(1+h_x^2)\cdot[k(\partial_y^2T)]_l^s\circ\Phi\cdot\psi\cdot\delta h\\
&\quad-\int_0^E\int_{\partial\Gamma_C}2h_x\cdot[k(\partial_yT)]_l^s\circ\Phi\cdot(\psi\cdot\boldsymbol{\mu})\cdot\delta h\\
&\quad+\int_0^E\int_{\Gamma_C}\partial_x(2h_x\cdot[k(\partial_yT)]_l^s\circ\Phi\cdot\psi)\cdot\delta h\\
&\quad+\int_{\Gamma_C}L\cdot\psi(E)\cdot\delta h(E)-\int_0^E\int_{\Gamma_C}L\cdot\partial_t\psi\cdot\delta h.
\end{aligned}\tag{3.27}$$

The same can be done for the sixth integral in (3.11)

$$\begin{aligned}
&D_h\Bigg(-\int_0^E\int_{\Gamma_C}\mathrm{VEL}(h,V_{\Gamma_I})\cdot\psi^C\Bigg)\cdot\delta h\\
&=-\int_0^E\int_{\Gamma_C}\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot\psi^C\cdot\delta h_x\\
&\quad-\int_0^E\int_{\Gamma_C}\sqrt{1+h_x^2}\cdot\partial_yV_{\Gamma_I}\circ\Phi\cdot\psi^C\cdot\delta h+\int_0^E\int_{\Gamma_C}\delta h_t\cdot\psi^C.
\end{aligned}$$

Again, treating the first

$$\begin{aligned}
&-\int_0^E\int_{\Gamma_C}\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot\psi^C\cdot\delta h_x\\
&=-\int_0^E\int_{\partial\Gamma_C}\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot(\psi^C\cdot\boldsymbol{\mu})\cdot\delta h
\end{aligned}$$

$$+\int_0^E\int_{\Gamma_C}\partial_x\left(\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot\psi^C\right)\cdot\delta h$$

and the third integral separately, we arrive at

$$\int_0^E\int_{\Gamma_C}\delta h_t\cdot\psi^C=\int_{\Gamma_C}\psi^C(E)\cdot\delta h(E)-\int_{\Gamma_C}\psi^C(0)\cdot\underbrace{\delta h(0)}_{\overset{(3.14)}{=}0}-\int_0^E\int_{\Gamma_C}\partial_t\psi^C\cdot\delta h,$$

which is combined as

$$\begin{aligned}
&D_h\left(-\int_0^E\int_{\Gamma_C}\mathrm{VEL}(h,V_{\Gamma_I})\cdot\psi^C\right)\cdot\delta h\\
=&-\int_0^E\int_{\partial\Gamma_C}\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot(\psi^C\cdot\boldsymbol{\mu})\cdot\delta h\\
&+\int_0^E\int_{\Gamma_C}\partial_x\left(\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot\psi^C\right)\cdot\delta h\\
&-\int_0^E\int_{\Gamma_C}\sqrt{1+h_x^2}\cdot\partial_yV_{\Gamma_I}\circ\Phi\cdot\psi^C\cdot\delta h\\
&+\int_{\Gamma_C}\psi^C(E)\cdot\delta h(E)-\int_0^E\int_{\Gamma_C}\partial_t\psi^C\cdot\delta h.
\end{aligned}\tag{3.28}$$

Together with (3.27) and (3.28), the variation of the Lagrange functional $\mathcal{L}$ with respect to h is

$$\begin{aligned}
0=&D_h\mathcal{L}\delta h=\Lambda\int_{\Gamma_C}(h(E)-h_d(E))\delta h(E)+\bar{\Lambda}\int_0^E\int_{\Gamma_C}(h-h_d)\delta h\\
&+D_h\left(-\int_0^E\int_{\Gamma_C}\mathrm{INT}(T,h)\cdot\psi\right)\cdot\delta h\\
&+D_h\left(-\int_0^E\int_{\Gamma_C}\mathrm{VEL}(h,V_{\Gamma_I})\cdot\psi^C\right)\cdot\delta h\\
=&\Lambda\int_{\Gamma_C}(h(E)-h_d(E))\delta h(E)+\bar{\Lambda}\int_0^E\int_{\Gamma_C}(h-h_d)\delta h
\end{aligned}$$

$$
\begin{aligned}
&-\int_0^E \int_{\Gamma_C} (1+h_x^2)\cdot [k(\partial_y^2 T)]_l^s \circ \Phi \cdot \psi \cdot \delta h \\
&-\int_0^E \int_{\partial\Gamma_C} 2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot (\psi \cdot \boldsymbol{\mu}) \cdot \delta h \\
&+\int_0^E \int_{\Gamma_C} \partial_x (2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot \psi) \cdot \delta h \\
&+\int_{\Gamma_C} L \cdot \psi(E) \cdot \delta h(E) - \int_0^E \int_{\Gamma_C} L \cdot \partial_t \psi \cdot \delta h \\
&-\int_0^E \int_{\partial\Gamma_C} \frac{h_x}{\sqrt{1+h_x^2}} \cdot V_{\Gamma_I} \circ \Phi \cdot (\psi^C \cdot \boldsymbol{\mu}) \cdot \delta h \\
&+\int_0^E \int_{\Gamma_C} \partial_x \left(\frac{h_x}{\sqrt{1+h_x^2}} \cdot V_{\Gamma_I} \circ \Phi \cdot \psi^C \right) \cdot \delta h \\
&-\int_0^E \int_{\Gamma_C} \sqrt{1+h_x^2} \cdot \partial_y V_{\Gamma_I} \circ \Phi \cdot \psi^C \cdot \delta h \\
&+\int_{\Gamma_C} \psi^C(E) \cdot \delta h(E) - \int_0^E \int_{\Gamma_C} \partial_t \psi^C \cdot \delta h \\
=&-\int_0^E \int_{\Gamma_C} \Bigg((1+h_x^2)\cdot [k(\partial_y^2 T)]_l^s \circ \Phi \cdot \psi - \partial_x (2h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot \psi) \\
&\qquad + L \cdot \partial_t \psi - \partial_x \left(\frac{h_x}{\sqrt{1+h_x^2}} \cdot V_{\Gamma_I} \circ \Phi \cdot \psi^C \right) \\
&\qquad + \sqrt{1+h_x^2} \cdot \partial_y V_{\Gamma_I} \circ \Phi \cdot \psi^C + \partial_t \psi^C - \bar{\Lambda}(h - h_d) \Bigg) \cdot \delta h \\
&-\int_0^E \int_{\partial\Gamma_C} h_x \left(2[k(\partial_y T)]_l^s \circ \Phi \cdot \psi + \frac{1}{\sqrt{1+h_x^2}} \cdot V_{\Gamma_I} \circ \Phi \cdot \psi^C \right) \cdot (\delta h \cdot \boldsymbol{\mu}) \\
&+\int_{\Gamma_C} \Big(\Lambda(h(E) - h_d(E)) + L \cdot \psi(E) + \psi^C(E) \Big) \delta h(E).
\end{aligned}
$$

By proper variation with δh, certain terms can be eliminated from the last equation and lead to the following adjoint equations

$$\begin{aligned}
&(1+h_x^2)\cdot[k(\partial_y^2 T)]_l^s\circ\Phi\cdot\psi\\
&-\partial_x(2h_x\cdot[k(\partial_y T)]_l^s\circ\Phi\cdot\psi)+L\cdot\partial_t\psi\\
&-\partial_x\left(\frac{h_x}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot\psi^C\right)\\
&+\sqrt{1+h_x^2}\cdot\partial_y V_{\Gamma_I}\circ\Phi\cdot\psi^C+\partial_t\psi^C-\bar{\Lambda}(h-h_d)=0, && \text{on } [0,E)\times\Gamma_C,\\
&2[k(\partial_y T)]_l^s\circ\Phi\cdot\psi+\frac{1}{\sqrt{1+h_x^2}}\cdot V_{\Gamma_I}\circ\Phi\cdot\psi^C=0, && \text{on } [0,E)\times\partial\Gamma_C,\\
&\Lambda(h(E)-h_d(E))+L\cdot\psi(E)+\psi^C(E)=0, && \text{on } \Gamma_C.
\end{aligned}\tag{3.29}$$

By substitution of the last equation in (3.21) into (3.17), it follows

$$\psi^C=0, \qquad \text{on } [0,E)\times\Gamma_C. \tag{3.30}$$

Substituting (3.30) into (3.29), we can simplify the system of adjoint equations for ψ to

$$\begin{aligned}
&L\cdot\partial_t\psi+(1+h_x^2)\cdot[k(\partial_y^2 T)]_l^s\circ\Phi\cdot\psi\\
&-\partial_x(2h_x\cdot[k(\partial_y T)]_l^s\circ\Phi\cdot\psi)-\bar{\Lambda}(h-h_d)=0, && \text{on } [0,E)\times\Gamma_C, && (3.31a)\\
&\psi=0, && \text{on } [0,E)\times\partial\Gamma_C, && (3.31b)\\
&\Lambda(h(E)-h_d(E))+L\cdot\psi(E)=0, && \text{on } \Gamma_C. && (3.31c)
\end{aligned}$$

The latter equations are a PDE system for the adjoint state ψ defined on the bottom boundary Γ_C. (3.31b) is the boundary condition and (3.31c) the initial value condition at terminal time E. Through the terms $\bar{\Lambda}(h-h_d)$ in (3.31a) and $\Lambda(h(E)-h_d(E))$ in (3.31c), ψ is directly coupled to the cost functional. Moreover, ψ is linked with the adjoint state ω and thereby with the adjoint states γ and particularly with φ, which has an important impact on the gradient of the cost functional (3.35).

3.2.3. Backward System

For the optimal control problem in this thesis, the adjoint equation in (3.2) is a PDE system consisting of the equations derived in the preceding six sections. Several Lagrange multipliers introduced in (3.11) have no influence on the adjoint state φ, and with that no influence on the gradient of the cost functional $\nabla\mathcal{K}$ (3.35). In particular, ω^I, ψ^C, ψ_{all}, ψ_{all}^I and γ^{out} are of no interest here because the adjoint equations are used only as a tool to formulate and compute $\nabla\mathcal{K}$. Hence, they are neglected in the following system.

For given states T, V_{all}, u and h, find functions ω, γ, π, ψ and φ such that

$$\begin{aligned}
\partial_t\omega + (u - V_{all})\cdot\nabla\omega + \alpha\Delta\omega &= 0, && \text{on } [0,E)\times\Omega,\\
\alpha\partial_{\boldsymbol{\mu}}\omega + u\cdot(\omega\cdot\boldsymbol{\mu}) &= 0, && \text{on } [0,E)\times(\Gamma_{out}\cap\partial\Omega_l),\\
\partial_{\boldsymbol{\mu}}\omega &= 0, && \text{on } [0,E)\times(\Gamma_N\cup(\Gamma_{out}\cap\partial\Omega_s)),\\
\omega\circ\Phi &= \sqrt{1+h_x^2}\cdot\psi, && \text{on } [0,E)\times\Gamma_C,\\
\omega &= 0, && \text{on } [0,E)\times(\Gamma_C\cup\Gamma_H),\\
\omega(E) &= 0, && \text{on } \Omega,\\
\partial_t\gamma + ((u - V_{all})\cdot\nabla)\gamma & && \\
-(\nabla u)^{\mathsf{T}}\cdot\gamma + \eta\Delta\gamma + \nabla\pi &= \omega\nabla T, && \text{on } [0,E)\times\Omega_l,\\
\nabla\cdot\gamma &= 0, && \text{on } [0,E)\times\Omega_l,\\
(\gamma\cdot\boldsymbol{\mu})\cdot(u - V_{all}) & && \\
+\eta\partial_{\boldsymbol{\mu}}\gamma + \pi\cdot\boldsymbol{\mu} &= 0, && \text{on } [0,E)\times(\Gamma_H\cup(\Gamma_{out}\cap\partial\Omega_l)),\\
\gamma &= 0, && \text{on } [0,E)\times(\Gamma_I\cup(\Gamma_N\cap\partial\Omega_l)),\\
\varphi &= -\gamma, && \text{on } [0,E)\times\Gamma_H,\\
\gamma(E) &= 0, && \text{on } \Omega_l,\\
L\cdot\partial_t\psi & && \\
+(1+h_x^2)\cdot[k(\partial_y^2 T)]_l^s\circ\Phi\cdot\psi & && \\
-\partial_x(2h_x\cdot[k(\partial_y T)]_l^s\circ\Phi\cdot\psi) &= \bar{\Lambda}(h - h_d), && \text{on } [0,E)\times\Gamma_C,\\
\psi &= 0, && \text{on } [0,E)\times\partial\Gamma_C,\\
\psi(E) &= -\frac{\Lambda}{L}(h(E) - h_d(E)), && \text{on } \Gamma_C.
\end{aligned} \tag{3.32}$$

Terminal values for the adjoint states ω, γ and ψ are given for the time E. Thus, in contrast to the forward system (3.9), the equations in (3.32) have to be solved backwards in time. Throughout this thesis, the latter system (3.32) is named the backward system.

3.3. Gradient Method

The optimal control problem can be solved with a gradient method. For this, not only the forward and backward systems are required, but also the gradient of the cost functional $\nabla\mathcal{K}$, which we derive in this section.
Following the derivation of (3.8), the Lagrange functional can be used to formulate

$$\begin{aligned}
D_{p_{in}}\mathcal{K}\delta p_{in} &= D_{p_{in}}\mathcal{L}\delta p_{in}\\
&= D_{p_{in}}\left(\frac{\lambda}{2}\int_0^E\int_{\Gamma_H}(p_{in})^2\right)\delta p_{in} - D_{p_{in}}\left(\int_0^E\int_{\Gamma_H}\mathrm{CONT}(p,u,p_{in})\cdot\varphi\right)\delta p_{in}\\
&= \int_0^E\int_{\Gamma_H}\lambda p_{in}\delta p_{in} + \int_0^E\int_{\Gamma_H}(\boldsymbol{\mu}\cdot\varphi)\delta p_{in}
\end{aligned}$$

$$= \int_0^E \int_{\Gamma_H} (\lambda p_{in} + (\boldsymbol{\mu} \cdot \varphi))\delta p_{in}.$$

The analogue to the gradient condition (3.6) reads

$$\int_0^E \int_{\Gamma_H} (\lambda p_{in} + (\boldsymbol{\mu} \cdot \varphi))(v - p_{in}) \geq 0, \qquad \text{for all } v \in \mathcal{C}_{ad} \tag{3.33}$$

where $\mathcal{C}_{ad} = \{c \in \mathcal{C} : \underline{c} \leq c(t) \leq \overline{c}, t \in [0, E]\}$. The unrestricted case $\mathcal{C}_{ad} = \mathcal{C}$ can be expressed with $\underline{c} = -\infty$, $\overline{c} = \infty$ which results in the following gradient equation instead of the latter inequality (3.33)

$$0 = \lambda p_{in} + \frac{1}{|\Gamma_H|} \int_{\Gamma_H} \boldsymbol{\mu} \cdot \varphi, \qquad t \in (0, E]. \tag{3.34}$$

As a consequence, the required gradient of the cost functional (3.10) can be expressed as

$$\nabla \mathcal{K} = \lambda p_{in} + \frac{1}{|\Gamma_H|} \int_{\Gamma_H} \boldsymbol{\mu} \cdot \varphi, \tag{3.35}$$

and is now available to be plugged into a gradient method.

3.3.1. Projected Gradient Method

The cost functional (3.10) is to be minimized

$$\min_{c \in \mathcal{C}_{ad}} \mathcal{K}(c).$$

Given a control $p_{in}^{k-1} \in \mathcal{C}_{ad}$, the direction of steepest descent is the negative gradient $-\nabla \mathcal{K}(p_{in}^{k-1})$. The projected gradient method [34], described in algorithm 1, uses this as the descent direction (step 6). Further, a step size s^k is computed in step 5 with algorithm 2. To ensure that the computed control is admissible, the projection $\mathbf{P}_{[\underline{c},\, \overline{c}]} : \mathcal{C} \to \mathcal{C}_{ad}$ is utilized pointwise in time (step 7).

$$\mathbf{P}_{[\underline{c},\, \overline{c}]}(c) := \max\{\underline{c}, \min\{c, \overline{c}\}\}.$$

A discussion of this technique can be found in section 3.3.3.

Steps 3 and 4 in algorithm 1 require the numerical solution of several PDE systems. This is done in a discretized setting. More details on this can be found in chapter 4. In addition, an *initial guess* $p_{in}^0 \in \mathcal{C}_{ad}$ is required as input to the algorithm. This issue is addressed in chapter 5.

Algorithm 1: Projected Gradient Method

```
   Input: initial control $p_{in}^0$
   Output: control $p_{in}^{k_{end}}$
1  $k = 1$
2  while not converged do
3      solve forward problem (3.9)
4      solve backward problem (3.32)
5      compute step size $s^k$
6      $d^k = \lambda p_{in}^{k-1} + \frac{1}{|\Gamma_H|} \int_{\Gamma_H} \boldsymbol{\mu} \cdot \varphi$
7      $p_{in}^k = \mathbf{P}_{[\underline{c},\, \overline{c}]}(p_{in}^{k-1} - s^k \cdot d^k)$
8      $k = k + 1$
9  end
```

Sufficient stopping criteria, evaluated in step 2 of algorithm 1, are crucial for the convergence of the projected gradient algorithm. Referring to the gradient equation (3.34), the algorithm should stop if the norm of the descent step is smaller than a certain tolerance δ_1

$$||d^k|| < \delta_1. \tag{3.36}$$

This quantity strongly depends on the parameters chosen in (3.10). Thus, it is possible that descent directions with very small norm lead to relative large advance in convergence. In this case, the step size s^k is chosen respectively large. So a more suitable criterion is

$$||s^k \cdot d^k|| < \delta_1. \tag{3.37}$$

In the restricted case, where the gradient equation (3.34) has to be replaced by the inequality (3.33), condition (3.36) may be unfulfilled even for a locally optimal control c^*. However, the corresponding step size would be zero since d^k is no descent direction anymore and (3.37) would be fulfilled.

Another possibility for a stopping criterion is to observe the relative change of the cost functional.

$$\frac{|\mathcal{K}(p_{in}^{k-1}) - \mathcal{K}(p_{in}^{k})|}{|\mathcal{K}(p_{in}^{k-1})|} < \delta_2, \tag{3.38}$$

where δ_2 is another tolerance.

It is well known, that gradient methods show a slow convergence behavior [34]. In our experiments, the algorithm is stopped after a predefined number of iteration steps

$$k > k_{max}. \tag{3.39}$$

Besides the stopping criteria, the choice of the step size s^k is of great significance for the performance of the projected gradient method.

Algorithm 2: Quadratic Line Minimization

```
   Input: The step direction $d^k$
   Output: step size $s$
1  $i = 1$
2  choose $s_0 = 0 < s_1 < s_2,\ \epsilon_1$
3  $k_j = \mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k-1} - s_j \cdot d^k)), \quad j = 0, 1, 2$
4  while not converged do
5      $q \in \mathbb{P}_2 : q(s_j) = k_j, \quad j = 0, 1, 2$
6      $s = \operatorname*{argmin}_{\tilde{s} \in [s_0, s_2]} q(\tilde{s})$
7      if $|s - s_2| < \epsilon_1$ then
8          $s_0 = s_1,\ k_0 = k_1,\ s_1 = s_2,\ k_1 = k_2$
9          $s_2 = s_2 + s_1 - s_0$
10         $k_2 = \mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k-1} - s_2 \cdot d^k))$
11     else if $s > s_1$ then
12         $s_0 = s_1,\ k_0 = k_1$
13         $s_1 = s$
14         $k_1 = \mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k-1} - s_1 \cdot d^k))$
15         $i = i + 1$
16     else
17         $s_2 = s_1,\ k_2 = k_1$
18         $s_1 = s$
19         $k_1 = \mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k-1} - s_1 \cdot d^k))$
20         $i = i + 1$
21     end
22 end
```

3.3.2. Line Minimization Algorithm

The algorithm to compute the step size is a modification of the method used in [40]. Three sampling points are evaluated to approximate $\mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k-1} - s \cdot d^k))$ with a quadratic polynomial q. The local minimum of q is used as the next sampling point to refine the approximation.

Step 7 in algorithm 2 is a special case. The minimal value of $q(s), s \in [s_0, s_2]$ is very close or equal to s_2 and the actual minimum is expected to be outside of the interval $[s_0, s_2]$. Thus, the range is extended. This is repeated until the minimum of q is inside the interval $[s_0, s_2]$ without increasing the iteration counter i. The right boundary is raised by $s_2 - s_1$ every time and grows linearly. In case $||d^k||$ is small, step 9 of algorithm 2 can be replaced with $s_2 = 2 \cdot s_2$. This leads to an exponential growth rate of s_2 and can save some iterations of algorithm 2.

