Inaugural-Dissertation

zur
Erlangung der Doktorwürde
der
Naturwissenschaftlich-Mathematischen Gesamtfakultät
der
Ruprecht-Karls-Universität
Heidelberg

vorgelegt von
Diplom-Mathematiker Dominik Meidner
aus Heidelberg

Tag der mündlichen Prüfung: 3. März 2008

Adaptive Space-Time Finite Element Methods for Optimization Problems Governed by Nonlinear Parabolic Systems

Gutachter: Prof. Dr. Rolf Rannacher
Prof. Dr. Dr. h. c. Hans Georg Bock

Abstract

Subject of this work is the development of concepts for the efficient numerical solution of optimization problems governed by parabolic partial differential equations. Optimization problems of this type arise for instance from the optimal control of physical processes and from the identification of unknown parameters in mathematical models describing such processes. For their numerical treatment, these generically infinite-dimensional optimal control and parameter estimation problems have to be discretized by finite-dimensional approximations. This discretization process causes errors which have to be taken into account to obtain reliable numerical results.

Focal point of the thesis at hand is the assessment of these discretization errors by a priori and especially a posteriori error analyses. Thereby, we consider Galerkin finite element discretizations of the state and the control variable in space and time. For the a priori analysis, we concentrate on the case of linear-quadratic optimal control problems. In this configuration, we prove error estimates of optimal order with respect to all involved discretization parameters. The a posteriori error estimation techniques are developed for a general class of nonlinear optimization problems. They provide separated and evaluable estimates for the errors caused by the different parts of the discretization and yield refinement indicators, which can be used for the automatic choice of suitable discrete spaces. The usage of adaptive refinement techniques within a strategy for balancing the several error contributions leads to efficient discretizations for the continuous problems.

The presented results and developed concepts are substantiated by various numerical examples including large scale optimization problems motivated by concrete applications from engineering and chemistry.

Zusammenfassung

Gegenstand dieser Arbeit ist die Entwicklung von Konzepten für das effiziente numerische Lösen von Optimierungsproblemen mit Beschränkungen durch parabolische partielle Differentialgleichungen. Probleme dieser Art entstehen beispielsweise bei der optimalen Steuerung physikalischer Prozesse sowie bei der Identifizierung unbekannter Parameter in mathematischen Modellen zur Beschreibung solcher Prozesse. Für ihre numerische Behandlung ist es notwendig, diese generisch unendlich-dimensionalen Probleme der optimalen Steuerung und Parameterschätzung mittels endlich-dimensionaler Approximationen zu diskretisieren. Dieser Diskretisierungsprozess verursacht Fehler, die berücksichtigt werden müssen, um verlässliche numerische Ergebnisse zu erhalten.

Schwerpunkt der vorliegenden Dissertation ist die Abschätzung dieser Diskretisierungsfehler mit Hilfe von a priori und insbesondere a posteriori Fehleranalysen. Dabei betrachten wir Finite-Elemente-Diskretisierungen der Zustands- und Kontrollvariablen in Ort und Zeit. Bei der a priori Analyse konzentrieren wir uns auf den Fall linear-quadratischer Optimalsteuerungsprobleme. Hierfür zeigen wir Fehlerabschätzungen von optimaler Ordnung bezüglich aller beteiligten Diskretisierungsparameter. Die Techniken zur a posteriori Fehlerschätzung werden für eine allgemeine Klasse nichtlinearer Optimierungsprobleme entwickelt. Sie liefern separierte und auswertbare Schätzungen der durch die verschiedenen Teile der Diskretisierung verursachten Fehler und stellen Verfeinerungsindikatoren für die automatische Wahl der geeigneten diskreten Räume bereit. Die Verwendung von adaptiven Verfeinerungstechniken innerhalb von Strategien zur Balancierung der einzelnen Fehlerbeiträge führt zu effizienten Diskretisierungen der kontinuierlichen Probleme.