In every iteration of the algorithm, there appears at least one evaluation of the cost functional $\mathcal{K}(\mathbf{P}_{[\underline{c},\ \overline{c}]}(p_{in}^{k-1} - s_j \cdot d^k))$. These evaluations require the solution of the forward problem (3.9) and are computationally expensive.

To limit the computational cost of the *Quadratic Line Minimization*, a stopping criterion in step 4 of algorithm 2 is determined by a maximum iteration number i_{max}. If $i > i_{max}$ the sampling point s_j with the smallest cost value $k_j, j = 0, 1, 2$ is returned. Further, with tolerances ϵ_2, ϵ_3, the algorithm stops if the newly computed minimum s of the polynomial q is close to an already existing sampling point

$$|s - s_j| < \epsilon_2, \qquad \text{for any } j = 0, 1, 2,$$

or if the relative change of the value of $\mathcal{K}$ at the new sampling point s is small

$$\frac{|\mathcal{K}(\mathbf{P}_{[\underline{c},\, \overline{c}]}(p_{in}^{k-1} - s \cdot d^k)) - k_j|}{|k_1|} < \epsilon_3, \qquad \text{for any } j = 0, 1, 2.$$

If the minimum of q computed in the first iteration step is zero or almost zero, it is assumed that the sampling points may have been too large and s_1 and s_2 are chosen $s_1 = \delta$, $s_2 = 2 \cdot \delta$ with $0 < \delta \ll 1$ small. If the minimum computed with these small sampling points is still close to zero, $s = 0$ is returned.

3.3.3. Treatment of Control Constraints

The treatment of control constraints, as described in section 3.3 and used in algorithm 1, is not trivial. Tröltzsch formulates a projected gradient algorithm in [34], which is similar to algorithm 1. He also names the difficulties to find a step size when control constraints are present. As Ziegenbalg [40] and Bernauer [8], we follow the approach in [34] to simply truncate the control point wise in time with the projection $\mathbf{P}_{[\underline{c},\, \overline{c}]}$ if it violates the constraints.

A different approach, which is suggested in [34] especially for non-linear problems, is to use second-order methods. A candidate would be the sequential quadratic programming (SQP) method. It is a Newton-type method. In contrast to Newton's method, the handling of state constraints is straight forward in the SQP-method [34]. However, second-order derivatives and second-order optimality conditions are required for this algorithm like in [2]. Hinze analyzes the SQP-method in [16] and compares it to first-order methods like the gradient algorithm. An analysis of the SQP-method for the control of a phase field equation can be found in [15].

Another way to include control constraints of box-type into the optimization of a Stefan problem in enthalpy formulation are penalty methods. Roubíček studies this in [28] and together with Verdi in [29].

Neittaanmäki describes a Stefan problem in enthalpy formulation in [24] and treats box-type control constraints with a regularization approach which is introduced in [33].

The next chapter describes how the PDE systems are discretized in space over time with a mesh of triangles and finite elements. It also mentions the numerical implementation in FEniCS.

4. Implementation and Discretization

Step 3 in algorithm 1 requires the forward system to be solved numerically. Since this can not be done in a continuous setting, a discretization is needed. The same holds for the backward system in step 4. In this chapter, the spatial and temporal discretization are described. The domain Ω is partitioned into a mesh of triangles, as pictured in section 4.1. The PDE systems (3.9) and (3.32) are discretized with finite elements and an implicit Euler scheme.The weak formulations are developed in section 4.2. For the numerical implementation, the software FEniCS 1.5.0 [1] is used in Python 2.7.6 [31] together with the Python package SciPy 0.15.1 [19]. The code is freely accessible at

`https://gitlab.mpi-magdeburg.mpg.de/baran/Stefan_Problem_in_FEniCS.git`.

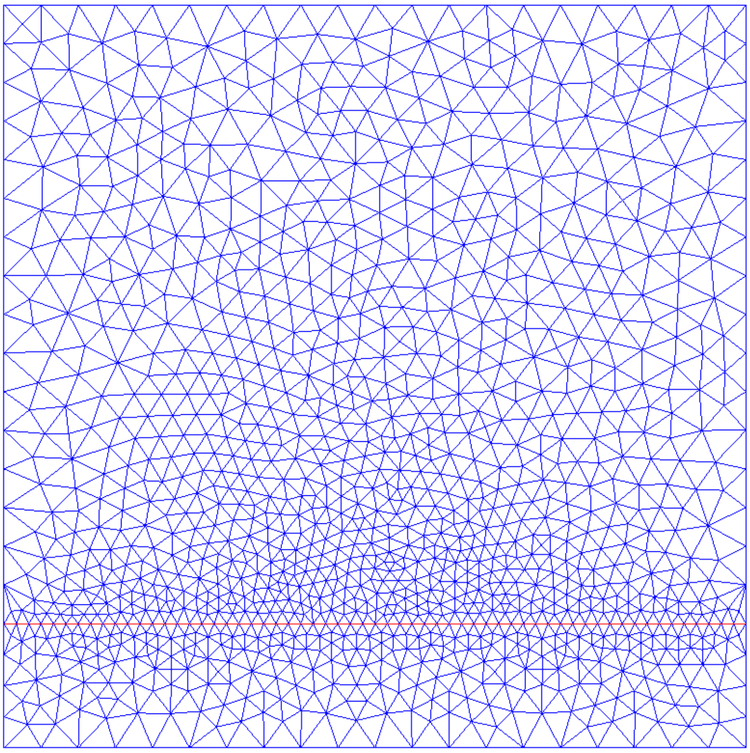


Figure 4.1.: Triangulation of the domain $\Omega(t_0)$ respecting the interface position (—).

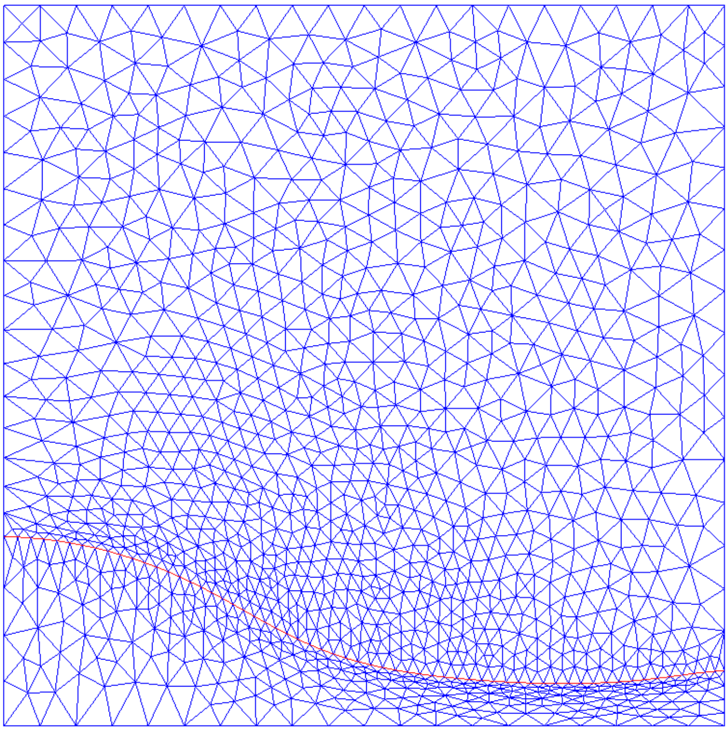


Figure 4.2.: Triangulation of the domain $\Omega(t_N)$ respecting the moved interface position.

4.1. Spatial Discretization

The domain $\Omega \subset \mathbb{R}^2$ is partitioned with a mesh of triangles. One example for $t = 0$ can be seen in figure 4.1. The interface Γ_I is respected by the triangulation. It is represented explicitly by edges of the mesh (—). These edges move in direction $V_{\Gamma_I} \cdot \boldsymbol{\nu}$ together with Γ_I as illustrated in figure 4.2. In order to prevent the triangulation from extreme deformation, $V_{\Gamma_I} \cdot \boldsymbol{\nu}$ is extended smoothly to V_{all} over the whole domain and the whole mesh is moved with V_{all}. Thus, the domain is discretized with a varying mesh for each time step.

4.2. Weak Formulations

In this section the weak formulations of the equations in (3.9) and (3.32) are presented. Let $0 = t_0 < \ldots < t_N = E$ be a partition of the time interval $[0, E]$. For $1 \leq n \leq N$, let $\Sigma = \Sigma^n$ be the triangulation of the domain $\Omega(t_n)$ at time t_n. It can be split into the triangulation of the solid and liquid phases respectively $\Sigma = \Sigma_s \dot{\cup} \Sigma_l$. The spatial discretizations of the boundaries Γ_C and Γ_I are titled Σ_C and Σ_I. Since the boundaries are one-dimensional, the discretizations are unions of intervals.

Heat Equation

The temperature at time t_n is approximated in a piecewise linear finite element space

$$T^n \in \mathcal{Q} := \{v \in C^0(\Omega) : v|_S \in \mathbb{P}_1, S \in \Sigma\}.$$

To obtain the weak formulation of the heat equation (2.4a), it is multiplied with a test function $v \in \mathcal{Q}$ and integrated over the domain Ω. Using integration by parts it can be reformulated.

$$\begin{aligned}
0 &= \int\limits_{\Omega} \mathrm{HEAT}(T, u, V_{all}) \cdot v = \int\limits_{\Omega_s} \mathrm{HEAT}(T, u, V_{all}) \cdot v + \int\limits_{\Omega_l} \mathrm{HEAT}(T, u, V_{all}) \cdot v \\
&= \int\limits_{\Omega_s} (\partial_t T + (u - V_{all}) \cdot \nabla T - k_s \Delta T) \cdot v + \int\limits_{\Omega_l} (\partial_t T + (u - V_{all}) \cdot \nabla T - k_l \Delta T) \cdot v \\
&= \int\limits_{\Omega_s} \partial_t T \cdot v + \int\limits_{\Omega_l} \partial_t T \cdot v + \int\limits_{\Omega_s} (u - V_{all}) \cdot \nabla T \cdot v + \int\limits_{\Omega_l} (u - V_{all}) \cdot \nabla T \cdot v \qquad (4.1) \\
&\quad - \int\limits_{\partial\Omega_s} k_s \partial_{\boldsymbol{\mu}} T \cdot v + \int\limits_{\Omega_s} k_s \nabla T \cdot \nabla v - \int\limits_{\partial\Omega_l} k_l \partial_{\boldsymbol{\mu}} T \cdot v + \int\limits_{\Omega_l} k_l \nabla T \cdot \nabla \\
&= \int\limits_{\Omega} \partial_t T \cdot v + \int\limits_{\Omega} (u - V_{all}) \cdot \nabla T \cdot v + \int\limits_{\Omega} \alpha \nabla T \cdot \nabla v - \int\limits_{\partial\Omega_s} k_s \partial_{\boldsymbol{\mu}} T \cdot v - \int\limits_{\partial\Omega_l} k_l \partial_{\boldsymbol{\mu}} T \cdot v.
\end{aligned}$$

The integrals over the boundaries are treated separately in what follows. The test function v is equal to zero on parts of the domain, where the temperature T is known due to Dirichlet boundary conditions. In this case, this holds on Γ_I since $T = T_m$ is known and analogously on Γ_H and Γ_C:

$$\begin{aligned}
&- \int\limits_{\partial\Omega_s} k_s \partial_{\boldsymbol{\mu}} T \cdot v - \int\limits_{\partial\Omega_l} k_l \partial_{\boldsymbol{\mu}} T \cdot v \\
= &- \int\limits_{(\Gamma_N \cup \Gamma_{out}) \cap \partial\Omega_s} k_s \underbrace{\partial_{\boldsymbol{\mu}} T}_{\overset{(2.4g)}{=} 0} \cdot v - \int\limits_{(\Gamma_N \cup \Gamma_{out}) \cap \partial\Omega_l} k_l \underbrace{\partial_{\boldsymbol{\mu}} T}_{\overset{(2.4g)}{=} 0} \cdot v \\
&- \int\limits_{\Gamma_I} k_s (\partial_{\boldsymbol{\mu}} T)_s \cdot \underbrace{v}_{\overset{(2.4f)}{=} 0} - \int\limits_{\Gamma_I} k_l (\partial_{\boldsymbol{\mu}} T)_l \cdot \underbrace{v}_{\overset{(2.4f)}{=} 0} \\
&- \int\limits_{\Gamma_H} k_l (\partial_{\boldsymbol{\mu}} T)_l \cdot \underbrace{v}_{\overset{(2.4d)}{=} 0} - \int\limits_{\Gamma_C} k_s (\partial_{\boldsymbol{\mu}} T)_s \cdot \underbrace{v}_{\overset{(2.4e)}{=} 0} \\
= &\ 0.
\end{aligned}$$

To discretize in time, the derivative of T with respect to time is replaced by a finite difference:

$$\int\limits_{\Omega} \partial_t T \cdot v = \int\limits_{\Omega} \frac{T^n - T^{n-1}}{\tau} \cdot v = \int\limits_{\Omega} \frac{1}{\tau} T^n \cdot v - \int\limits_{\Omega} \frac{1}{\tau} T^{n-1} \cdot v,$$

where $\tau = t_n - t_{n-1}$ is the time step size.
The equation (4.1) can be expressed in terms of a bilinear form

$$a_T(T^n, v) := \int\limits_{\Omega} \frac{1}{\tau} T^n \cdot v + \int\limits_{\Omega} (u^n - V^n_{all}) \cdot \nabla T^n \cdot v + \int\limits_{\Omega} \alpha \nabla T^n \cdot \nabla v,$$

and a linear form

$$L_T(v) := \int\limits_{\Omega} \frac{1}{\tau} T^{n-1} \cdot v,$$

as

$$a_T(T, v) = L_T(v), \qquad \text{for all } t \in [t_1, \dots, t_N], \text{and for all } v \in \mathcal{Q}.$$

Interface Movement

To compute the velocity of the interface V_{Γ_I} in normal direction at time t_n, the Stefan condition (2.2) is solved in a variational form and V_{Γ_I} is approximated in a piecewise linear finite element space

$$V^n_{\Gamma_I} \in \mathcal{V}_{\Gamma_I} := \{v \in C^0(\Gamma_I) : v|_{\hat{S}} \in \mathbb{P}_1, \hat{S} \in \Sigma_I\}.$$

To accomplish this, again a weak formulation is used:

$$\int\limits_{\Gamma_I} V^n_{\Gamma_I} \cdot v = \int\limits_{\Gamma_I} \frac{\tau}{L} [k(\nabla T^{n-1})]^s_l \cdot \boldsymbol{\nu} \cdot v, \quad \text{for all } n \in [1, \dots, N], \text{and for all } v \in \mathcal{V}_{\Gamma_I}.$$

Since the position of the interface is known and matched by facets and vertices of the mesh, these facets and vertices can be moved along $V^n_{\Gamma_I} \cdot \boldsymbol{\nu}$. With the positions of the interface in the actual and the previous time step, the graph h and its derivatives can be determined:

$$h^n \in \mathcal{H} := \{v \in C^0(\Gamma_C) : v|_{\hat{S}} \in \mathbb{P}_2, \hat{S} \in \Sigma_C\},$$
$$h^n_x \in \hat{\mathcal{H}} := \{v \in C^0(\Gamma_C) : v|_{\hat{S}} \in \mathbb{P}_1, \hat{S} \in \Sigma_C\}.$$

Mesh Movement

In order to expand the movement of the interface along the interface normals in a continuous and smooth way, the Laplace equation (2.5) is solved for

$$V^n_{all} \in \mathcal{V} := \{v \in C^0(\Omega)^2 : v|_S \in \mathbb{P}^2_1, S \in \Sigma\}.$$

The weak formulation of this reads as follows:

$$\begin{aligned}
0 =\; & \int_{\Omega} \mathrm{MESH}(V_{all}^{n}) \cdot v \\
=\; & \int_{\Omega} \Delta V_{all}^{n} \cdot v = \int_{\partial\Omega} \partial_{\boldsymbol{\mu}} V_{all}^{n} \cdot v - \int_{\Omega} \nabla V_{all}^{n} \cdot \nabla v = - \int_{\Omega} \nabla V_{all}^{n} \cdot \nabla v \\
& + \int_{\Gamma_C \cup \Gamma \tilde{N}} \partial_{\boldsymbol{\mu}} V_{all}^{n} \cdot \underbrace{v}_{\overset{(2.5)}{=}\, 0} + \int_{\Gamma_H \cup \Gamma_{out} \cup (\Gamma_N \setminus \Gamma_{\tilde{N}})} \partial_{\boldsymbol{\mu}} V_{all}^{n} \cdot v \\
=\; & - \int_{\Omega} \nabla V_{all}^{n} \cdot \nabla v \\
& + \int_{\Gamma_H \cup \Gamma_{out} \cup (\Gamma_N \setminus \Gamma_{\tilde{N}})} \partial_{\boldsymbol{\mu}} V_{all}^{n} \cdot v, \quad \text{for all } t \in [t_1, \dots, t_N], \text{and for all } v \in \mathcal{V}.
\end{aligned}$$

Navier–Stokes Equations

The approximation of the velocity and pressure in the liquid phase can be computed with $\mathbb{P}_2/\mathbb{P}_1$ Taylor–Hood finite elements [6]:

$$\begin{aligned}
u^{n} &\in \mathcal{U} := \{ v \in C^{0}(\Omega_l)^{2} : v|_{S} \in \mathbb{P}_2^{2}, S \in \Sigma_l \}, \\
p^{n} &\in \mathcal{P} := \{ v \in C^{0}(\Omega_l) : v|_{S} \in \mathbb{P}_1, S \in \Sigma_l \}.
\end{aligned}$$

The weak formulation of the Navier–Stokes equations is obtained by multiplying with test functions $v \in \mathcal{U}$ and $q \in \mathcal{P}$ and integrating over the domain Ω_l:

$$\begin{aligned}
0 &= \int_{\Omega_l} \mathrm{NSE}(u, p, V_{all}) \cdot v + \int_{\Omega_l} \mathrm{DIV}(u) \cdot q \\
&= \int_{\Omega_l} (\partial_t u + ((u - V_{all}) \cdot \nabla) u - \eta \Delta u + \nabla p) \cdot v + \int_{\Omega_l} \nabla \cdot u \cdot q \\
&= \int_{\Omega_l} \partial_t u \cdot v + \int_{\Omega_l} ((u - V_{all}) \cdot \nabla) u \cdot v - \int_{\partial\Omega_l} \eta \partial_{\boldsymbol{\mu}} u \cdot v \\
&\quad + \int_{\Omega_l} \eta \nabla u \cdot \nabla v + \int_{\partial\Omega_l} p \cdot \boldsymbol{\mu} \cdot v - \int_{\Omega_l} p \cdot \nabla \cdot v + \int_{\Omega_l} \nabla \cdot u \cdot q \\
&= \int_{\Omega_l} \partial_t u \cdot v + \int_{\Omega_l} ((u - V_{all}) \cdot \nabla) u \cdot v + \int_{\Omega_l} \eta \nabla u \cdot \nabla v - \int_{\Omega_l} p \cdot \nabla \cdot v \\
&\quad + \int_{\Omega_l} \nabla \cdot u \cdot q + \int_{\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)} (p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u) \cdot \underbrace{v}_{\overset{(2.6a)}{=}\, 0} \\
&\quad + \int_{\Gamma_H} \underbrace{(p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u)}_{\overset{(2.6a)}{=}\, p_{in} \cdot \boldsymbol{\mu}} \cdot v + \int_{\Gamma_{out} \cap \partial\Omega_l} \underbrace{(p \cdot \boldsymbol{\mu} - \eta \partial_{\boldsymbol{\mu}} u)}_{\overset{(2.6a)}{=}\, 0} \cdot v
\end{aligned}$$

$$= \int\limits_{\Omega_l} \partial_t u \cdot v + \int\limits_{\Omega_l} ((u - V_{all}) \cdot \nabla) u \cdot v + \int\limits_{\Omega_l} \eta \nabla u \cdot \nabla v \tag{4.2}$$
$$- \int\limits_{\Omega_l} p \cdot \nabla \cdot v + \int\limits_{\Omega_l} \nabla \cdot u \cdot q + \int\limits_{\Gamma_H} p_{in} \cdot \boldsymbol{\mu} \cdot v.$$

The derivative of u with respect to time in (4.2) is discretized in time as follows:

$$\int\limits_{\Omega_l} \partial_t u \cdot v = \int\limits_{\Omega_l} \frac{u^n - u^{n-1}}{\tau} \cdot v = \int\limits_{\Omega_l} \frac{1}{\tau} u^n \cdot v - \int\limits_{\Omega_l} \frac{1}{\tau} u^{n-1} \cdot v.$$