Die präsentierten Ergebnisse und entwickelten Konzepte werden durch verschiedene numerische Tests bestätigt. Im Rahmen dieser Tests werden auch Optimierungsprobleme betrachtet, die durch konkrete Anwendungen aus den Ingenieurwissenschaften und der Chemie motiviert sind.

Contents

1 Introduction

This work is devoted to the development of efficient discretization techniques for the numerical solution of optimization problems governed by parabolic partial differential equations (PDEs). The two main topics covered by this thesis are the *a priori* and *a posteriori* error analysis for Galerkin space-time finite element discretizations of such optimization problems. Thereby, the a priori analysis investigates the convergence properties of the proposed discretizations and proves the asymptotic dependence of the discretization error on the discretization parameters. In contrast, the developed a posteriori error estimation techniques provide access to the capabilities of adaptive refinement of all involved types of discretizations leading to algorithms for the efficient numerical solution of the considered problems.

In particular, we investigate the numerical solution of constrained optimization problems where the constraint is given by means of a parabolic PDE. From an abstract point of view, we consider the minimization of a *cost functional* depending on the *state* u and the *control* q, subject to a possibly nonlinear state equation

$$\begin{aligned} \partial_t u + A(q,u) &= f, \\ u(0) &= u_0(q), \end{aligned}$$

describing a mathematical model for the concrete physical process in mind. Here, both the differential operator A and the initial condition u_0 may depend on the control q. This allows a simultaneous treatment of *optimal control* and *parameter identification* problems.

In the class of optimal control problems, the control q is employed to drive the considered process into a desired state or to keep the process running within a region with certain desired properties. Here, the operator A is typically given as

$$A(q,u) = C(u) + B(q)$$

with a (nonlinear) differential operator C and a usually linear control operator B. In parameter identification problems, the variable q denotes unknown parameters. Here, one is interested in recovering these parameters from observations which can be incorporated in the cost functional by the least-squares approach.

Both optimal control and parameter identification problems are generally infinite-dimensional optimization problems. For their numerical treatment, it is unavoidable to consider finite-dimensional approximations of these problems. In the considered context of time-dependent optimization problems, the finite-dimensional problems are constructed by discretization of the state and the control variables in time and space. All steps of discretization involved in this process induce errors. Hence, we observe a discretization error between the solution (q,u) of the continuous optimization problem and the solution (q_σ, u_σ) of its finite-dimensional approximation. The assessment of this error by a priori and a posteriori error analysis is the main objective in this thesis.

The a priori analysis is derived for linear-quadratic optimal control problems. We prove asymptotic convergence of the discretization error with respect to the different discretization parameters for the time and space discretization of the state and the control variables. These estimates rely on the regularity of the continuous optimal solution (q, u) which is itself determined by the regularity of the data, by the smoothness of the computational domain, and possibly by compatibility conditions between the initial condition, the right-hand side, and the boundary conditions. In contrast, the concept of a posteriori error estimation provides techniques for the automatic choice of suitable discretizations leading to efficient approximation algorithms. Thereby, all the necessary information is obtained from the computed discrete optimal solution (q_σ, u_σ) and no a priori information on the optimal solution (q, u) of the continuous problem is needed.

Since, depending on the size of the finite-dimensional approximations, the consumption of computing time for solving time-dependent optimization problems is comparatively high, efficient adaptive refinement techniques viewed as model reduction approach are crucial for the solution of such problems. The computations are quite expensive because of two reasons: The computational costs for the simulation of nonstationary PDEs are already high, since in every step of an (implicit) time stepping scheme a stationary PDE has to be solved. Additionally, the costs for the optimization of a process usually exceed the costs for the simulation. Our approach to cope with these difficulties is based on a posteriori error estimation which separately assesses the discretization errors caused by all parts of Galerkin discretizations used to carry the infinite-dimensional optimization problem to a finite-dimensional level. Thereby, the discretization error is measured with respect to a given quantity of interest. For optimal control problems, this quantity often coincides with the cost functional. However, in the case of parameter identification problems, the cost functional acts only as an instrument for identifying the unknown parameters and does not have any physical meaning. This motivates the consideration of error estimation with respect to a quantity of interest given as a further functional depending on the state and the control.