The equation (4.2) can now be expressed in terms of a multilinear form

$$a_u(u^n, p^n, v, q) := \int\limits_{\Omega_l} \frac{1}{\tau} u^n \cdot v + \int\limits_{\Omega_l} ((u^n - V_{all}^n) \cdot \nabla) u^n \cdot v + \int\limits_{\Omega_l} \eta \nabla u^n \cdot \nabla v$$
$$- \int\limits_{\Omega_l} p^n \cdot \nabla \cdot v + \int\limits_{\Omega_l} \nabla \cdot u^n \cdot q,$$

and a bilinear form

$$L_u(v, q) := \int\limits_{\Omega_l} \frac{1}{\tau} u^{n-1} \cdot v - \int\limits_{\Gamma_H} p_{in}^n \cdot \boldsymbol{\mu} \cdot v,$$

as

$$a_u(u, p, v, q) = L_u(v, q), \qquad \text{for all } t \in [t_1, \dots, t_N], \text{and for all } v \in \mathcal{U}, q \in \mathcal{P}.$$

Adjoint Interface

The adjoint equations in (3.32) are solved backwards in time. Assuming (3.9) is already solved, the solution of (3.32) for $t_N = E$ is known and serves as initial value. Instead, the solution for $t_0 = 0$ is among the unknowns. The adjoint state ψ at time t_{n-1} is approximated in a piecewise quadratic finite element space

$$\psi^{n-1} \in \mathcal{D} := \{v \in C^0(\Gamma_C) : v|_{\hat{S}} \in \mathbb{P}_2, \hat{S} \in \Sigma_C^{n-1}\}.$$

The adjoint equation (3.31a) is multiplied with a test function $v \in \mathcal{D}$ and integrated over Γ_C

$$0 = \int\limits_{\Gamma_C} L \cdot \partial_t \psi \cdot v + \int\limits_{\Gamma_C} (1 + h_x^2) \cdot [k(\partial_y^2 T)]_l^s \circ \Phi \cdot \psi \cdot v$$
$$- \int\limits_{\Gamma_C} \partial_x (2 h_x \cdot [k(\partial_y T)]_l^s \circ \Phi \cdot \psi) \cdot v - \int\limits_{\Gamma_C} \bar{\Lambda}(h - h_d) \cdot v.$$

The discretization in time is realized by a finite difference

$$\int\limits_{\Gamma_C} L \cdot \partial_t \psi \cdot v = \int\limits_{\Gamma_C} L \cdot \frac{\psi^n - \psi^{n-1}}{\tau} \cdot v = \int\limits_{\Gamma_C} \frac{L}{\tau} \psi^n \cdot v - \int\limits_{\Gamma_C} \frac{L}{\tau} \psi^{n-1} \cdot v.$$

With this, the bilinear and linear forms for the adjoint state ψ read

$$\begin{aligned} a_\psi(\psi^{n-1}, v) :=& \int\limits_{\Gamma_C} \frac{L}{\tau} \psi^{n-1} \cdot v - \int\limits_{\Gamma_C} (1 + (h_x^{n-1})^2) \cdot [k(\partial_y^2 T^{n-1})]_l^s \circ \Phi \cdot \psi^{n-1} \cdot v \\ &+ \int\limits_{\Gamma_C} \partial_x (2 h_x^{n-1} \cdot [k(\partial_y T^{n-1})]_l^s \circ \Phi \cdot \psi^{n-1}) \cdot v, \\ L_\psi(v) :=& \int\limits_{\Gamma_C} \frac{L}{\tau} \psi^n \cdot v - \int\limits_{\Gamma_C} \bar{\Lambda} (h^{n-1} - h_d^{n-1}) \cdot v, \end{aligned}$$

and the weak formulation for ψ can be formulated as

$$a_\psi(\psi, v) = L_\psi(v), \qquad \text{for all } t \in [t_0, \dots, t_{N-1}], \text{and for all } v \in \mathcal{D}.$$

Adjoint Temperature

To compute the adjoint temperature ω at time t_{n-1}, it is approximated in a piecewise quadratic finite element space

$$\omega^{n-1} \in \mathcal{O} := \{v \in C^0(\Omega) : v|_S \in \mathbb{P}_2, S \in \Sigma^{n-1}\}.$$

By multiplying with a test function $v \in \mathcal{O}$, integrating equation (3.16a) over Ω and applying integration by parts, the weak formulation is obtained

$$\begin{aligned} 0 =& \int\limits_{\Omega} (\partial_t \omega + (u - V_{all}) \cdot \nabla \omega + \alpha \Delta \omega) \\ =& \int\limits_{\Omega} \partial_t \omega \cdot v + \int\limits_{\Omega} (u - V_{all}) \cdot \nabla \omega \cdot v + \int\limits_{\Omega_s} k_s \Delta \omega \cdot v + \int\limits_{\Omega_l} k_l \Delta \omega \cdot v \\ =& \int\limits_{\Omega} \partial_t \omega \cdot v + \int\limits_{\Omega} (u - V_{all}) \cdot \nabla \omega \cdot v + \int\limits_{\partial \Omega_s} k_s \partial_{\boldsymbol{\mu}} \omega \cdot v - \int\limits_{\Omega_s} k_s \nabla \omega \cdot \nabla v \\ &+ \int\limits_{\partial \Omega_l} k_l \partial_{\boldsymbol{\mu}} \omega \cdot v - \int\limits_{\Omega_l} k_l \nabla \omega \cdot \nabla v \\ =& \int\limits_{\Omega} \partial_t \omega \cdot v + \int\limits_{\Omega} (u - V_{all}) \cdot \nabla \omega \cdot v - \int\limits_{\Omega} \alpha \nabla \omega \cdot \nabla v \\ &+ \int\limits_{\partial \Omega_s} k_s \partial_{\boldsymbol{\mu}} \omega \cdot v + \int\limits_{\partial \Omega_l} k_l \partial_{\boldsymbol{\mu}} \omega \cdot v. \end{aligned}$$

The boundary integrals are treated separately in what follows. On the corresponding parts of the boundaries the conditions (3.16b) - (3.16e) are plugged in

$$
\begin{aligned}
&\int\limits_{\partial\Omega_s} k_s \partial_{\boldsymbol{\mu}}\omega \cdot v + \int\limits_{\partial\Omega_l} k_l \partial_{\boldsymbol{\mu}}\omega \cdot v \\
=& \int\limits_{(\Gamma_N \cup \Gamma_{out}) \cap \partial\Omega_s} k_s \underbrace{\partial_{\boldsymbol{\mu}}\omega}_{\overset{(3.16c)}{=} 0} \cdot v + \int\limits_{\Gamma_H} k_s \partial_{\boldsymbol{\mu}}\omega \cdot \underbrace{v}_{\overset{(3.16d)}{=} 0} + \int\limits_{\Gamma_I} k_s \partial_{\boldsymbol{\mu}}\omega \cdot \underbrace{v}_{\overset{(3.16e)}{=} 0} \\
&+ \int\limits_{\Gamma_{out} \cap \partial\Omega_l} \underbrace{k_l \partial_{\boldsymbol{\mu}}\omega}_{\overset{(3.16b)}{=} -u \cdot (\omega \cdot \boldsymbol{\mu})} \cdot v + \int\limits_{\Gamma_N \cap \partial\Omega_l} k_l \underbrace{\partial_{\boldsymbol{\mu}}\omega}_{\overset{(3.16c)}{=} 0} \cdot v \\
&+ \int\limits_{\Gamma_H} k_l \partial_{\boldsymbol{\mu}}\omega \cdot \underbrace{v}_{\overset{(3.16d)}{=} 0} + \int\limits_{\Gamma_I} k_l \partial_{\boldsymbol{\mu}}\omega \cdot \underbrace{v}_{\overset{(3.16e)}{=} 0} \\
=& - \int\limits_{\Gamma_{out} \cap \partial\Omega_l} u \cdot (\omega \cdot \boldsymbol{\mu}) \cdot v.
\end{aligned}
$$

To discretize in time the derivative with respect to time is replaced with a difference quotient

$$
\int\limits_{\Omega} \partial_t \omega \cdot v = \int\limits_{\Omega} \frac{\omega^n - \omega^{n-1}}{\tau} \cdot v = \int\limits_{\Omega} \frac{1}{\tau}\omega^n \cdot v - \int\limits_{\Omega} \frac{1}{\tau}\omega^{n-1} \cdot v.
$$

The latter equations are used to define the bilinear

$$
\begin{aligned}
a_\omega(\omega^{n-1}, v) :=& \int\limits_{\Omega} \frac{1}{\tau}\omega^{n-1} \cdot v - \int\limits_{\Omega} (u^{n-1} - V_{all}^{n-1}) \cdot \nabla\omega^{n-1} \cdot v \\
&+ \int\limits_{\Omega} \alpha \nabla\omega^{n-1} \cdot \nabla v + \int\limits_{\Gamma_{out} \cap \partial\Omega_l} u^{n-1} \cdot (\omega^{n-1} \cdot \boldsymbol{\mu}) \cdot v,
\end{aligned}
$$

and the linear form for the adjoint temperature ω

$$
L_\omega(v) := \int\limits_{\Omega} \frac{1}{\tau}\omega^n \cdot v.
$$

Thus, the weak formulation reads

$$
a_\omega(\omega, v) = L_\omega(v), \qquad \text{for all } t \in [t_0, \dots, t_{N-1}], \text{and for all } v \in \mathcal{O}.
$$

Adjoint Navier–Stokes Equations

For the numerical solution of the adjoint Navier–Stokes equations (3.24) the adjoint velocity γ and adjoint pressure π are approximated at time t_{n-1} with $\mathbb{P}_2/\mathbb{P}_1$ Taylor–Hood finite elements:

$$\begin{aligned} \gamma^{n-1} &\in \mathcal{G} := \{v \in C^0(\Omega_l)^2 : v|_S \in \mathbb{P}_2^2, S \in \Sigma_l^{n-1}\}, \\ \pi^{n-1} &\in \mathcal{J} := \{v \in C^0(\Omega_l) : v|_S \in \mathbb{P}_1, S \in \Sigma_l^{n-1}\}. \end{aligned} \tag{4.3}$$

Multiplication with test functions $v \in \mathcal{G}$ and $q \in \mathcal{J}$ and integration over Ω_l, together with integration by parts, leads to

$$\begin{aligned} 0 =& \int\limits_{\Omega_l} (\partial_t \gamma - (\nabla u)^T \cdot \gamma + ((u - V_{all}) \cdot \nabla)\gamma + \eta \Delta \gamma + \nabla \pi - \omega \nabla T) \cdot v + \int\limits_{\Omega_l} \nabla \cdot \gamma \cdot q \\ =& \int\limits_{\Omega_l} \partial_t \gamma \cdot v - \int\limits_{\Omega_l} (\nabla u)^T \cdot \gamma \cdot v + \int\limits_{\Omega_l} ((u - V_{all}) \cdot \nabla)\gamma \cdot v + \int\limits_{\Omega_l} \nabla \pi \cdot v \\ & - \int\limits_{\Omega_l} \omega \nabla T \cdot v + \int\limits_{\Omega_l} \nabla \cdot \gamma \cdot q + \int\limits_{\partial\Omega_l} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot v - \int\limits_{\Omega_l} \eta \nabla \gamma \cdot \nabla v. \end{aligned}$$

The integral over the boundary can be split up and reads

$$\begin{aligned} & \int\limits_{\partial\Omega_l} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot v \\ =& \int\limits_{\Gamma_I \cup (\Gamma_N \cap \partial\Omega_l)} \eta \partial_{\boldsymbol{\mu}} \gamma \cdot \underbrace{v}_{\overset{(3.24c)}{=} 0} + \int\limits_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} \underbrace{\eta \partial_{\boldsymbol{\mu}} \gamma}_{\overset{(3.24b)}{=} -(\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) - \pi \cdot \boldsymbol{\mu}} \cdot v \\ =& - \int\limits_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} (\gamma \cdot \boldsymbol{\mu}) \cdot (u - V_{all}) \cdot v - \int\limits_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} \pi \cdot \boldsymbol{\mu} \cdot v. \end{aligned}$$

To discretize in time, the derivative of γ with respect to time is replaced by a finite difference:

$$\int\limits_{\Omega_l} \partial_t \gamma \cdot v = \int\limits_{\Omega_l} \frac{\gamma^n - \gamma^{n-1}}{\tau} \cdot v = \int\limits_{\Omega_l} \frac{1}{\tau} \gamma^n \cdot v - \int\limits_{\Omega_l} \frac{1}{\tau} \gamma^{n-1} \cdot v.$$

Substituting this into the latter equations, leads to the multilinear form

$$\begin{aligned} a_{\gamma,\pi}(\gamma^{n-1}, \pi^{n-1}, v, q) :=& \int\limits_{\Omega_l} \frac{1}{\tau} \gamma^{n-1} \cdot v + \int\limits_{\Omega_l} (\nabla u^{n-1})^T \cdot \gamma^{n-1} \cdot v \\ & - \int\limits_{\Omega_l} ((u^{n-1} - V_{all}^{n-1}) \cdot \nabla)\gamma^{n-1} \cdot v - \int\limits_{\Omega_l} \nabla \pi^{n-1} \cdot v \\ & - \int\limits_{\Omega_l} \nabla \cdot \gamma^{n-1} \cdot q + \int\limits_{\Gamma_H \cup (\Gamma_{out} \cap \partial\Omega_l)} (\gamma^{n-1} \cdot \boldsymbol{\mu}) \cdot (u^{n-1} - V_{all}^{n-1}) \cdot v \end{aligned}$$

$$+\int\limits_{\Gamma_H\cup(\Gamma_{out}\cap\partial\Omega_l)} \pi^{n-1}\cdot\boldsymbol{\mu}\cdot v+\int\limits_{\Omega_l}\eta\nabla\gamma^{n-1}\cdot\nabla v,$$

and bilinear form

$$L_\gamma(v,q):=\int\limits_{\Omega_l}\frac{1}{\tau}\gamma^n\cdot v-\int\limits_{\Omega_l}\omega^{n-1}\nabla T^{n-1}\cdot v,$$

which result in the weak formulation

$$a_\gamma(\gamma,\pi,v,q)=L_{\gamma,\pi}(v,q),\qquad \text{for all } t\in[t_0,\dots,t_{N-1}], \text{and for all } v\in\mathcal{G}, q\in\mathcal{J}.$$

5. Numerical Examples

In this chapter, the general setting and three examples are discussed to validate the presented optimal control approach of a Stefan problem numerically. The first two examples in section 5.2 and section 5.3 aim to stabilize the interface to a flat position. They demonstrate that not all desirable interface positions are reachable due to the model chosen in this thesis. Further, the importance of well-chosen weights $\lambda, \Lambda, \bar{\Lambda}$ in the cost functional is highlighted. In contrast to the first two settings, the controlled interface moves upwards in the third example (section 5.4). It is used to show the influence of the selection of an initial guess to the convergence of the projected gradient method.

5.1. General Setting

The domain described in chapter 2 is a unit square $\Omega = [0,1] \times [0,1]$. The boundary regions are

$$\begin{aligned}
\Gamma_H &= \{0\} \times [0.6, 0.8], \\
\Gamma_C &= [0,1] \times \{0\}, \\
\Gamma_{out} &= \{1\} \times [0.2, 0.4], \\
\Gamma_{\tilde{N}} &= [0,1] \times \{1\}, \\
\Gamma_N &= (\{0\} \times ([0,0.6] \cup [0.8,1])) \cup (\{1\} \times ([0,0.2] \cup [0.4,1])) \cup ([0,1] \times \{1\}),
\end{aligned}$$

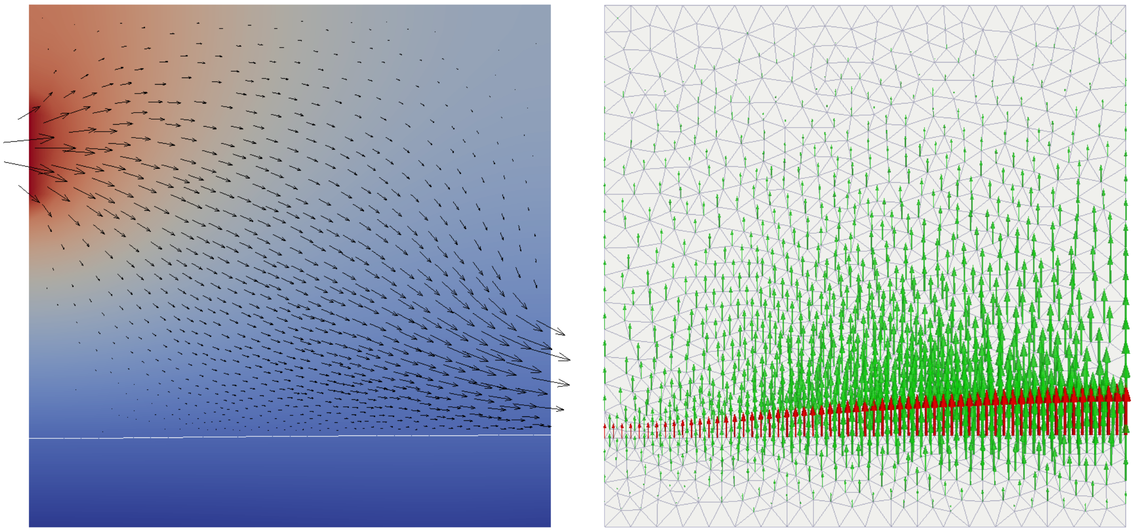
and the initial interface position is $\Gamma_I = [0,\ 1] \times \{\frac{1}{6}\}$. The constants for the two-phase Stefan problem are

$$T_C = -0.6,\ T_H = 4,\ T_m = 0,\ \eta = 0.05,\ k_s = 1,\ k_l = 0.6,\ L = 150.$$

As initial temperature distribution serves $T_0 = 4y - \frac{2}{3}$. The tolerances and maximum iteration numbers of the gradient algorithm and the line minimization are

$$\begin{gathered}
\delta_1 = 10^{-8},\ \delta_2 = 10^{-4},\ k_{max} = 100, \\
\epsilon_1 = 10^{-12},\ \epsilon_2 = 10^{-4},\ \epsilon_3 = 10^{-4},\ \delta = 0.05,\ i_{max} = 5.
\end{gathered}$$

As described in section 4.1, the domain is separated into 2388 triangles with 3622 edges and 1235 vertices. 63 of the vertices are positioned on the interface. For the computation of the interface graph h, the interval $\Gamma_C = [0,1]$ is partitioned into 69 equally distributed intervals. This results in 141 data points for the graph h due to its quadratic finite element approximation. The terminal time is set to $E = 1$ and the time interval is discretized with step size 0.01 into 100 intervals. Figure 5.1 illustrates the numerical solution of (3.9) with $p_{in} = 1$ constant.



(a) Temperature distribution (background color) with velocity in the liquid (black arrows) and interface position (white line).

(b) Interface velocity (red arrows) extended to the whole domain (green arrows).

Figure 5.1.: Numerical solution of the forward problem at $t = 1$.

The control constraints are set to

$$\underline{c} := 0 \leq p_{in}(t) \leq 20 =: \overline{c} \qquad \text{for all } t \in [0,1].$$

The lower bound ensures that the inflow and outflow boundary are not switched. The upper bound is necessary in some cases since the robustness of the FEniCS internal solvers is limited. This affects the Navier–Stokes equations and the adjoint Navier–Stokes equations. If $p_{in} > 20$ for more than 20 time steps in this setting, the nonlinear Navier–Stokes-solver may not converge. For the adjoint Navier–Stokes equations the linear solver gets inaccurate and reveals certain “blow-up” effects. These problems can be overcome by reducing the time step size. But the discretization of the time interval is constant throughout the execution of the gradient method. In the majority of cases, the control constraints are inactive for the computed control. Nevertheless, the control constraints can become active for the sample points within the line minimization algorithm if the step size is overestimated. This behavior mainly depends on the choice of the weight parameters in the cost functional.

5.2. Stabilizing to a Flat Position

The desired interface position is a straight line moving from the start position at $y = \frac{1}{6}$ to $y = 0.166$:

$$h_d(x,t) = \frac{1}{6} - t \cdot (\frac{1}{6} - 0.166).$$

The sought control should keep the interface as flat as possible and prevent it from moving away. The inner boundary is requested to move slightly downwards by $\frac{2}{3} \cdot 10^{-3}$.