In what follows, we summarize the contents of the remaining chapters of the thesis at hand:

Theoretical Results

In Chapter 2, we introduce necessary notations and provide the precise formulation of the considered abstract optimization problem in a suitable functional analytic setting. Furthermore, standard techniques for proving existence and uniqueness of optimal solutions are sketched and first and second order optimality conditions are derived. We close this chapter by discussing different approaches for calculating first and second derivatives of the reduced cost functional required for applying derivative-based optimization algorithms to PDE-constrained optimization.

Space-Time Finite Element Discretization

Chapter 3 is devoted to the discretization of the considered nonstationary optimization problems. To this end, we employ Galerkin finite element methods separately in space and time to discretize the state variable. The control variable is discretized by a Galerkin approach, too. This allows us to give computable representations of the discrete gradient and Hessian like done in Chapter 2 for the continuous problem. The use of exact discrete derivatives is important for the convergence of optimization algorithms. Galerkin-type discretizations offer a natural way of deriving the discrete adjoint formulations, since discretization by means of the Galerkin approach exhibits the property that discretization and optimization interchange. That is, the *discretize-then-optimize* approach equals the *optimize-then-discretize* approach in this context. Furthermore, the a priori error analysis presented in Chapter 5 as well as our systematic approach to a posteriori error estimation presented in Chapter 6 rely on the usage of the proposed Galerkin discretizations.

We close this chapter by presenting some numerical tests confirming the correctness of the discrete derivatives computed via the proposed concepts.

Algorithmic Aspects of Numerical Optimization

In Chapter 4, we address algorithmic aspects of numerical methods for solving the prototypical PDE-constrained optimization problems considered in this thesis. We describe two abstract variants of Newton-based optimization loops which are concretized afterwards in view of different linear solvers and globalization techniques. We discuss possible globalization techniques such as line search and trust-region methods as well as the aspects of efficiently solving the linear systems arising in Newton methods. Thereby, we focus especially on matrix-free algorithms, since assembling the entire Hessian is prohibitive in large scale optimization.

Before substantiating our approach by numerical tests, we analyze storage reduction techniques which provide the possibility of reducing the amount of memory required for executing the proposed optimizations algorithms. This so-called *checkpointing* approach reduces the storage requirements during the computations of adjoint solutions by recomputing necessary solution samples of the state equation.

A Priori Error Analysis

In Chapter 5, we develop an a priori error analysis for Galerkin finite element discretizations in the case of a linear-quadratic optimal control problem. We provide error estimates of optimal order with respect to all involved discretization parameters for the discretization of the state space by discontinuous Galerkin methods in time and conforming continuous Galerkin methods in space combined with different types of Galerkin discretizations for the control variable.

Moreover, in the derived estimates, the influences of the different types of discretizations and also the influences of the temporal and spatial regularity properties of the optimal solution are separated. For the lowest degrees of space and time discretizations, a result similar to the one developed here can be found in the literature; see the introduction of Chapter 5 for detailed references. Besides the fact that our result also holds for higher order discretizations, we also do not need to impose conditions on the ratio of the temporal and spatial discretization parameters; they can be chosen independently of each other.

Apart from the a priori estimate for the error in the control variable, we also present convergence results for the optimal state and the corresponding adjoint state. Additionally, an estimate for the convergence of the error in terms of the optimal value of the cost functional is given.

To confirm the proved orders of convergence, we present numerical results for a configuration with known analytical optimal solution.

A Posteriori Error Estimation and Adaptivity

The second focal point of this thesis is the derivation of a posteriori error estimates for space-time finite element discretizations of parabolic optimization problems. In Chapter 6, we provide error estimates that assess the discretization error with respect to a given quantity of interest and separate the influences of different parts of the discretization (time and space discretization of the state and discretization of the control) on this error. Thereby, the considered quantity of interest may coincide with the cost functional of the optimization problem or may express another goal for the computation.

The developed error estimation techniques rely on concepts for a posteriori error estimation for optimization problems with elliptic constraints from Becker and Kapp [6] and Becker, Kapp, and Rannacher [7] for the cost functional, and from Becker and Vexler [11, 12] for a different quantity of interest. These approaches are extended to the case of optimization problems governed by parabolic equations to establish efficient adaptive algorithms which successively improve the accuracy of the computed solutions by constructing locally refined meshes for the time and space discretizations.

Furthermore, an equilibration strategy is used to balance the different discretization errors by deciding when to refine which of the involved discretizations. This procedure is crucial for the efficiency of a space-time adaptive algorithm. It strongly depends on the availability of reliable quantitative estimates for the separated discretization errors. In contrast to the error estimates derived in this thesis, heuristic error indicators based for instance on smoothness properties of the optimal solution do usually not meet these requirements. Also error estimators involving interpolation or stability constants can not be employed for equilibration, since the values of these constants are unknown for the concrete configuration of the optimization problem.

We conclude with the discussion of two numerical examples showing the capabilities of the proposed techniques and demonstrating their advantages compared to a more heuristic based mesh refinement.

Applications

In Chapter 7, we apply the developed a posteriori error analysis and the adaptive refinement techniques to two optimization problems taken from the literature motivated by concrete applications from engineering and chemistry.

As a first example, we consider the optimal control of a laser-induced hardening process of a workpiece made of steel. Thereby, the goal of the optimization is to adjust the intensity of the laser beam in such a way, that the thickness of the hardened part of the workpiece is close to a desired hardening profile.

In the second example, we consider a model for describing freely propagating laminar flames through a channel and their response to a cooled obstacle. The modeling of this process is done using a one-species reaction mechanism governed by an Arrhenius law. Our aim here is to estimate an unknown parameter in this Arrhenius term. This is typical for situations where the error in terms of the cost functional is of minor interest. Therefore, we assess here the error directly in terms of the unknown parameter to be identified.

Conclusions and Perspectives

In the concluding last chapter, we summarize the results presented in the thesis at hand and discuss some ideas on possible extensions and future work.

2 Theoretical Results

In this chapter, we state the precise formulation of the optimization problems to be considered and discuss some of their theoretical aspects.

In Section 2.1, we give some basic notations used throughout this thesis. Then, we continue in Section 2.2 by formulating the optimization problem we deal with in an abstract functional analytic manner. In Section 2.3, we sketch techniques for proving existence and uniqueness of optimal solutions. After stating first and second order optimality conditions in Section 2.4, we close this chapter with Section 2.5 by discussing different approaches to calculate first and second derivatives of the reduced cost functional, necessary to apply derivative-based optimization algorithms to PDE-constrained optimization problems.

2.1 Basic notations

Throughout this thesis, $\varOmega$ denotes a bounded domain in $\mathbb{R}^n$, $n \in \{2, 3\}$, with Lipschitz boundary $\partial\varOmega$; see Grisvard [42] for the precise definition. Furthermore, we denote by $I := (0, T)$ a bounded time interval with $0 < T < \infty$.

We adopt the standard notations for Lebesgue spaces $L^p(D)$ and Sobolev spaces $W^{m,p}(D)$ with $1 \leq p \leq \infty$, $m \in \mathbb{N}$, and $D \subseteq \varOmega$, $D \subseteq \partial\varOmega$, or $D \subseteq I$. Moreover, we use Lebesgue and Sobolev spaces of mappings with values in a Banach space Z. These spaces are denoted by $L^p(D, Z)$ and $W^{m,p}(D, Z)$. The standard spaces fit into this notation via the choice $Z = \mathbb{R}$. A detailed derivation of these spaces by means of the concepts of the Bochner integral can be found for instance in Dautray and Lions [25] and Wloka [88]. For $p = 2$, we denote the spaces $W^{m,2}(D, Z)$ as usual by $H^m(D, Z)$.