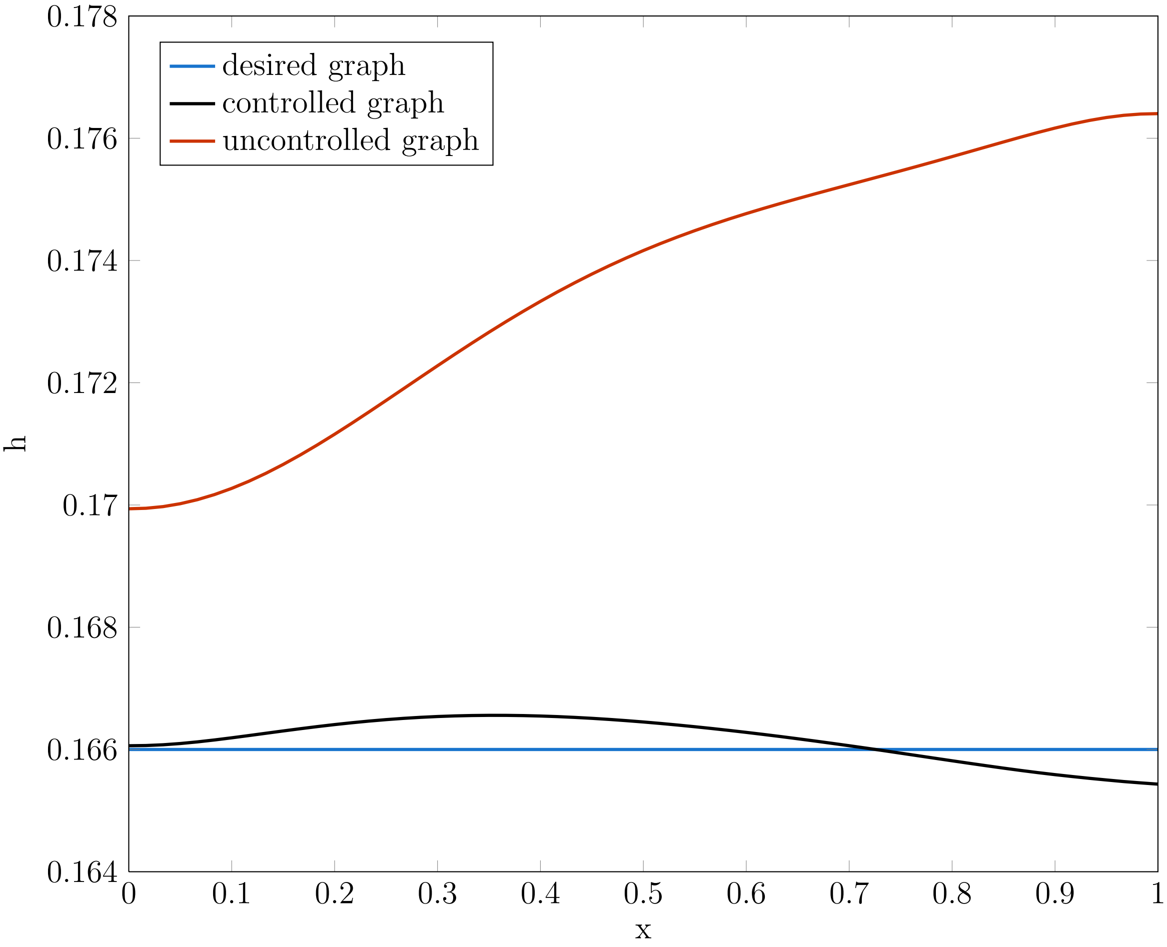


Figure 5.2.: Interface graphs with $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$.

The weight parameters in the cost functional are set to

$$\lambda = 10^{-10},\ \Lambda = 100,\ \bar{\Lambda} = 0.$$

So the cost functional primarily measures the distance of the interface to the desired interface at the end of the time interval and does not track the interface position for all points of time. As initial guess for the projected gradient algorithm, we use $c_1^0(t) \equiv 1$.

The described problem is not symmetric. The fluid in the liquid phase always flows from left to right and the transported heat acts on the interface position more on the right side near the outflow boundary than on the left side. Thus, the interface velocity V_{Γ_I} tends to take values of lager magnitude on the right side. Additionally, the initial interface position is closer to the cooling boundary than to the heat source. Thereby the interface always moves upwards in the beginning of a forward simulation. Then, it has to be moved back downwards for this example. This causes the interface graph to form a shallow hilltop in the middle. These asymmetric effects implicate that the controlled interface can not be expected to be completely flat and to match the desired interface perfectly.

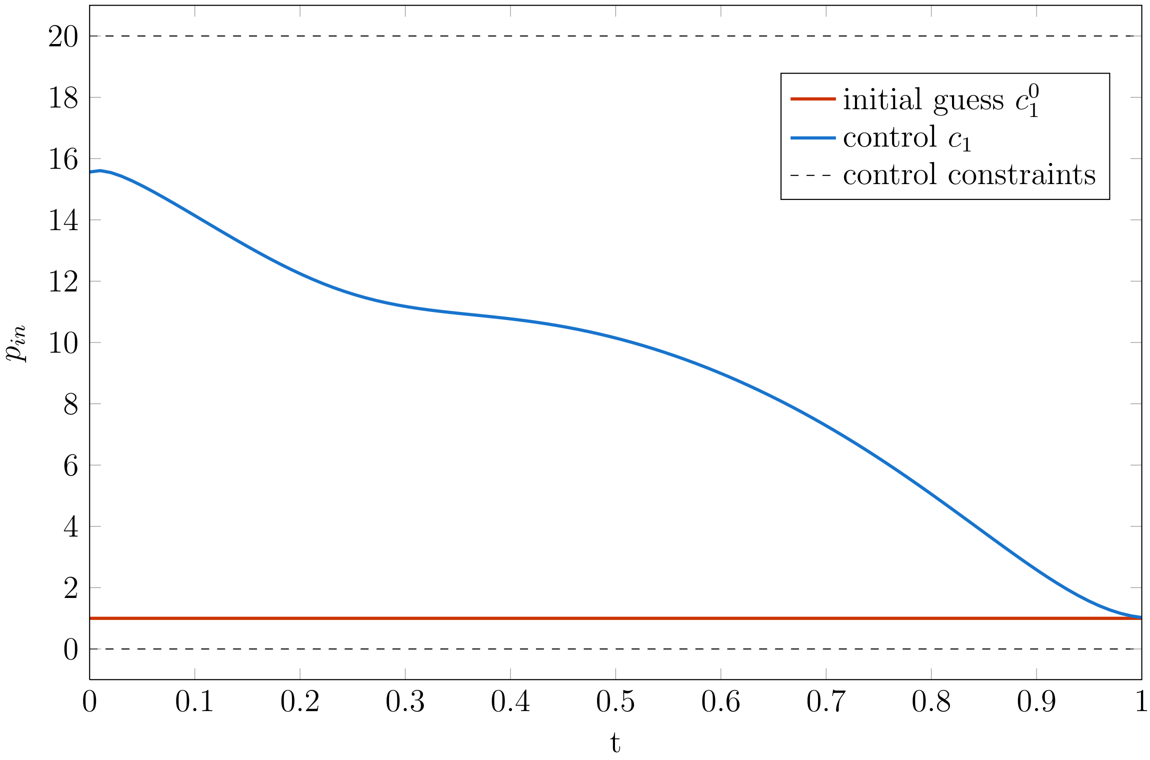


Figure 5.3.: Computed control for $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$.

Figure 5.2 shows the interface graphs of the controlled interface compared to the uncontrolled interface. The presented algorithm is able to compute a control c_1 (figure 5.3) which keeps the interface close to the desired interface. The distance and control cost terms in (3.10) are

$$\begin{aligned}
\mathfrak{d} &:= \int_{\Gamma_C} (h(E,x) - h_d(E,x))^2 &&= 1.4317 \cdot 10^{-7}, \\
\mathfrak{d}_{all} &:= \int_0^E \int_{\Gamma_C} (h(t,x) - h_d(t,x))^2 &&= 7.0578 \cdot 10^{-7}, \\
\mathfrak{p} &:= \int_0^E \int_{\Gamma_H} (p_{in}(t))^2 &&= 15.38.
\end{aligned}$$

The results for this and all the following examples can be found in A.3 – A.5 likewise. The pathway of the controlled interface is illustrated in figure 5.4. The computed control c_1, which can be seen in figure 5.3, mainly acts at the beginning of the time interval to stop the interface from moving upwards and moves it back downwards to the desired position. The control constraints are inactive at all points.

Algorithm 1 stopped after 4 iteration steps caused by stopping criterion (3.38). Figure 5.5 shows the cost functional at every iteration step.

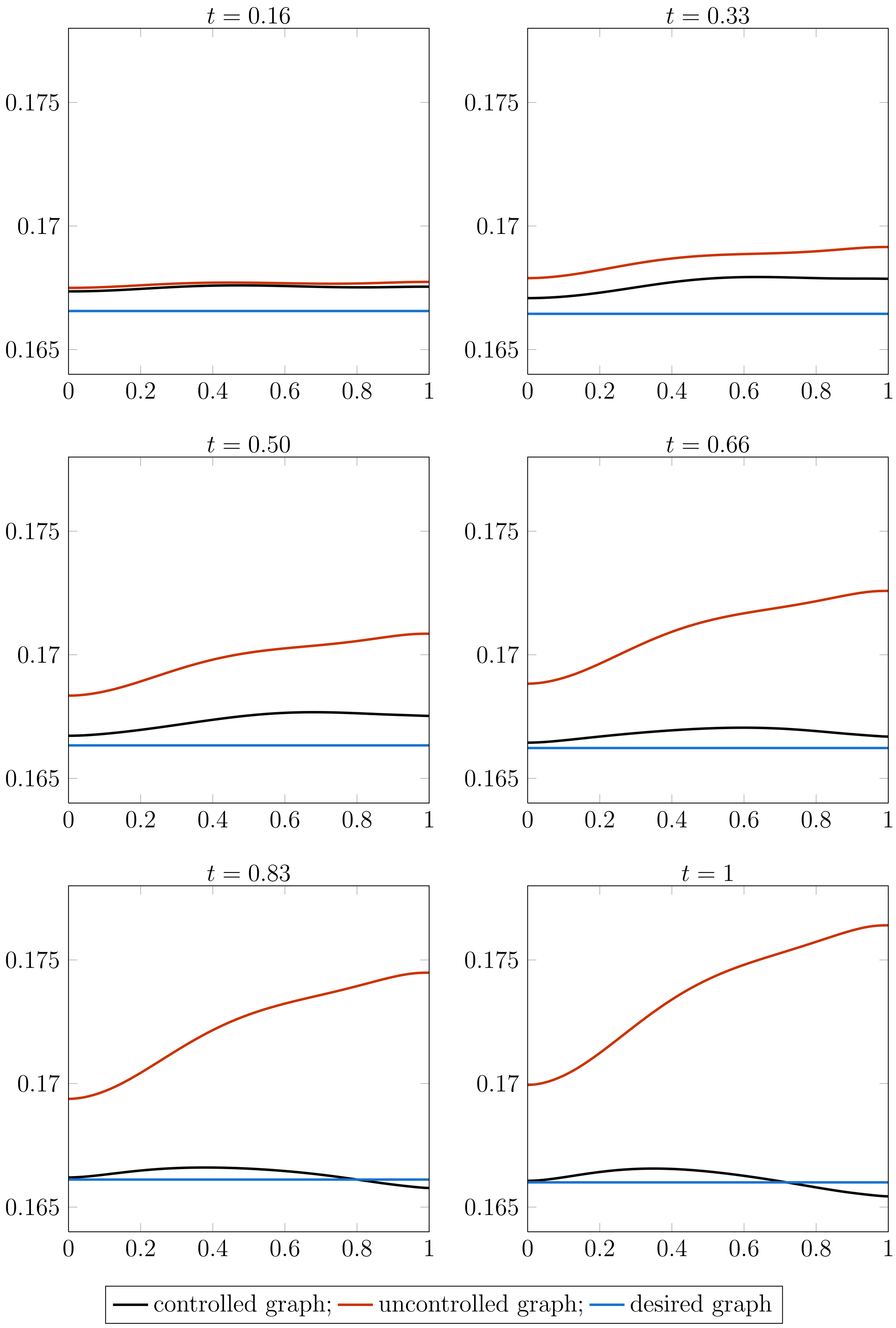


Figure 5.4.: Interface graphs with $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$ at various t.

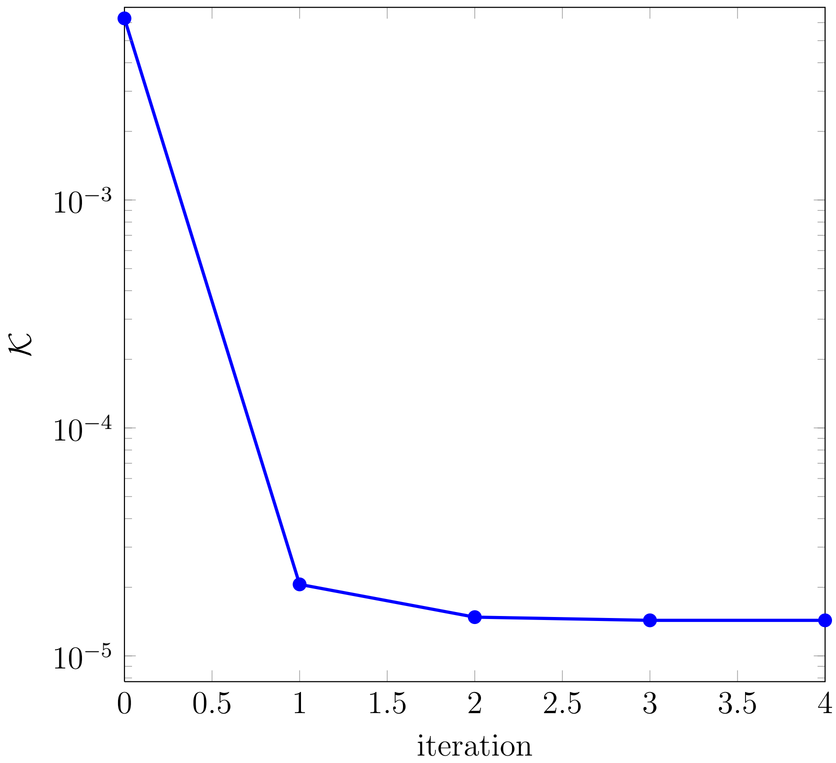


Figure 5.5.: Cost functional for $\lambda = 10^{-10}, \Lambda = 100, \bar{\Lambda} = 0$.

The convergence behavior is influenced by the choice of the weight factors in the cost functional (3.10). Running the algorithm with different sets of weights

$$\begin{aligned} c_1 &: \lambda = 10^{-10}, \Lambda = 10^2, \bar{\Lambda} = 0, \\ c_2 &: \lambda = 10^{-10}, \Lambda = 10^5, \bar{\Lambda} = 10^4, \\ c_3 &: \lambda = 10^{-10}, \Lambda = 10^5, \bar{\Lambda} = 10^3, \\ c_4 &: \lambda = 10^{-10}, \Lambda = 10^4, \bar{\Lambda} = 10^5 \end{aligned}$$

changes the convergence speed and quality of the computed control. With $\bar{\Lambda} \neq 0$ the interface position is tracked over the whole time interval by the cost functional. Since the cost functional is not comparable among these parameter sets, instead of the cost functional, the two distances $\mathfrak{d}$ and $\mathfrak{d}_{all}$ are displayed in figures 5.6 and 5.7. In both figures the values for the initial value are omitted since they are identical. In case of control c_3, both distances are larger than with control c_1. But for control c_4 the interface can be moved closer to the desired position over the whole time interval at the expense of a larger distance at $t = E$.

Since the two-phase Stefan problem is non-linear, the cost functional must be assumed non-convex [8]. Consequently, the projected gradient algorithm can only approximate stationary points of the cost functional. To which stationary point the algorithm converges, primarily depends on the initial guess.

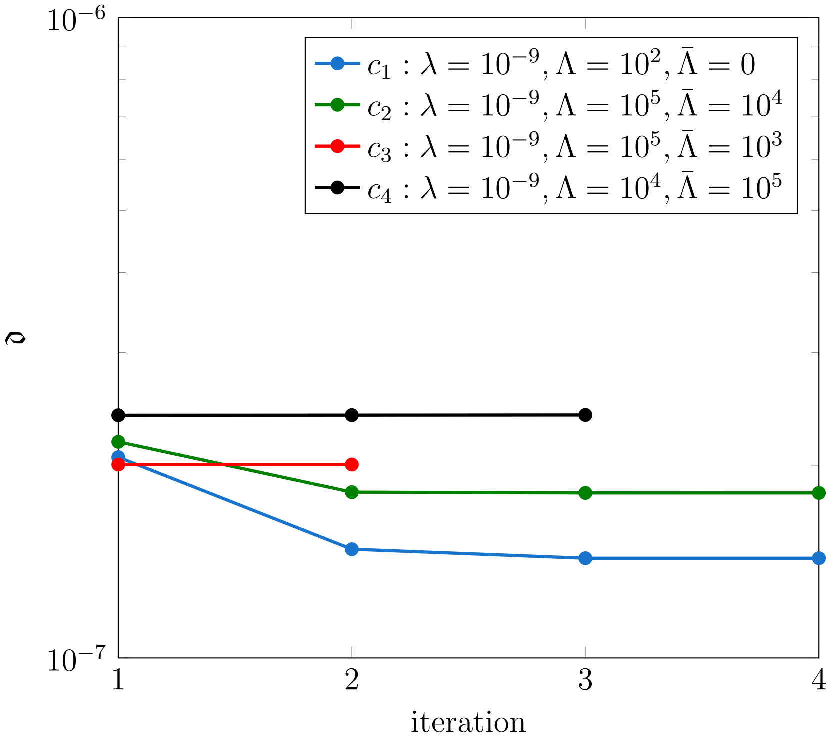


Figure 5.6.: Interface distance $\mathfrak{d}$ at $t = E$ for different parameter sets.

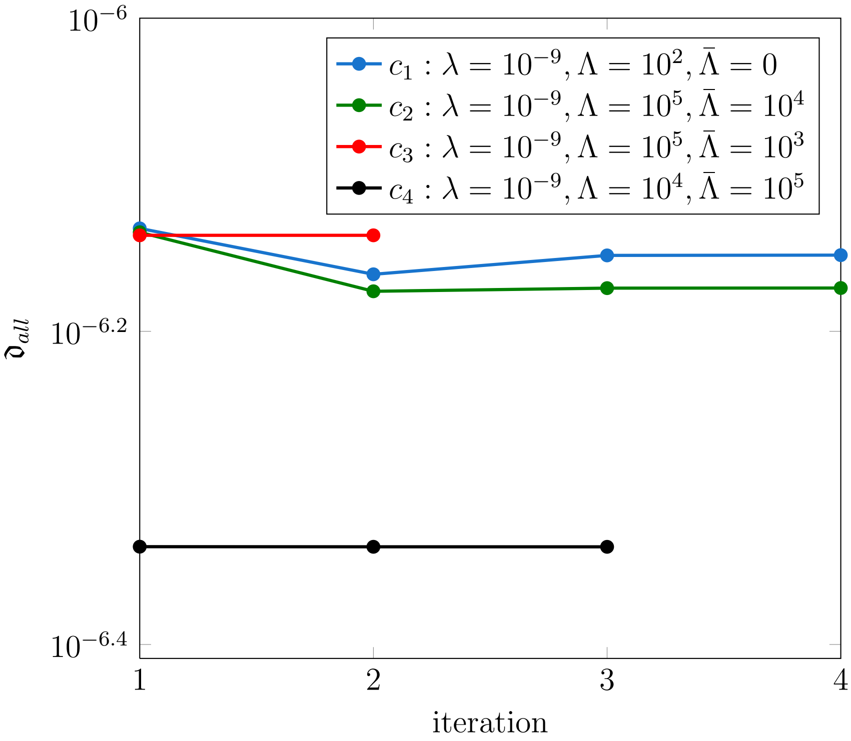


Figure 5.7.: Interface distance $\mathfrak{d}_{all}$ for the whole time interval for different parameter sets.