Table 2.1. Connection between the notation of the variables in numerical analysis and optimal control theory

Variable	Numerical analysis	Optimal control theory
Control	q	u
State	u	y
Adjoint state	z	p

In contrast to the notation used in publications from optimization theory (cf. for example Lions [53] or Tröltzsch [78]), we employ here the notation used in the numerical analysis community. That is, we denote the control by q, the state by u, and the adjoint state by z. The correspondence to the other notation is summarized in Table 2.1.

2.2 Abstract optimization problem

The optimization problems considered in this thesis are formulated in the following abstract setting: Let V and H be Hilbert spaces with

$$V \overset{\mathrm{d}}{\hookrightarrow} H,$$

where the injection of V into H is continuous and dense. H is identified with its dual space H^*. With V^*, the dual space of V, we have the Gelfand triple

$$V \overset{\mathrm{d}}{\hookrightarrow} H \cong H^* \overset{\mathrm{d}}{\hookrightarrow} V^*. \tag{2.1}$$

In the triple (2.1), V is densely embedded in H and H^* is densely embedded in V^*. Additionally, the corresponding injections are continuous. The duality pairing between the Hilbert space V and its dual V^* is denoted by $\langle \cdot, \cdot \rangle_{V^* \times V}$.

Remark 2.1. Let $i\colon V \to H$ be the injection of V into H. Then, its dual $i^*\colon H^* \to V^*$ is the injection of H^* into V^*. Because of the definition of i^*, every element $h \in H \cong H^*$ can be understood as linear continuous functional on V in virtue of the identity

$$\langle i^*(h), v \rangle_{V^* \times V} = (h, i(v))_H \quad \forall v \in V,$$

where $(\cdot, \cdot)_H$ is the inner product of H. Since H^* is densely embedded in V^*, every functional $\langle v^*, \cdot \rangle_{V^* \times V}$ can be uniformly approximated by inner products $(h, i(\cdot))_H$. That is, we can regard the continuous continuation of $(\cdot, \cdot)_H$ onto $V^* \times V$ as new representation formula for functionals in V^*. A more detailed derivation of this concept can be found for example in Gajewski, Gröger, and Zacharias [38], Lions [53], and Wloka [88].

We now tend to give the precise definition of the abstract optimization problem constrained by a parabolic PDE. We consider on the time interval I the abstract parabolic equation

$$\begin{aligned} \partial_t u(t) + A(q(t), u(t)) &= f(t) \qquad\qquad \text{for almost all } t \in I, \\ u(0) &= u_0(q(0)). \end{aligned} \tag{2.2}$$

Here and in the sequel, $q(t)$ from a spatial Hilbert space R denotes the control and $u(t) \in V$ denotes the state. The right-hand side is given by $f(t) \in V^*$ and the initial condition is modeled via $u_0\colon R \to H$.

In this abstract setting, we assume $A\colon R \times V \to V^*$ to be a spatial differential operator which is elliptic with respect to V and is given in weak form by the semilinear form $\bar{a}\colon R \times V \times V \to \mathbb{R}$ as

$$\langle A(\bar{q}, \bar{u}), \bar{v} \rangle_{V^* \times V} = \bar{a}(\bar{q}, \bar{u})(\bar{v}) \quad \forall \bar{u}, \bar{v} \in V,\ \forall \bar{q} \in R.$$

Remark 2.2. Here, both the differential operator A and the initial condition u_0 may depend on the control q. This allows a simultaneous treatment of both optimal control and parameter identification problems. For optimal control problems, the operator A is typically given on $R \times V$ by

$$A(\bar{q}, \bar{u}) = C(\bar{u}) - B(\bar{q}),$$

with a possibly nonlinear operator $C\colon V \to V^*$ and a usually linear control operator $B\colon R \to V^*$. In parameter identification problems, the variable q denotes the unknown parameters to be determined and may enter the operator A in a nonlinear way. The case of initial control is included via the q-dependent initial condition u_0. Even if formulation (2.2) allows the control q to enter via the differential operator A and the initial condition u_0 at the same time, we assume for simplicity throughout this thesis that either A or u_0 depends on q.