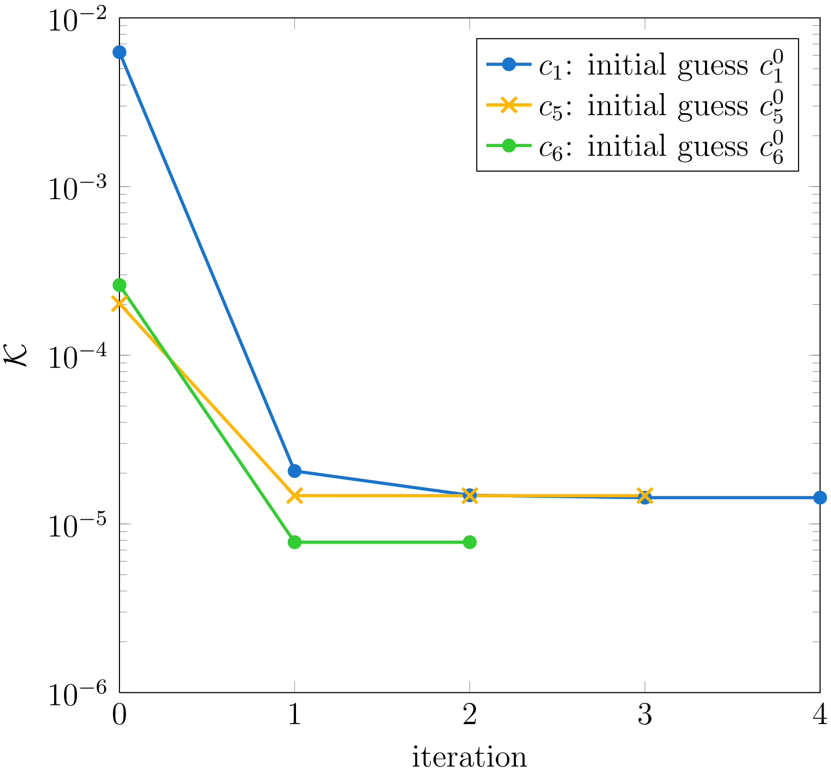


Figure 5.8.: Cost functional for different initial guesses.

The following functions are taken as alternative initial guesses

$$c_5^0(t) \equiv 12,$$
$$c_6^0(t) = \begin{cases} 20, & t \in [0, 0.17], \\ 10, & t \in (0.17, 1]. \end{cases}$$

Both intend to induce higher velocities of the fluid at the beginning of the time interval to prevent the interface from moving upwards. Through this, the algorithm is expected to show better convergence behavior.

The weights of the cost function are set to the same values as for c_1:

$$\lambda = 10^{-10}, \ \Lambda = 100, \ \bar{\Lambda} = 0.$$

The projected gradient algorithm stops after 3 iterations with the control c_5 with stopping criterion (3.38). As expected, it converges slightly faster with the initial guess c_5^0 than with c_1^0 but does not reach a considerable smaller cost value (see figure 5.8). On the contrary, the algorithm converges clearly faster towards c_6 which also has a notable smaller cost value. In this case, it stopped after 2 iterations with stopping criterion (3.37). Looking at the computed controls in figure 5.9, the algorithm appears to converge to completely different stationary points. And also the interface graphs in figure 5.10 are different from each other. Again the control constraints are inactive for all controls and all points of time.

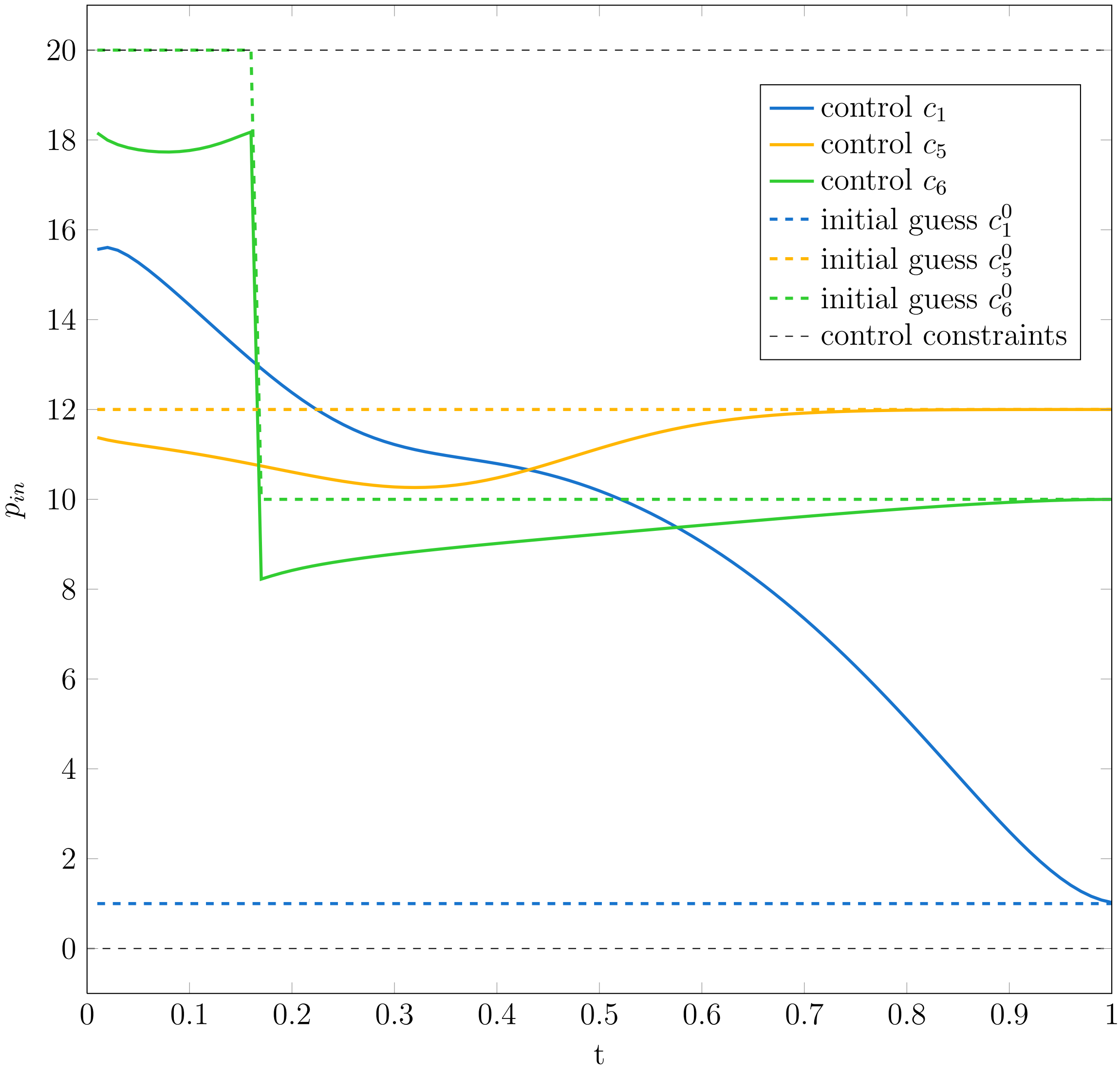


Figure 5.9.: Computed controls c_1, c_5, c_6 with different initial guesses.

The algorithm seems to be unable to dissolve the controls from the initial guess near the terminal time $t = E$. This is due to the structure of the backward system (3.32) on which the gradient of the cost functional is dependent, which is used as the descent direction to update the control. The adjoint state φ occurring in the gradient is coupled with the adjoint velocity γ. The initial value $\gamma(E)$ is equal to zero on the liquid phase Ω_l. So the adjoint state γ has to "build momentum" during the first backwards time steps. On the one hand, this is a limitation to the adjoint-based optimal control approach. On the other hand, the control variable $p_{in}(t)$ has less influence on the behavior of the forward system and thus on the interface position for $t \to E$. Therefore, this behavior of the algorithm makes sense from the gradient point of view.

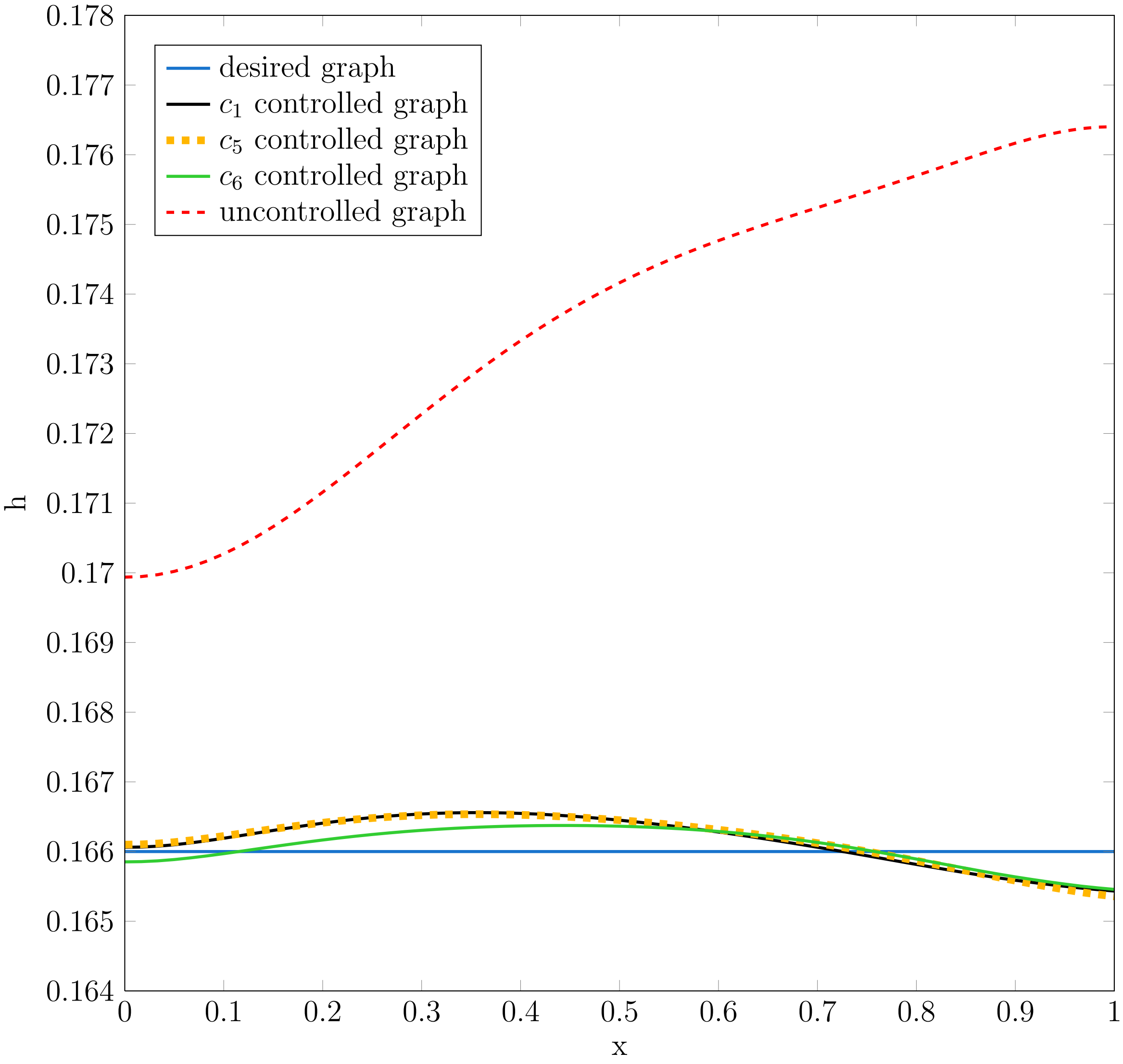


Figure 5.10.: Interface graphs for the controls c_1, c_5, c_6.

For the control c_6 the distance and control cost terms in (3.10) are

$$\begin{aligned}\mathfrak{d} &= 7.7936 \cdot 10^{-8},\\ \mathfrak{d}_{all} &= 6.0702 \cdot 10^{-7},\\ \mathfrak{p} &= 19.481.\end{aligned}$$

The cost for the control is higher than for c_1 but the distances are significantly smaller.

Nevertheless, the algorithm is not able to find a control which matches the desired interface perfectly due to the already mentioned asymmetry of the problem. Hence, the performance of algorithm 1 is analyzed for an actually reachable desired interface position in the next section.

5.3. Stabilizing to a Reachable Flat Position

To target a desired interface position which is actually reachable by the system, the forward simulation runs with the following control

$$\tilde{c}_d := \begin{cases} 20, & t \in [0, 0.17], \\ 8.5, & t \in (0.17, 0.25], \\ 7.5, & t \in (0.25, 0.63), \\ 8.5, & t \in [0.63, 1]. \end{cases}$$

With the initial guess and weight parameters set as

$$\lambda = 10^{-10}, \ \Lambda = 10^5, \ \bar{\Lambda} = 0,$$
$$\tilde{c}_1^0(t) \equiv 1,$$

the control $\tilde{c}_1$ computed by the projected gradient algorithm is able to approximate the desired interface position closely as can be seen in figure 5.11, where it is almost indistinguishable from the desired interface graph.

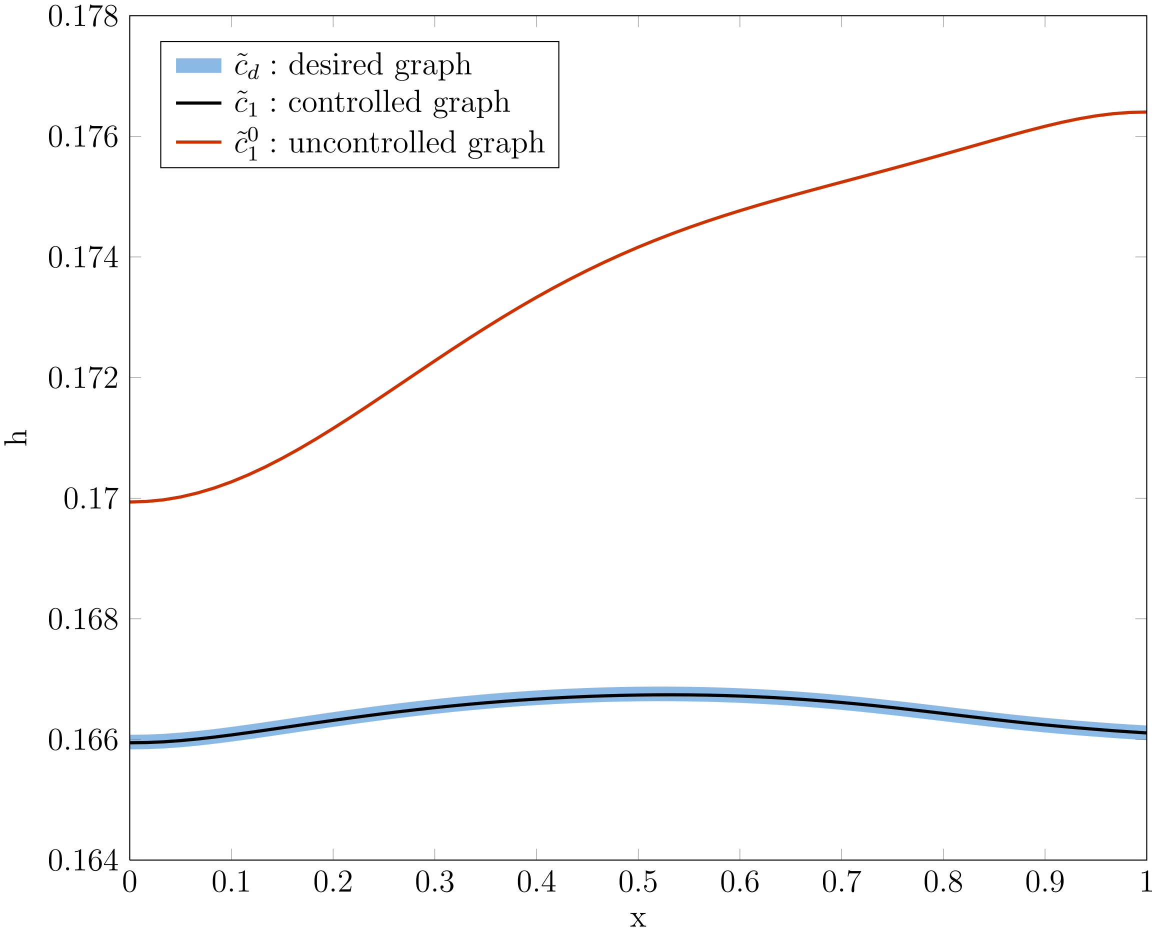


Figure 5.11.: Interface graphs with a reachable interface position $\tilde{c}_d$.

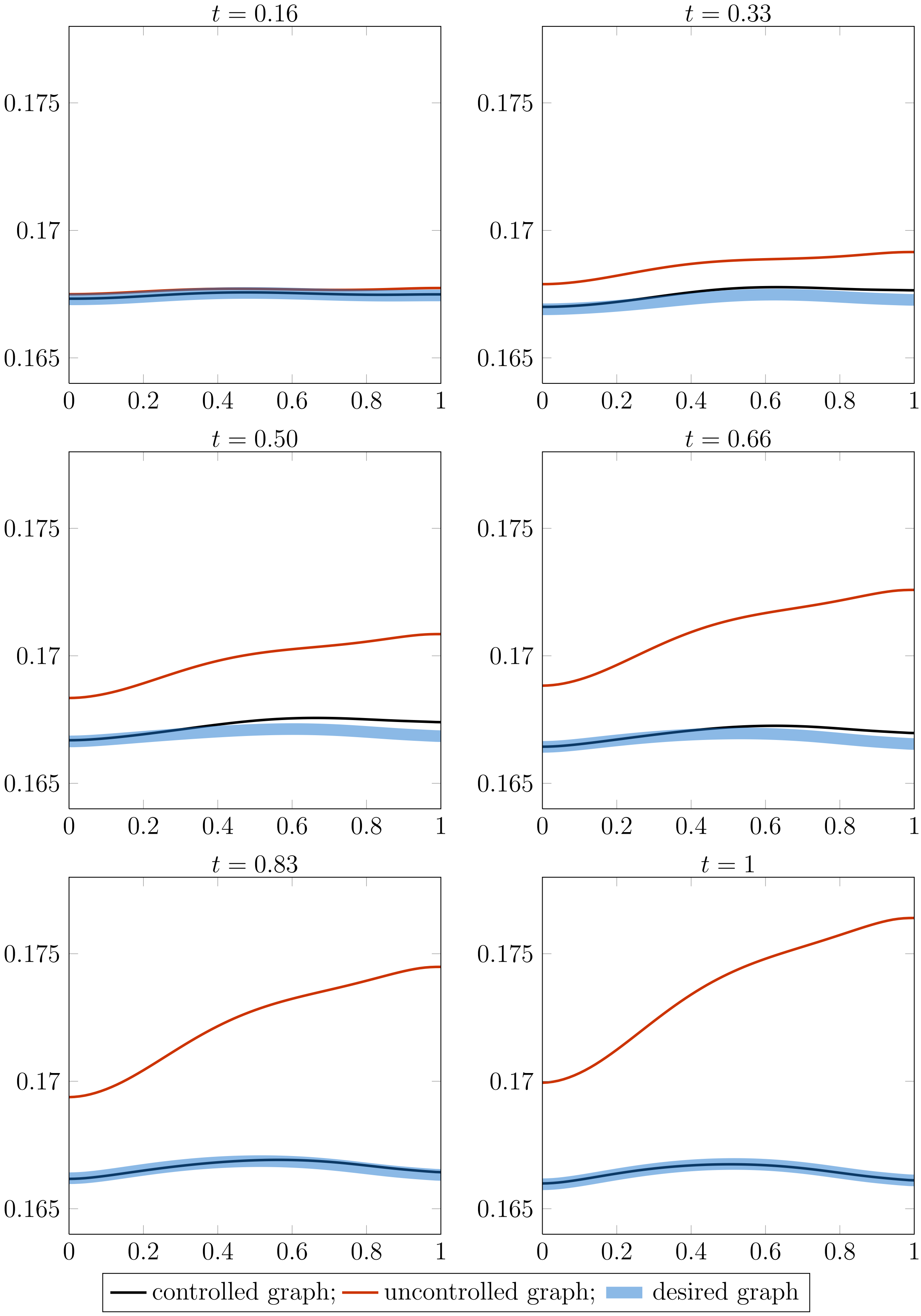


Figure 5.12.: Interface graphs with a reachable interface position $\tilde{c}_d$ and control $\tilde{c}_1$ for various t.