To define the weak formulation of problem (2.2), we introduce the Hilbert space for the states $X := W(I)$ defined as

$$W(I) = \left\{ v \;\middle|\; v \in L^2(I,V) \text{ and } \partial_t v \in L^2(I,V^*) \right\}.$$

It is well known that the space X is continuously embedded in $C(\bar{I},H)$, see for example Dautray and Lions [25].

To cover most of the possible concrete applications, the Hilbert space of the controls Q is chosen as a subspace of $L^2(I,R)$, that is

$$Q \subseteq L^2(I,R).$$

The inner product and the norm on Q are denoted by $(\cdot,\cdot)_Q$ and $\|\cdot\|_Q$, respectively.

Remark 2.3. This definition of the control space Q is motivated by the case when the control enters via the semilinear form. But also the case of q-dependent initial condition is covered by this choice by defining Q as

$$Q = \mathcal{P}_0(\bar{I},R) \subseteq L^2(I,R),$$

where $\mathcal{P}_0(\bar{I},R)$ denotes the space of constant polynomials defined on $\bar{I}$ with values in R.

After these preliminaries, we pose the *state equation* in a weak form: Find for given control $q \in Q$ a state $u \in X$ such that

$$\begin{aligned}\int_I (\partial_t u(t),\varphi(t))_H \,dt + \int_I \bar{a}(q(t),u(t))(\varphi(t))\,dt = \int_I (f(t),\varphi(t))_H \,dt \quad \forall \varphi \in X,\\ u(0) = u_0(q(0)),\end{aligned}$$

where $f \in L^2(I,V^*)$ represents the right-hand side of the state equation and $u_0\colon R \to H$ denotes a mapping describing control-dependent initial conditions. Note, that the inner products $(\partial_t u(t),\varphi(t))_H$ and $(f(t),\varphi(t))_H$ have to be understood accordingly to Remark 2.1. The solvability of this equation is discussed at the beginning of the following section.

For simplicity of notation, we skip the index H at the inner product $(\cdot,\cdot) := (\cdot,\cdot)_H$ and rewrite the state equation by means of the definitions

$$(v,w)_I := \int_I (v(t),w(t))\,dt \quad \text{and} \quad a(q,u)(\varphi) := \int_I \bar{a}(q(t),u(t))(\varphi(t))\,dt$$

in the more compact representation

$$(\partial_t u,\varphi)_I + a(q,u)(\varphi) + (u(0),\varphi(0)) = (f,\varphi)_I + (u_0(q),\varphi(0)) \quad \forall \varphi \in X, \tag{2.3}$$

where the initial condition is coupled to the state equation by $\varphi(0)$. This choice is motivated by the definition of the Lagrangian given in Section 2.5; cf. also Remark 2.9. Additionally, we skipped for brevity and also due to Remark 2.3 the argument "0" in the formulation of the initial condition.

The *objective* or *cost functional* $J\colon Q \times X \to \mathbb{R}$ for stating the optimization problem is defined using two functionals $J_1\colon V \to \mathbb{R}$ and $J_2\colon H \to \mathbb{R}$ by

$$J(q,u) := \int_I J_1(u(t))\,dt + J_2(u(T)) + \frac{\alpha}{2}\|q-\hat{q}\|_Q^2, \tag{2.4}$$

where a regularization (or cost) term of Tikhonov type is added, which involves a regularization parameter $\alpha \geq 0$ and a reference control $\hat{q} \in Q$.