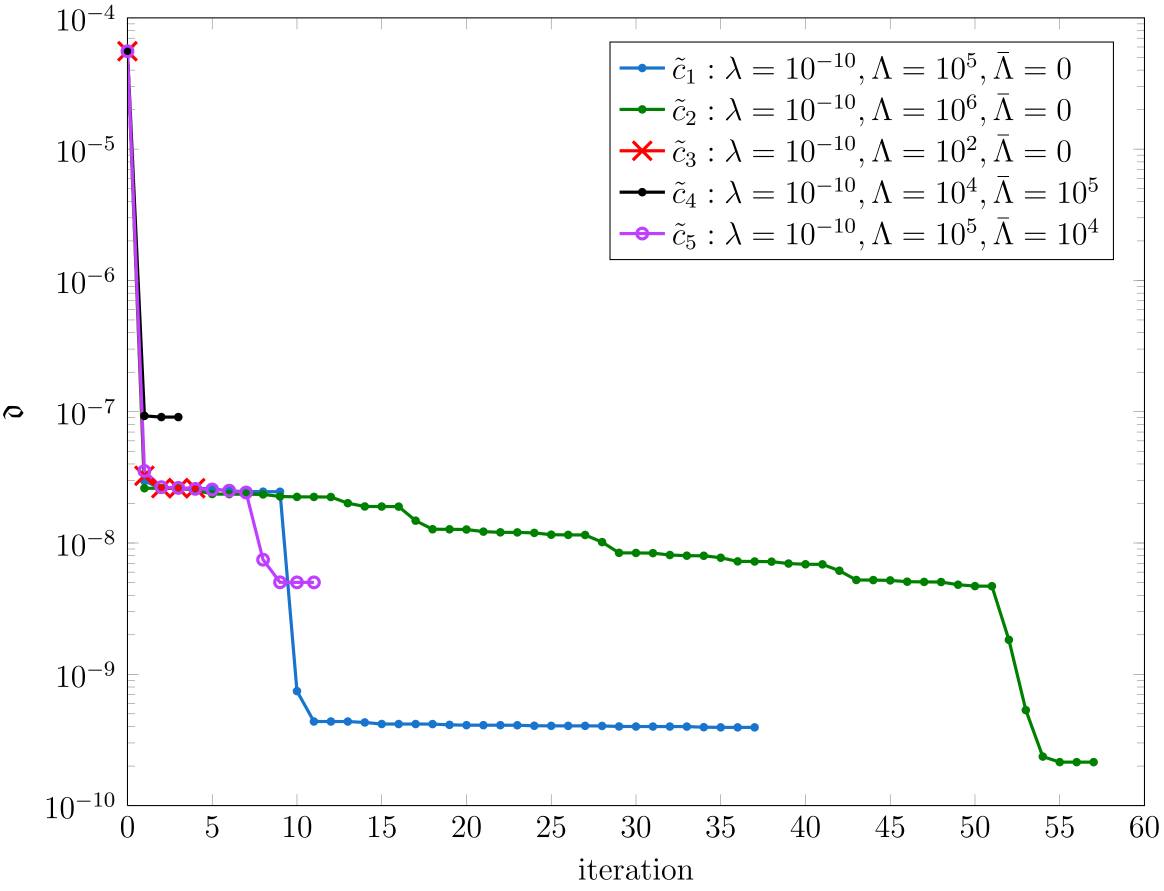


Figure 5.13.: Interface distance $\mathfrak{d}$ at $t = E$ for different parameter sets.

The pathway of the interface graph is illustrated in figure 5.12. The distance and control cost terms in (3.10) for this example confirm the quality of the computed control

$$\begin{aligned}\mathfrak{d} &= 3.9422 \cdot 10^{-10},\\ \mathfrak{d}_{all} &= 4.5865 \cdot 10^{-8},\\ \mathfrak{p} &= 15.6606.\end{aligned}$$

As for the previous example in section 5.2, the weight factors in the cost functional influence the convergence behavior and destination. Figure 5.13 and 5.14 present the distance terms in the cost functional for the iteration steps of the projected gradient algorithm with the following sets of parameters

$$\begin{aligned}\tilde{c}_1 &: \lambda = 10^{-10}, \Lambda = 10^5, \bar{\Lambda} = 0,\\ \tilde{c}_2 &: \lambda = 10^{-10}, \Lambda = 10^6, \bar{\Lambda} = 0,\\ \tilde{c}_3 &: \lambda = 10^{-10}, \Lambda = 10^2, \bar{\Lambda} = 0,\\ \tilde{c}_4 &: \lambda = 10^{-10}, \Lambda = 10^4, \bar{\Lambda} = 10^5,\\ \tilde{c}_5 &: \lambda = 10^{-10}, \Lambda = 10^5, \bar{\Lambda} = 10^4.\end{aligned}$$

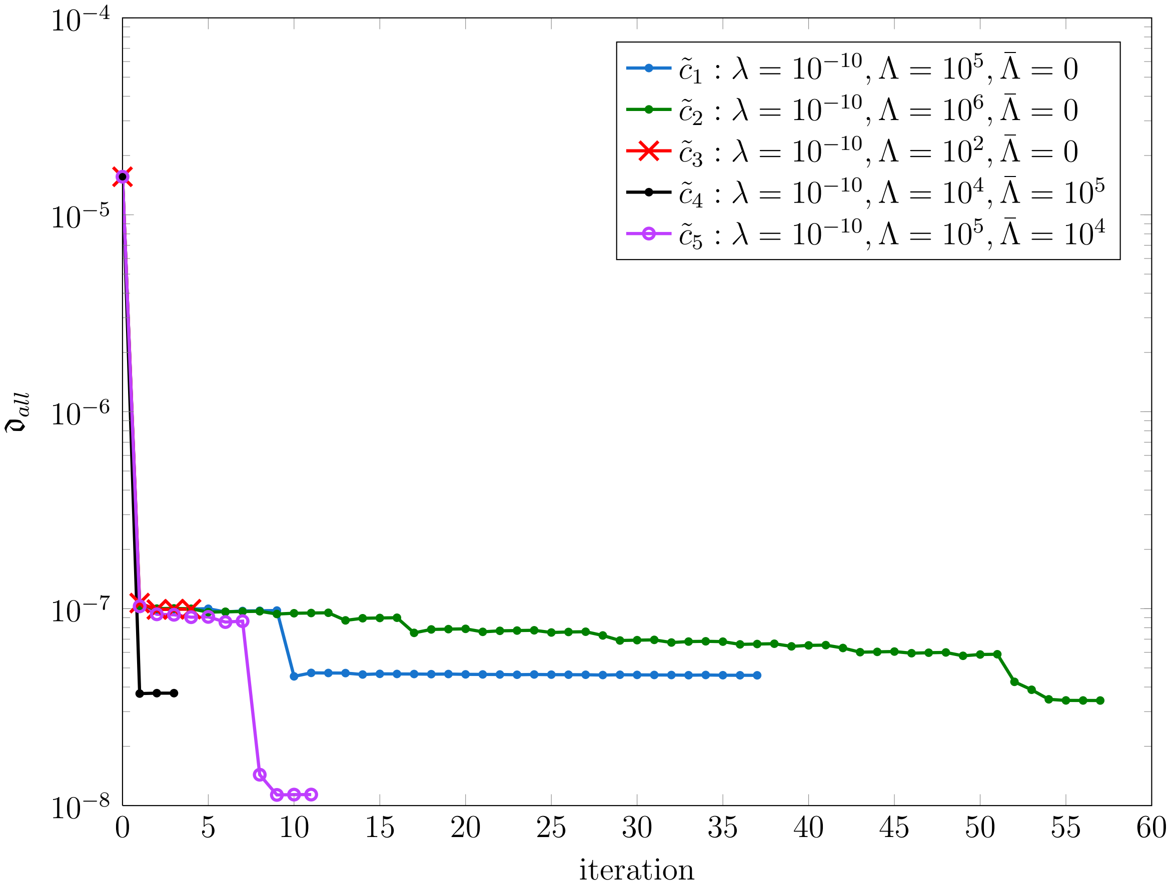


Figure 5.14.: Interface distance $\mathfrak{d}_{all}$ the whole time interval for different parameter sets.

Once more, the distance of the interface over the whole time interval can be reduced by setting the corresponding weight $\bar{\Lambda}$ unequal to zero. But in contrast to the example from the latter section, the factors that produce good results there do not necessarily produce good results here and vice versa. This effect gets even more obvious when the alternative initial guesses

$$\tilde{c}_6^0(t) := c_5^0(t) \equiv 12,$$
$$\tilde{c}_7^0(t) := c_6^0(t) = \begin{cases} 20, & t \in [0, 0.17], \\ 10, & t \in (0.17, 1], \end{cases}$$

are used as input for the gradient algorithm with different weight factors in the cost functional. The results are illustrated in figure 5.15. With input $\tilde{c}_6^0$ and $\tilde{c}_7^0$, the algorithm converges faster in the first iteration steps. However, it does not reach the same accuracy as the control $\tilde{c}_1$. If the same weight factors as for $\tilde{c}_1$ and $\tilde{c}_7$ are used together with $\tilde{c}_6^0$ as input, the algorithm even fails to converge (A.4).

The factor λ, which penalizes the control cost $\mathfrak{p}$ in (3.10), can be used to reduce the control cost and acts as a regularization. This could make the control constraints dispensable. These constraints were inactive for all computed controls by now.

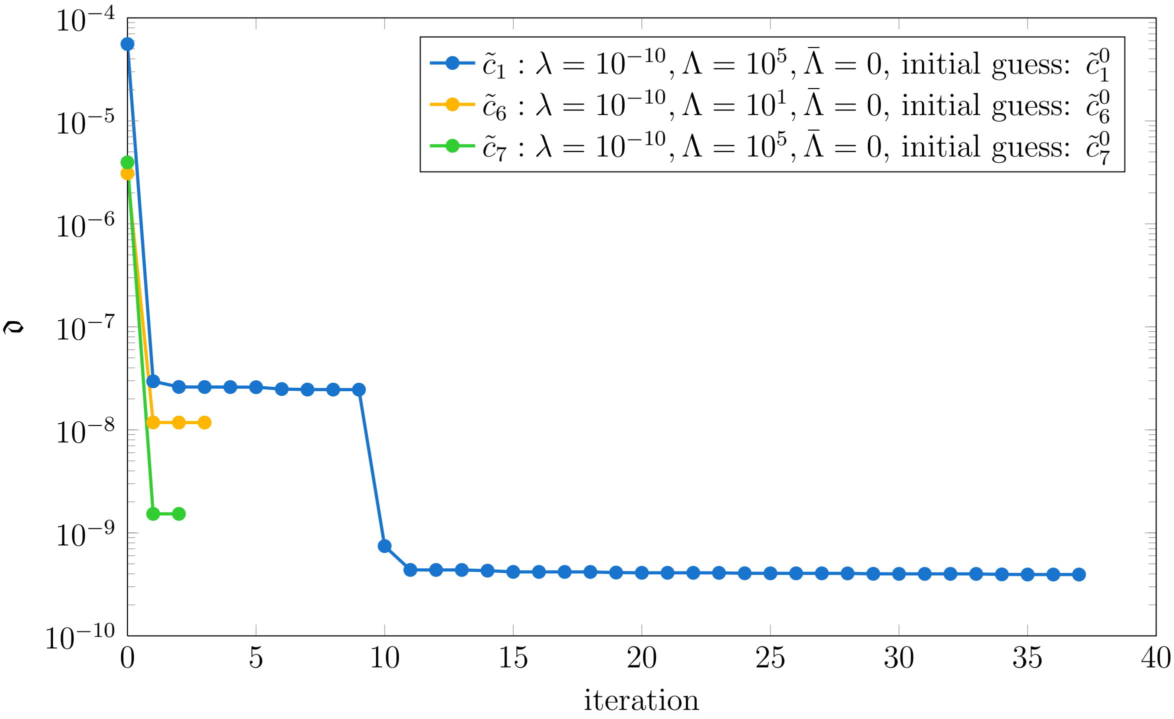


Figure 5.15.: Interface distance $\mathfrak{d}$ at $t = E$ for different initial guesses and parameter sets.

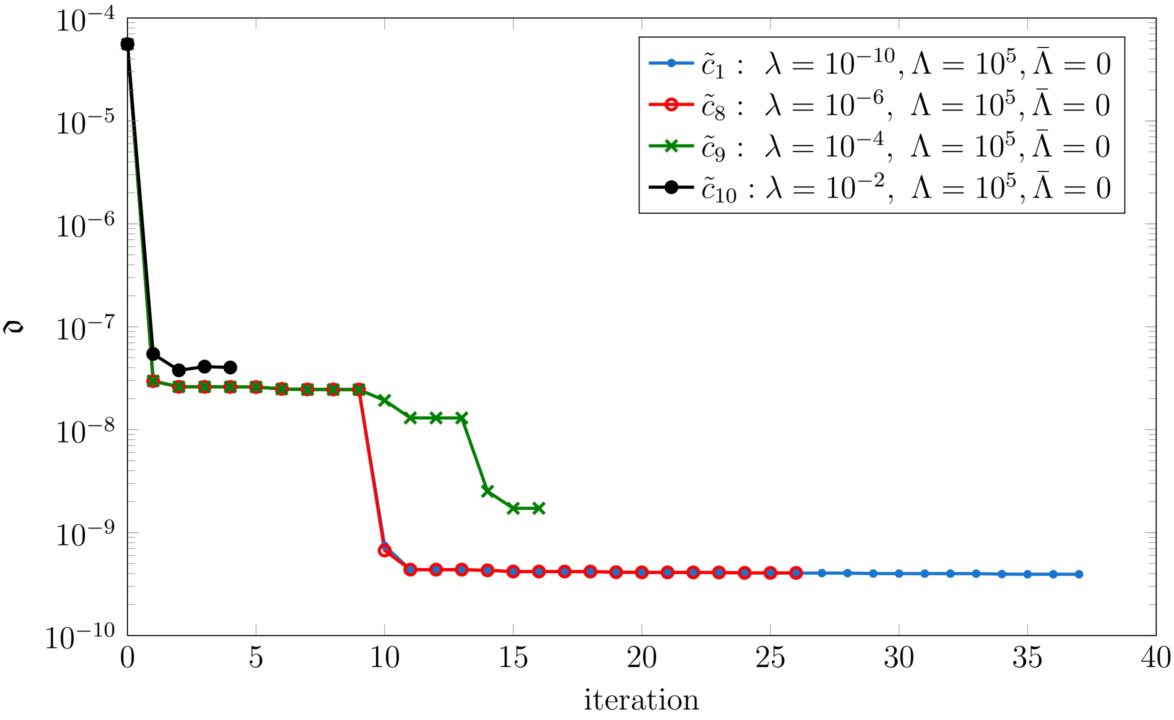


Figure 5.16.: Interface distance $\mathfrak{d}$ at $t = E$ for different values of λ.

For completeness, the influence of λ on the convergence behavior is analyzed nevertheless. The interface distances at the end time $t = E$ are displayed in figure 5.16 for the following parameter sets

$$\begin{aligned}
\tilde{c}_1 &: \ \lambda = 10^{-10}, \Lambda = 10^5, \bar{\Lambda} = 0,\\
\tilde{c}_8 &: \ \lambda = 10^{-6},\ \ \Lambda = 10^5, \bar{\Lambda} = 0,\\
\tilde{c}_9 &: \ \lambda = 10^{-4},\ \ \Lambda = 10^5, \bar{\Lambda} = 0,\\
\tilde{c}_{10} &: \lambda = 10^{-2},\ \ \Lambda = 10^5, \bar{\Lambda} = 0.
\end{aligned}$$

For the control $\tilde{c}_8$ with $\lambda = 10^{-6}$, the differences in the cost function terms are negligible. By further increasing λ, the control cost $\mathfrak{p}$ can be reduced (see A.4), but this also leads to higher distances $\mathfrak{d}$ and $\mathfrak{d}_{all}$.

Controlling the interface in the opposite direction works also satisfactorily as can be seen in the next section.

5.4. Moving to a Reachable Position

For this example, the desired and uncontrolled interface positions of the previous examples are switched. The interface is requested to move from the start position upwards. The desired graph is produced with the control $\hat{c}_d := c_1^0 \equiv 1$.

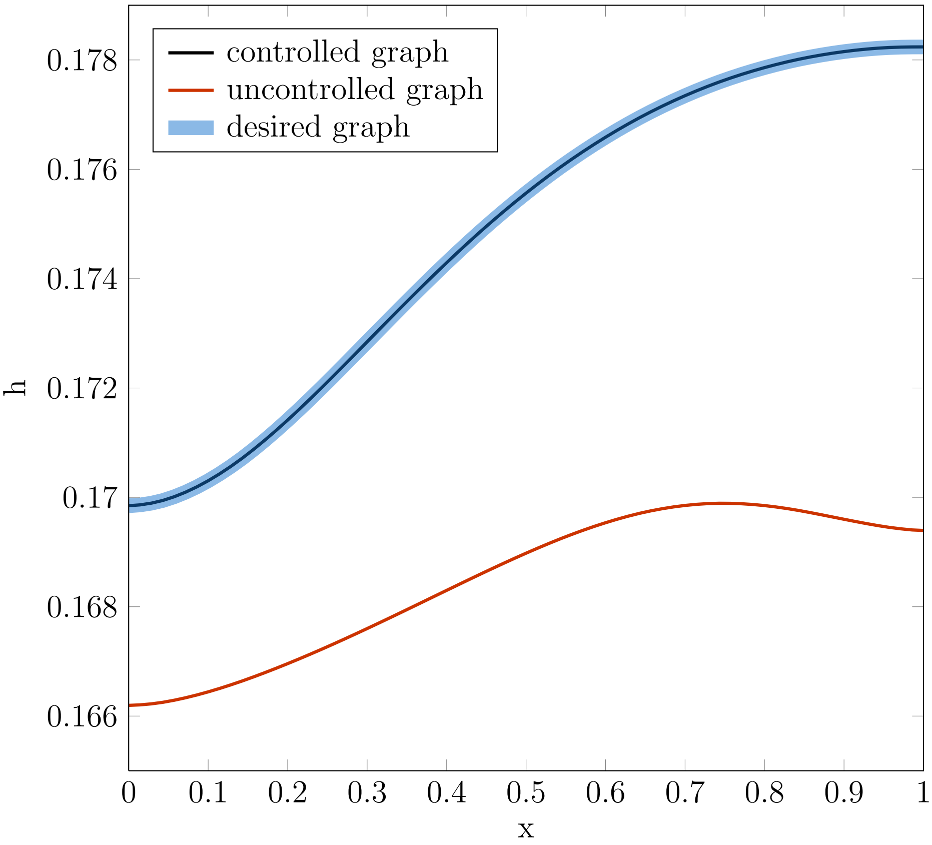


Figure 5.17.: Interface graphs with a reachable interface position $\hat{c}_d$.

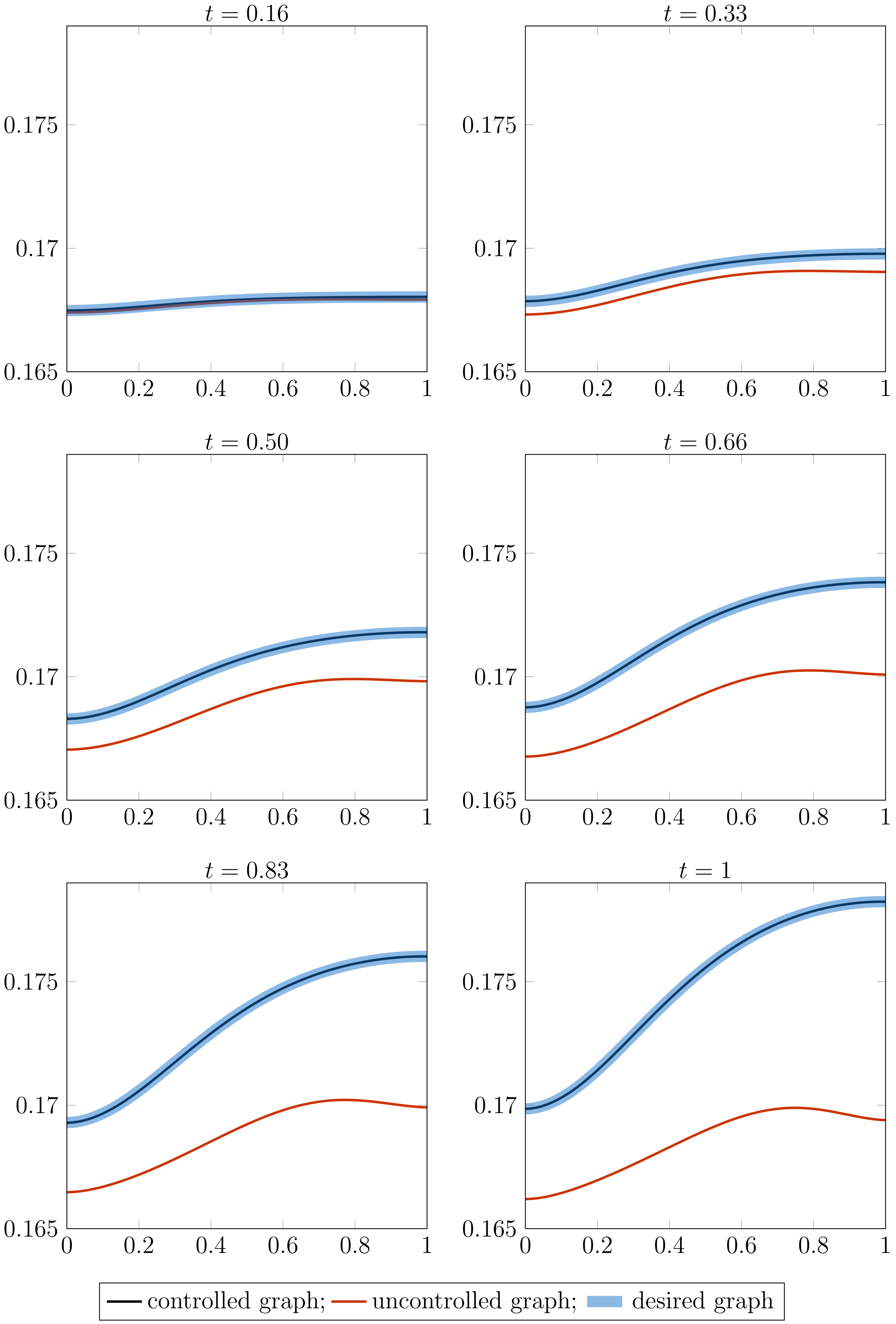


Figure 5.18.: Interface graphs with a reachable interface position $\hat{c}_d$ for various t.