The corresponding *parabolic optimization problem* is formulated as follows:

$$\text{Minimize } J(q,u) \text{ subject to (2.3)}, \; (q,u) \in Q \times X. \tag{$\mathbb{P}$}$$

Now, we present four examples of linear-quadratic and also nonlinear parabolic optimization problems fitting in the derived abstract framework:

Example 2.1 (Distributed control)**.** We consider for a given desired solution profile $\hat{u} \in L^2(I, L^2(\Omega))$ the control problem

$$\text{Minimize } J(q,u) = \frac{1}{2}\int_I \|u(t)-\hat{u}(t)\|^2_{L^2(\Omega)}\,dt + \frac{\alpha}{2}\int_I \|q(t)-\hat{q}(t)\|^2_{L^2(\Omega)}\,dt \tag{2.5a}$$

subject to the linear heat equation with $\varepsilon > 0$

$$\begin{aligned} \partial_t u - \varepsilon\Delta u &= q \qquad &&\text{in } \Omega \times I,\\ u &= 0 &&\text{on } \partial\Omega \times I,\\ u &= 0 &&\text{on } \Omega \times \{\,0\,\}. \end{aligned} \tag{2.5b}$$

To embed this example in the abstract setting, we choose the spaces

$$H = L^2(\Omega), \quad V = H^1_0(\Omega), \quad R = L^2(\Omega), \quad \text{and} \quad Q = L^2(I,R) = L^2(I,L^2(\Omega)),$$

define the functionals

$$J_1(u(t)) = \frac{1}{2}\|u(t)-\hat{u}(t)\|^2_{L^2(\Omega)} \quad \text{and} \quad J_2(u(T)) = 0,$$

the semilinear form

$$a(q,u)(\varphi) = \int_I\int_\Omega \varepsilon\nabla u(x,t)\nabla\varphi(x,t)\,dx\,dt - \int_I\int_\Omega q(x,t)\varphi(x,t)\,dx\,dt,$$

and the right-hand side and the initial condition

$$f = 0 \quad \text{and} \quad u_0(q) = 0.$$

Example 2.2 (Neumann/Robin boundary control)**.** We consider for a given desired state $\hat{u} \in L^2(\Omega)$ the control problem

$$\text{Minimize } J(q,u) = \frac{1}{2}\|u(T) - \hat{u}\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\int_I \|q(t) - \hat{q}(t)\|^2_{L^2(\partial\Omega)}\,dt$$

subject to the linear heat equation with Stefan Boltzmann boundary condition and $\beta, \varepsilon > 0$

$$\begin{aligned} \partial_t u - \varepsilon\Delta u &= 0 && \text{in } \Omega \times I, \\ \varepsilon\partial_n u &= \beta(q^4 - u^4) && \text{on } \partial\Omega \times I, \\ u &= 0 && \text{on } \Omega \times \{\,0\,\}. \end{aligned}$$

To embed this example in the abstract setting, we choose the spaces

$$H = L^2(\Omega), \quad V = H^1(\Omega), \quad R = L^2(\partial\Omega), \quad \text{and} \quad Q = L^2(I,R) = L^2(I, L^2(\partial\Omega)),$$

define the functionals

$$J_1(u(t)) = 0 \quad \text{and} \quad J_2(u(T)) = \frac{1}{2}\|u(T) - \hat{u}\|^2_{L^2(\Omega)},$$

the semilinear form

$$a(q,u)(\varphi) = \int_I\int_\Omega \varepsilon\nabla u(x,t)\nabla\varphi(x,t)\,dx\,dt - \int_I\int_{\partial\Omega} \beta(q^4(x,t) - u^4(x,t))\varphi(x,t)\,ds\,dt,$$

and the right-hand side and the initial condition

$$f = 0 \quad \text{and} \quad u_0(q) = 0.$$

Actually, we have to impose L^∞-constraints on the control because otherwise q^4 would in general not be integrable. As we consider in this thesis only unconstrained optimization problems, we assume here the constraint to be inactive at the optimal solution.