With initial guess and weight parameters

$$\begin{aligned}&\lambda = 10^{-10},\ \Lambda = 10^{6},\ \bar{\Lambda} = 0,\\ &\hat{c}_1^0(t) \equiv 10,\end{aligned}$$

the desired interface can be approximated very accurate as figures 5.17, 5.18 and the resulting cost functional terms demonstrate

$$\begin{aligned}\mathfrak{d} &= 4.2946 \cdot 10^{-13},\\ \mathfrak{d}_{all} &= 2.2703 \cdot 10^{-11},\\ \mathfrak{p} &= 0.4723.\end{aligned}$$

There are two special attributes about this example. Firstly, the interface distance over the whole time interval $\mathfrak{d}_{all}$ (figure 5.20) is lower with the corresponding tracking term in the cost functional deactivated ($\bar{\Lambda} = 0$) than with those tracking terms activated ($\bar{\Lambda} > 0$). The distance terms with parameter sets

$$\begin{aligned}&\hat{c}_1 : \lambda = 10^{-10}, \Lambda = 10^{6}, \bar{\Lambda} = 0,\\ &\hat{c}_2 : \lambda = 10^{-10}, \Lambda = 10^{6}, \bar{\Lambda} = 10^{5},\\ &\hat{c}_3 : \lambda = 10^{-10}, \Lambda = 10^{5}, \bar{\Lambda} = 10^{4},\\ &\hat{c}_4 : \lambda = 10^{-4},\ \Lambda = 10^{6}, \bar{\Lambda} = 0,\\ &\hat{c}_5 : \lambda = 10^{-10}, \Lambda = 10^{2}, \bar{\Lambda} = 0,\end{aligned}$$

can be seen in figure 5.19 and 5.20.
Secondly, the lower control constraint, $p_{in} \geq 0$, is active for the approximated controls after the first iteration step of the gradient algorithm. This can be observed based on the kinks in the controls in figure 5.21. For the control $\hat{c}_5$ the constraint is active even after the last iteration step. It preserves the velocity from changing the flow direction.

The initial guesses

$$\begin{aligned}&\hat{c}_6^0(t) \equiv 5,\\ &\hat{c}_7^0(t) \equiv 2,\\ &\hat{c}_8^0(t) \equiv 0,\end{aligned}$$

are closer to the control $\hat{c}_d$, which produces the desired interface. They are expected to conclude in better convergence behavior. This holds for the control $\hat{c}_8$ (figure 5.22). It can reach almost the same accuracy as $\hat{c}_1$ with 6 iteration steps less. The interface distance at terminal time can be reduced further by $\hat{c}_6$ and furthermore by $\hat{c}_7$. But with 57 and 94 iteration steps, they converge slower than $\hat{c}_1$ and $\hat{c}_8$. For the various initial guesses, the same weight factors as for $\hat{c}_1$ are used.

$$\lambda = 10^{-10},\ \Lambda = 10^{6},\ \bar{\Lambda} = 0.$$

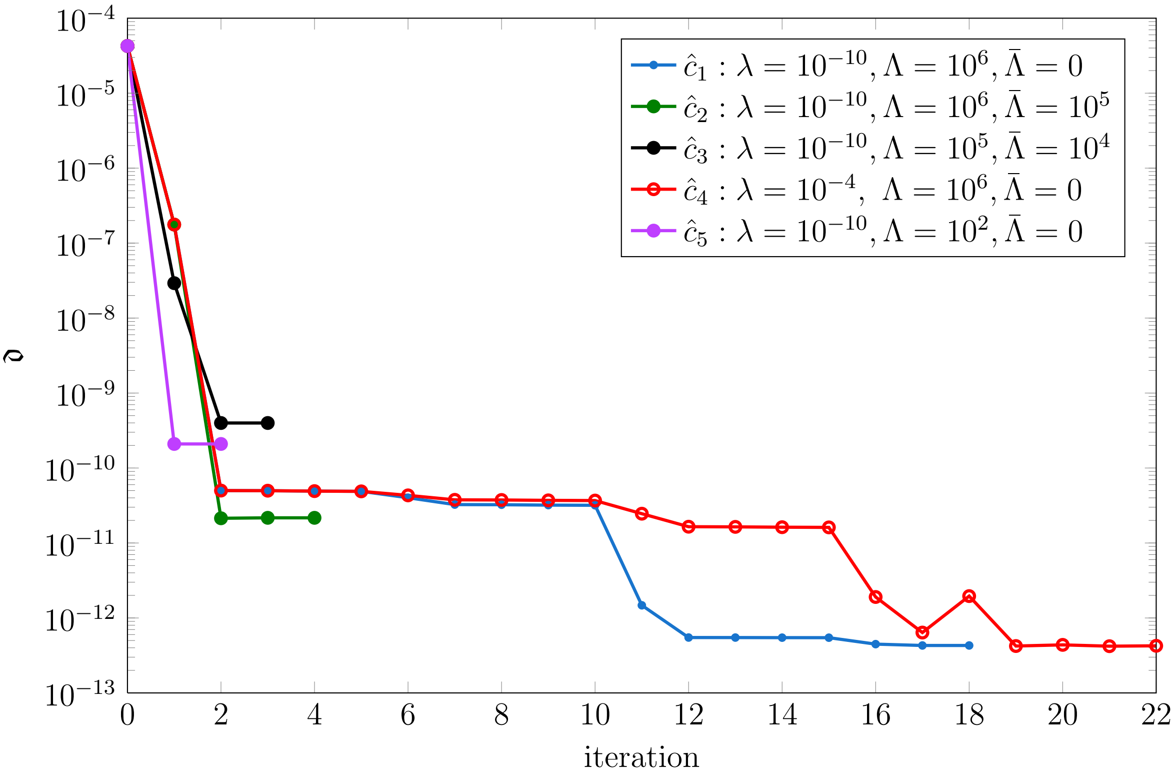


Figure 5.19.: Interface distance $\mathfrak{d}$ at $t = E$ for different parameter sets.

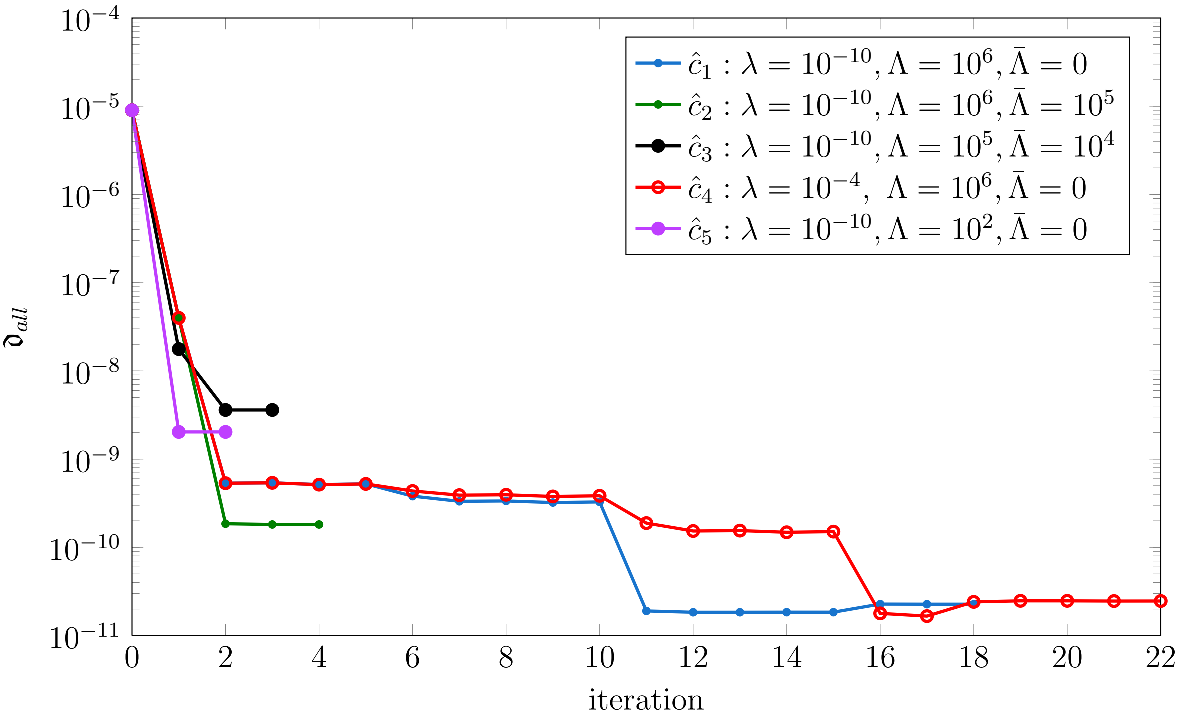


Figure 5.20.: Interface distance $\mathfrak{d}_{all}$ the whole time interval for different parameter sets.

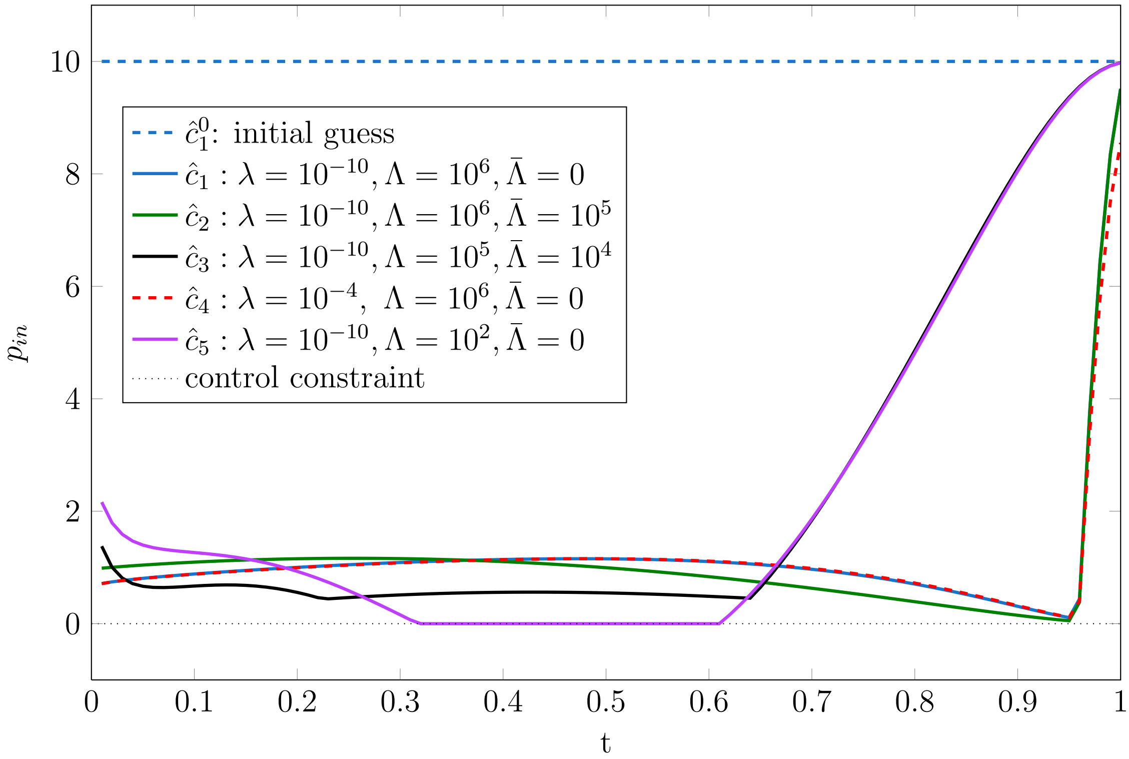


Figure 5.21.: Computed controls $\hat{c}_1$ to $\hat{c}_5$ with different parameter sets.

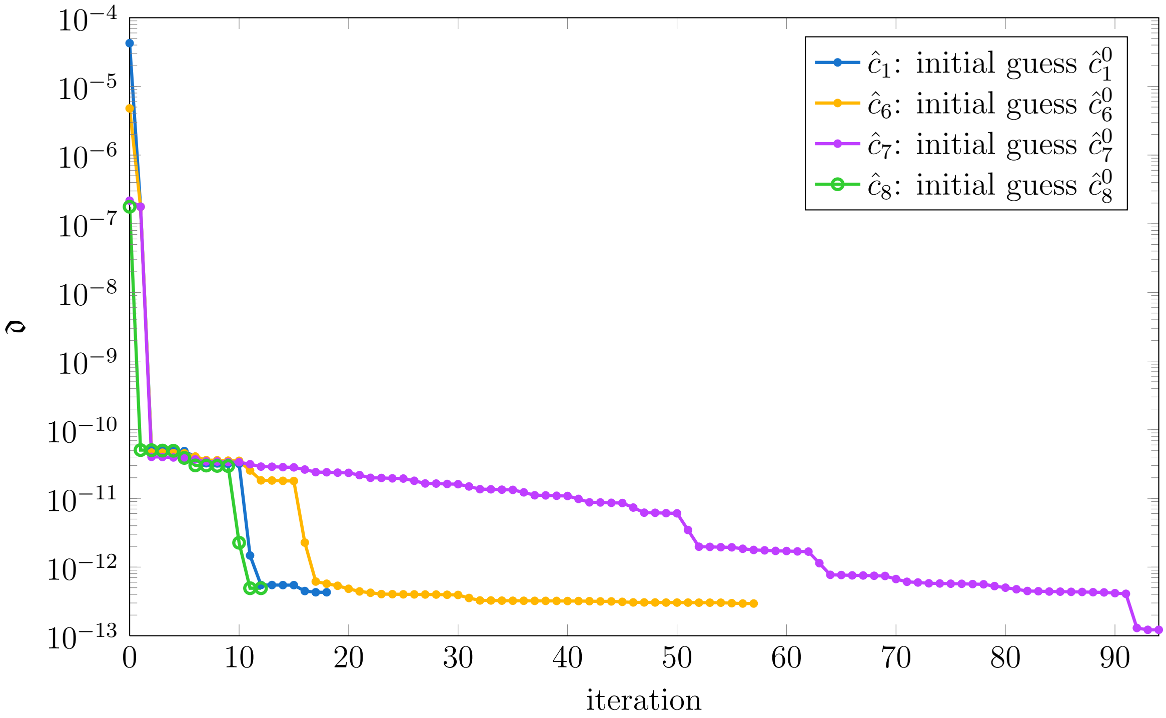


Figure 5.22.: Interface distance $\mathfrak{d}$ at $t = E$ for different controls $\hat{c}_1$, $\hat{c}_6$, $\hat{c}_7$, $\hat{c}_8$ with different initial guesses.

6. Conclusions

This thesis has introduced an approach for the optimal control of the interface position in a Stefan problem fully coupled with the Navier–Stokes equations. Compared to existing research, the new problem setting has increased in complexity. The mesh movement method used in this thesis is able to track the moving boundary and is fully included into the PDE systems.

To our extend of knowledge, this is the first attempt to combine mesh movement methods and finite elements for the optimal control of a two-dimensional two-phase Stefan problem. The control of the inflow pressure acts relatively indirect on the interface position, which makes the control of this non-linear problem a challenging task.

An important contribution of this thesis is the formulation of the adjoint system and, as a result of this, the first-order necessary optimality conditions using a formal Lagrange approach. Revealed by this, the gradient of the quadratic tracking-type cost functional can be utilized for a projected gradient algorithm. As illustrated with several numerical examples, this algorithm can approach the desired state accurately. A powerful quadratic line minimization algorithm is integrated into the gradient method. Moreover, the examples have demonstrated that such a method of steepest descent is limited to approximate stationary points and is heavily dependent on the choice of the cost functional. Weight factors that lead to good results in one setting can be an inappropriate selection for another setting and vice versa. Thus, a general purpose selection strategy is not close at hand.

Another major contribution of this thesis is the numerical implementation of the PDE systems, mesh movement techniques, and algorithms in Python with the usage of FEniCS and SciPy.

7. Perspectives

The approach proposed in this thesis showed off its potential in several numerical examples. Testing the algorithms for other settings would be interesting. Some more simple possibilities are to change the initial interface position to something else than a straight line or to change the shape of the domain. Extending the model to three spatial dimensions or to an m-phase Stefan problem should be realizable. Certainly, this would cause additional work, not only on the theoretical, but especially on the implementation side.

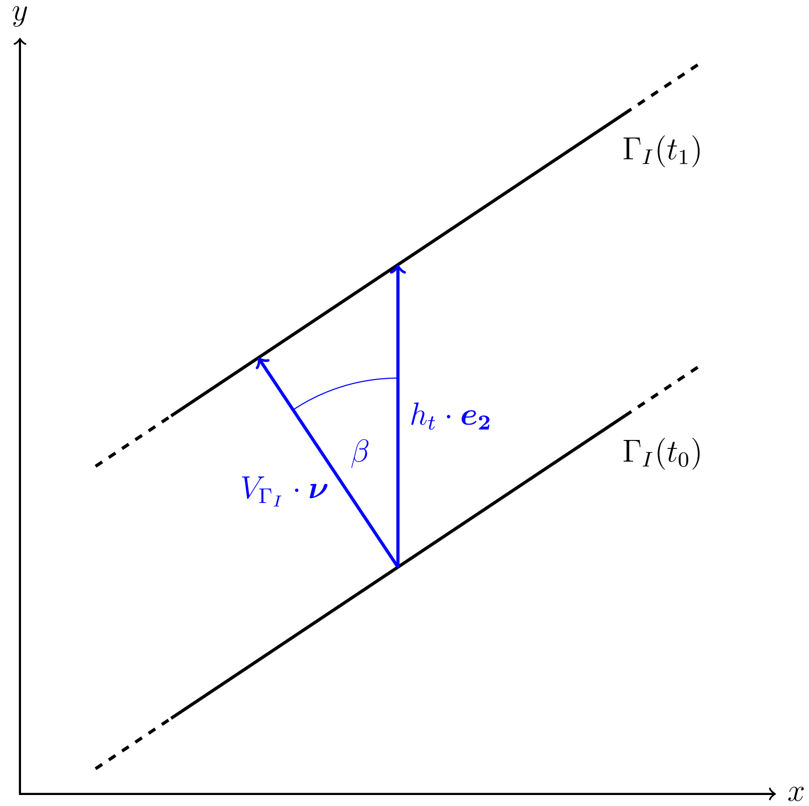
From the mathematical point of view, higher-order optimality conditions and consequently higher-order methods, are desirable. They might lead to faster convergence of the computationally intensive approximations. Neither the existence nor the uniqueness of solutions of the two-phase Stefan problem, as formulated in this thesis, are known. A first step would be the derivation of a rigorous functional analytical framework for the problem. Besides the quadratic cost functional used in this thesis, others, for example of L^1 or L^∞ type, might be of interest. Additionally, the curvature of the interface graph could be used as a measure of the flatness of the interface in the cost functional.

If the optimal control approach in this thesis is investigated satisfactorily, the next major step is to develop a closed loop optimal control system for the two-phase Stefan problem. For this boundary feedback stabilization approach, well-known linear quadratic regulator techniques might be applicable [4].

A. Appendix

A.1. Interface Normal Representation

The angle between the interface normal $\boldsymbol{\nu}$ and $\boldsymbol{e_2} = [0,1]^T$ may be named $\beta = \measuredangle(\boldsymbol{\nu}, \boldsymbol{e_2})$. For two points of time $t_0, t_1 \in [0, E], t_0 < t_1$ with $t_1 - t_0 = \tau \to 0$, the interfaces $\Gamma_I(t_0)$ and $\Gamma_I(t_1)$ are assumed to be parallel and locally a straight line. The two vectors $h_t \cdot \boldsymbol{e_2}$ and $V_{\Gamma_I} \cdot \boldsymbol{\nu}$ point on $\Gamma_I(t_1)$ if they have the same starting point on $\Gamma_I(t_0)$.


y
$\Gamma_I(t_1)$
$h_t \cdot \boldsymbol{e_2}$
β
$\Gamma_I(t_0)$
$V_{\Gamma_I} \cdot \boldsymbol{\nu}$
x

With the formula

$$\cos\beta = \frac{w_1 \cdot w_2}{||w_1|| \cdot ||w_2||},$$

for two vectors $w_1, w_2 \in \mathbb{R}^2$ this leads to

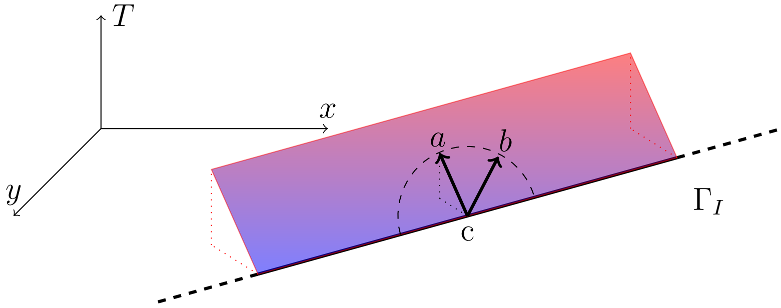
$$\cos\beta = \frac{V_{\Gamma_I}}{h_t} = \frac{\boldsymbol{\nu} \cdot h_t \cdot \boldsymbol{e_2}}{h_t},$$

and

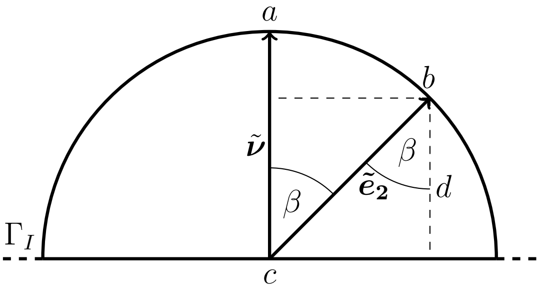
$$V_{\Gamma_I} = \begin{bmatrix} 0 \\ h_t \end{bmatrix} \cdot \boldsymbol{\nu} = \partial_t \begin{bmatrix} x \\ h \end{bmatrix} \cdot \boldsymbol{\nu}. \tag{A.1}$$

A.2. Stefan Condition Reformulation

As in section 2.1, $\beta = \measuredangle(\boldsymbol{\nu}, \boldsymbol{e_2})$. May $c \in \mathbb{R}^3$ be a point on the interface and the third dimension denote the temperature T. The interface normal $\tilde{\boldsymbol{\nu}} := \epsilon \cdot \boldsymbol{\nu}, \epsilon \to 0$ and $\tilde{\boldsymbol{e}}_{\boldsymbol{2}} := \epsilon \cdot \boldsymbol{e_2}$ are scaled. Thus, the temperature T is assumed to be locally linear. The points $a := c + [\tilde{\boldsymbol{\nu}}, \partial_{\boldsymbol{\nu}} T]^T$ and $b := c + [0, \epsilon, \partial_y T]^T$ are on the plane defined by the temperature close to the interface.



Imagine d as the distance between b and the interface.


a
b
$\tilde{\boldsymbol{\nu}}$
β
β
$\tilde{\boldsymbol{e}}_{\boldsymbol{2}}$
d
Γ_I
c

This leads to

$$\cos\beta = \frac{\tilde{\boldsymbol{\nu}} \cdot \tilde{\boldsymbol{e}}_{\boldsymbol{2}}}{||\tilde{\boldsymbol{\nu}}|| \cdot ||\tilde{\boldsymbol{e}}_{\boldsymbol{2}}||} = \frac{1}{\sqrt{1+h_x^2}} \cdot \begin{bmatrix} -h_x \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{1+h_x^2}},$$
$$\cos\beta = \frac{d}{||\boldsymbol{e_2}||} = d = \frac{1}{\sqrt{1+h_x^2}},$$

implying that

$$\partial_y T = \partial_{\boldsymbol{\nu}} T \cdot d = \frac{1}{\sqrt{1+h_x^2}} \cdot \partial_{\boldsymbol{\nu}} T. \tag{A.2}$$

The gradient of the temperature in vertical direction can be expressed in dependence of the gradient in normal direction

$$(\nabla T)_{s/l} \circ \Phi = \sqrt{1+h_x^2} (\partial_y T)_{s/l} \circ \Phi \qquad \text{on } \Gamma_C.$$

The Stefan condition (2.4b) can be reformulated to

$$\begin{aligned} &\sqrt{1+h_x^2} \cdot [k_s(\nabla T)_s - k_l(\nabla T)_l] \circ \Phi - L \cdot h_t \\ =&(1+h_x^2) \cdot [k_s(\partial_y T)_s - k_l(\partial_y T)_l] \circ \Phi - L \cdot h_t = 0. \end{aligned}$$

A.3. Results for the Example from Section 5.2

Table A.1.: Results for the example from section 5.2.

initial guess	λ	Λ	$\bar{\Lambda}$	$\mathfrak{d}$	$\mathfrak{d}_{all}$	$\mathfrak{p}$	**# iterations**
c_1^0	10^{-10}	10^2	0	$1.4317 \cdot 10^{-7}$	$7.0578 \cdot 10^{-7}$	15.3800	4
c_1^0	10^{-10}	10^5	10^4	$1.8107 \cdot 10^{-7}$	$6.7245 \cdot 10^{-7}$	15.0412	4
c_1^0	10^{-10}	10^5	10^3	$2.0051 \cdot 10^{-7}$	$7.2671 \cdot 10^{-7}$	14.7371	2
c_1^0	10^{-10}	10^4	10^5	$2.3960 \cdot 10^{-7}$	$4.5955 \cdot 10^{-7}$	15.0645	3
c_1^0	10^{-10}	10^5	0	$1.9919 \cdot 10^{-7}$	$7.2675 \cdot 10^{-7}$	14.7474	2
c_5^0	10^{-10}	10^2	0	$1.4710 \cdot 10^{-7}$	$1.7631 \cdot 10^{-6}$	19.9472	3
c_6^0	10^{-10}	10^2	0	$7.7936 \cdot 10^{-8}$	$6.0702 \cdot 10^{-7}$	19.4810	2

Initial guesses:

$$
\begin{aligned}
c_1^0(t) &\equiv 1, \\
c_5^0(t) &\equiv 12, \\
c_6^0(t) &= \begin{cases} 20, & t \in [0, 0.17], \\ 10, & t \in (0.17, 1]. \end{cases}
\end{aligned}
$$

Terms from the cost functional (3.10):

$$
\begin{aligned}
\mathfrak{d} &:= \int_{\Gamma_C} (h(E,x) - h_d(E,x))^2, \\
\mathfrak{d}_{all} &:= \int_0^E \int_{\Gamma_C} (h(t,x) - h_d(t,x))^2, \\
\mathfrak{p} &:= \int_0^E \int_{\Gamma_H} (p_{in}(t) I(x))^2.
\end{aligned}
$$

A.4. Results for the Example from Section 5.3

Table A.2.: Results for the example from section 5.3.

initial guess	λ	Λ	$\bar{\Lambda}$	$\mathfrak{d}$	$\mathfrak{d}_{all}$	$\mathfrak{p}$	**# iterations**
$\tilde{c}_1^0$	10^{-10}	10^5	0	$3.9422 \cdot 10^{-10}$	$4.5865 \cdot 10^{-8}$	15.6606	37
$\tilde{c}_1^0$	10^{-10}	10^6	0	$2.1424 \cdot 10^{-10}$	$3.4164 \cdot 10^{-8}$	15.7936	57
$\tilde{c}_1^0$	10^{-10}	10^2	0	$2.6145 \cdot 10^{-8}$	$9.9363 \cdot 10^{-8}$	14.1653	4
$\tilde{c}_1^0$	10^{-10}	10^4	10^5	$9.1034 \cdot 10^{-8}$	$3.7233 \cdot 10^{-8}$	13.8367	3
$\tilde{c}_1^0$	10^{-10}	10^5	10^4	$5.0140 \cdot 10^{-9}$	$1.1374 \cdot 10^{-8}$	15.3050	11
$\tilde{c}_6^0$	10^{-10}	10^1	0	$1.1778 \cdot 10^{-8}$	$7.8689 \cdot 10^{-7}$	19.2143	3
$\tilde{c}_6^0$	10^{-10}	10^5	0	$3.0897 \cdot 10^{-6}$	$4.7712 \cdot 10^{-7}$	22.5000	1
$\tilde{c}_7^0$	10^{-10}	10^5	0	$1.5294 \cdot 10^{-9}$	$9.7993 \cdot 10^{-8}$	18.5422	2
$\tilde{c}_1^0$	10^{-6}	10^5	0	$4.0616 \cdot 10^{-10}$	$4.6221 \cdot 10^{-8}$	15.6531	26
$\tilde{c}_1^0$	10^{-4}	10^5	0	$1.7234 \cdot 10^{-9}$	$4.6233 \cdot 10^{-8}$	15.2697	16
$\tilde{c}_1^0$	10^{-2}	10^5	0	$4.0210 \cdot 10^{-8}$	$9.8700 \cdot 10^{-8}$	13.9190	4

Initial guesses:

$$\begin{aligned}
&\tilde{c}_1^0(t) \equiv 1,\\
&\tilde{c}_6^0(t) \equiv 12,\\
&\tilde{c}_7^0(t) = \begin{cases} 20, & t \in [0, 0.17],\\ 10, & t \in (0.17, 1]. \end{cases}
\end{aligned}$$

A.5. Results for the Example from Section 5.4

Table A.3.: Results for the example from section 5.4.

initial guess	λ	Λ	$\bar{\Lambda}$	$\mathfrak{d}$	$\mathfrak{d}_{all}$	$\mathfrak{p}$	**# iterations**
$\hat{c}_1^0$	10^{-10}	10^6	0	$4.2946 \cdot 10^{-13}$	$2.2703 \cdot 10^{-11}$	0.4723	18
$\hat{c}_1^0$	10^{-10}	10^6	10^5	$2.1660 \cdot 10^{-11}$	$1.8172 \cdot 10^{-10}$	0.4588	4
$\hat{c}_1^0$	10^{-10}	10^5	10^4	$3.9863 \cdot 10^{-10}$	$3.6088 \cdot 10^{-9}$	2.3250	3
$\hat{c}_1^0$	10^{-4}	10^6	0	$4.2506 \cdot 10^{-13}$	$2.4663 \cdot 10^{-11}$	0.4082	22
$\hat{c}_1^0$	10^{-10}	10^6	0	$2.0937 \cdot 10^{-10}$	$2.0346 \cdot 10^{-9}$	2.0347	2
$\hat{c}_6^0$	10^{-10}	10^6	0	$2.9456 \cdot 10^{-13}$	$1.5957 \cdot 10^{-11}$	0.2999	57
$\hat{c}_7^0$	10^{-10}	10^6	0	$1.2191 \cdot 10^{-13}$	$1.2069 \cdot 10^{-11}$	0.1824	94
$\hat{c}_8^0$	10^{-10}	10^6	0	$4.9211 \cdot 10^{-13}$	$2.1927 \cdot 10^{-11}$	0.1341	12

Initial guesses:

$$\begin{aligned} \hat{c}_1^0(t) &\equiv 10, \\ \hat{c}_6^0(t) &\equiv 5, \\ \hat{c}_7^0(t) &\equiv 2, \\ \hat{c}_8^0(t) &\equiv 0. \end{aligned}$$

Bibliography

[1] M. S. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes, and G. N. Wells. The FEniCS Project Version 1.5. *Archive of Numerical Software*, 3(100), 2015. `http://fenicsproject.org/`.

[2] H. Antil, R. H. Nochetto, and P. Sodré. Optimal Control of a Free Boundary Problem: Analysis with Second-Order Sufficient Conditions. *SIAM Journal on Control and Optimization*, 52(5):2771–2799, 2014. `http://dx.doi.org/10.1137/120893306`.

[3] H. Antil, R. H. Nochetto, and P. Sodré. Optimal Control of a Free Boundary Problem with Surface Tension Effects: A Priori Error Analysis. *SIAM Journal on Numerical Analysis*, 53(5):2279–2306, 2015. `http://dx.doi.org/10.1137/140958360`.

[4] E. Bänsch, P. Benner, J. Saak, and H. K. Weichelt. Riccati-based boundary feedback stabilization of incompressible Navier–Stokes flows. *SIAMSciComp*, 37(2):A832–A858, 2015. `http://dx.doi.org/10.1137/140980016`.

[5] E. Bänsch, J. Paul, and A. Schmidt. An ALE FEM for solid-liquid phase transitions with free melt surface. *Berichte aus der Technomathematik 10-07*, 2010. `http://www.math.uni-bremen.de/zetem/cms/media.php/262/report1007.pdf`.

[6] E. Bänsch, J. Paul, and A. Schmidt. An ALE finite element method for a coupled Stefan problem and Navier-Stokes equations with free capillary surface. *International Journal for Numerical Methods in Fluids*, 71(10):1282–1296, 2013. `http://dx.doi.org/10.1002/fld.3711`.

[7] G. Beckett, J. A. Mackenzie, and M. L. Robertson. A Moving Mesh Finite Element Method for the Solution of Two-Dimensional Stefan Problems. *Journal of Computational Physics*, 168(2):500 – 518, 2001. `http://dx.doi.org/10.1006/jcph.2001.6721`.

[8] M. Bernauer. *Motion Planning for the Two-Phase Stefan Problem in Level Set Formulation.* PhD thesis, Chemnitz University of Technology, 2010. `http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-63654`.

[9] M. K. Bernauer and R. Herzog. Optimal Control of the Classical Two-Phase Stefan Problem in Level Set Formulation. *SIAM Journal on Scientific Computing*, 33(1):342–363, 2011. `http://dx.doi.org/10.1137/100783327`.

[10] M. O. Bristeau, R. Glowinski, and J. Periaux. Numerical methods for the Navier–Stokes equations. Applications to the simulation of compressible and incompressible viscous flows. *Computer Physics Reports*, 6(1):73 – 187, 1987. `http://dx.doi.org/10.1016/0167-7977(87)90011-6`.

[11] J. Donea, S. Giuliani, and J. P. Halleux. An arbitrary Lagrangian–Eulerian finite element method for transient dynamic fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 33(1):689 – 723, 1982. `http://dx.doi.org/10.1016/0045-7825(82)90128-1`.

[12] C. Eck, H. Garcke, and P. Knabner. *Mathematische Modellierung*. Springer, 2011. `http://dx.doi.org/10.1007/978-3-642-18424-6`.

[13] R. Glowinski. *Finite element methods for the numerical simulation of incompressible viscous flow. Introduction to the control of the Navier–Stokes equations.* Lectures in Applied Mathematics 28 (1991): 219-301, 1991.

[14] S. C. Gupta (Of Indian Institute of Science in Bangalore). *The classical Stefan problem : basic concepts, modelling and analysis.* Elsevier, 2003. `http://trove.nla.gov.au/work/9624319`.

[15] M. Heinkenschloss and F. Tröltzsch. Analysis of the Lagrange-SQP-Newton method for the control of a phase field equation. *Control and Cybernetics*, Vol. 28, no 2:177–211, 1999.

[16] M. Hinze. *Optimal and instantaneous control of the instationary Navier–Stokes equations.* Habilitationsschrift, 2002. `http://www.math.uni-hamburg.de/home/hinze/Psfiles/habil_mod.pdf`.

[17] M. Hinze and S. Ziegenbalg. Optimal control of the free boundary in a two-phase Stefan problem with flow driven by convection. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 87(6):430–448, 2007. `http://dx.doi.org/10.1002/zamm.200610326`.

[18] T. J. R. Hughes, W. K. Liu, and T. K. Zimmermann. Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 29(3):329 – 349, 1981. `http://dx.doi.org/10.1016/0045-7825(81)90049-9`.

[19] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python. `http://www.scipy.org/`, 2001–2013.

[20] G. Leugering, S. Engell, A. Griewank, M. Hinze, R. Rannacher, V. Schulz, M. Ulbrich, and S. Ulbrich. *Constrained Optimization and Optimal Control for Partial Differential Equations.* Birkhäuser, 2012. `http://dx.doi.org/10.1007/978-3-0348-0133-1`.

[21] A. Logg, K. Mardal, G. N. Wells, et al. *Automated Solution of Differential Equations by the Finite Element Method.* Springer, 2012. `http://dx.doi.org/10.1007/978-3-642-23099-8`.

[22] J. Marburger. Adjoint-Based Optimal Control of Time-Dependent Free Boundary Problems. *ArXiv e-prints*, 2012. `http://adsabs.harvard.edu/abs/2012arXiv1212.3789M`.

[23] A. Nadeshda, M. Y. Antonov, V. S. Borisov, et al. *Computational Technologies. Advanced Topics.* De Gruyter, 2014. `http://www.degruyter.com/view/product/428706`.

[24] P. Neittaanmäki. Optimal control for state constrained two-phase Stefan problems. *Birkhäuser*, 99:309–316, 1991. `http://dx.doi.org/10.1007/978-3-0348-5715-4_27`.

[25] M. Niezgodka, A. Crowley, and A. M. Meirmanov. *The Stefan Problem.* De Gruyter, 2011. `http://www.degruyter.com/view/product/172604`.

[26] R. H. Nochetto, M. Paolini, and C. Verdi. An Adaptive Finite Element Method for Two-Phase Stefan Problems in Two Space Dimensions. I: Stability and Error Estimates. *Mathematics of Computation*, 57(195):73–108, 1991. `http://doi.org/10.2307/2938664`.

[27] R. H. Nochetto, M. Paolini, and C. Verdi. An Adaptive Finite Element Method for Two-Phase Stefan Problems in Two Space Dimensions. II: Implementation and Numerical Experiments. *SIAM Journal on Scientific and Statistical Computing*, 12(5):1207–1244, 1991. `http://dx.doi.org/10.1137/0912065`.

[28] T. Roubíček. Optimal control of a Stefan problem with state-space constraints. *Numerische Mathematik*, 50(6):723–744. `http://dx.doi.org/10.1007/BF01398381`.

[29] T. Roubíček and C. Verdi. A stable approximation of a constrained optimal control for continuous casting. *Numerical Functional Analysis and Optimization*, 13(5-6):487–494, 1992. `http://dx.doi.org/10.1080/01630569208816494`.

[30] L. I. Rubinshteĭn. *The Stefan Problem.* Translations of mathematical monographs. American Mathematical Society, 1971.

[31] Stichting Mathematisch Centrum, Amsterdam, The Netherlands. Python. `http://www.python.org`, 1991–1995.

[32] C. Taylor and P. Hood. A numerical solution of the Navier–Stokes equations using the finite element technique. *Computers & Fluids*, 1(1):73 – 100, 1973. `http://dx.doi.org/10.1016/0045-7930(73)90027-3`.

[33] D. Tiba and M. Tiba. *Approximation for control problems with pointwise state constraints*, volume 91. Birkhäuser, 1989.

[34] F. Tröltzsch. *Optimale Steuerung partieller Differentialgleichungen: Theorie, Verfahren und Anwendungen.* Vieweg+Teubner, 2009. `http://link.springer.com/book/10.1007%2F978-3-8348-9357-4`.

[35] A. Visintin. *Models of Phase Transitions.* Birkhäuser, 1996. `http://dx.doi.org/10.1007/978-1-4612-4078-5`.

[36] R. E. White. A Numerical Solution of the Enthalpy Formulation of the Stefan Problem. *SIAM Journal on Numerical Analysis*, 19(6):1158–1172, 1982. `http://dx.doi.org/10.1137/0719083`.

[37] R. E. White. An Enthalpty Formulation of the Stefan Problem. *SIAM Journal on Numerical Analysis*, 19(6):1129–1157, 1982. `http://dx.doi.org/10.1137/0719082`.

[38] N. Zabaras, B. Ganapathysubramanian, and L. Tan. Modelling dendritic solidification with melt convection using the extended finite element method. *Journal of Computational Physics*, 218(1):200 – 227, 2006. `http://www.sciencedirect.com/science/article/pii/S0021999106000787`.

[39] N. Zabaras and D. Samanta. A stabilized volume-averaging finite element method for flow in porous media and binary alloy solidification processes. *International Journal for Numerical Methods in Engineering*, 60(6):1103–1138, 2004. `http://dx.doi.org/10.1002/nme.998`.

[40] S. Ziegenbalg. *Kontrolle freier Ränder bei der Erstarrung von Kristallschmelzen.* PhD thesis, Technische Universität Dresden, 2008. `http://nbn-resolving.de/urn:nbn:de:bsz:14-ds-1212521184972-55836`.

Statutory Declarations

I declare that I have developed and written the enclosed Master Thesis completely by myself, and have not used sources or means without declaration in the text. Any thoughts from others or literal quotations are clearly marked. The Master Thesis was not used in the same or in a similar version to achieve an academic grading or is being published elsewhere.

Location, Date, Signature