Example 2.3 (Control via initial condition)**.** We consider for a given desired solution profile $\hat{u} \in L^2(I, L^2(\partial\Omega))$ the control problem

$$\text{Minimize } J(q,u) = \frac{1}{2}\int_I \|u(t) - \hat{u}(t)\|^2_{L^2(\partial\Omega)}\,dt + \frac{\alpha}{2}\|q - \hat{q}\|^2_{L^2(\Omega)}$$

subject to the Ginzburg Landau equation with $\varepsilon > 0$

$$\begin{aligned} \partial_t u - \varepsilon\Delta u + u + u^3 &= 0 && \text{in } \Omega \times I, \\ \varepsilon\partial_n u &= 0 && \text{on } \partial\Omega \times I, \\ u &= q && \text{on } \Omega \times \{\,0\,\}. \end{aligned}$$

To embed this example in the abstract setting, we choose the spaces

$$H = L^2(\Omega), \quad V = H^1(\Omega), \quad R = L^2(\Omega), \quad \text{and} \quad Q = \mathcal{P}_0(\bar{I}, R) = \mathcal{P}_0(\bar{I}, L^2(\Omega)),$$

define the functionals

$$J_1(u(t)) = \frac{1}{2}\|u(t) - \hat{u}(t)\|^2_{L^2(\partial\Omega)} \quad \text{and} \quad J_2(u(T)) = 0,$$

the semilinear form

$$a(q,u)(\varphi) = \int_I \int_\Omega \varepsilon \nabla u(x,t) \nabla \varphi(x,t)\, dx\, dt + \int_I \int_\Omega (u(x,t) + u^3(x,t))\varphi(x,t)\, dx\, dt,$$

and the right-hand side and the initial condition

$$f = 0 \quad \text{and} \quad u_0(q) = q.$$

Due to the choice of the control space Q, we have

$$\frac{\alpha}{2}\|q - \hat{q}\|^2_{L^2(\Omega)} = \frac{\alpha}{2T}\int_I \|q(t) - \hat{q}\|^2_{L^2(\Omega)}\, dt = \frac{\alpha}{2T}\|q - \hat{q}\|^2_Q,$$

and also this concrete regularization fits in the abstract framework with the scaled regularization parameter α/T.

Up to now, all presented examples were control problems with infinite-dimensional control space Q. The following last example shows that also parameter estimation problems with finite-dimensional parameter space Q fit in our setting:

Example 2.4 (Parameter estimation)**.** We consider for given reference measurements $\hat{u} \in L^2(I, L^2(\Omega)^n)$ the parameter estimation problem

$$\text{Minimize } J(q,u) = \frac{1}{2}\int_I \|\nabla u(t) - \hat{u}(t)\|^2_{L^2(\Omega)^n}\, dt + \frac{\alpha}{2}|q - \hat{q}|^2$$

subject to the nonlinear reaction-diffusion equation with $\varepsilon > 0$

$$\begin{aligned} \partial_t u - \varepsilon\Delta u + \exp(q^2 u) &= 2 && \text{in } \Omega \times I, \\ u &= 0 && \text{on } \partial\Omega \times I, \\ u &= 0 && \text{on } \Omega \times \{\,0\,\}. \end{aligned}$$

To embed this example in the abstract setting, we choose the spaces

$$H = L^2(\Omega), \quad V = H^1_0(\Omega), \quad R = \mathbb{R}, \quad \text{and} \quad Q = \mathcal{P}_0(\bar{I}, R) = \mathcal{P}_0(\bar{I}, \mathbb{R}),$$

define the functionals

$$J_1(u(t)) = \frac{1}{2}\|\nabla u(t) - \hat{u}(t)\|^2_{L^2(\Omega)^n} \quad \text{and} \quad J_2(u(T)) = 0,$$

the semilinear form

$$a(q,u)(\varphi) = \int_I \int_\Omega \nabla u(x,t) \nabla \varphi(x,t)\, dx\, dt + \int_I \int_\Omega \exp(q^2 u(x,t))\varphi(x,t)\, dx\, dt,$$

and the right-hand side and the initial condition

$$f = 2 \quad \text{and} \quad u_0(q) = 0.$$

As in Example 2.3, the regularization employed here is equivalent to the regularization term $\alpha/2T\|q - \hat{q}\|^2_Q$. However, in contrast to the examples presented before with infinite-dimensional control space, the solvability of this finite-dimensional parameter estimation problem can be assured also for the case $\alpha = 0$ under an additional condition on the cost functional: The reference measurements $\hat{u}$ have to be chosen such that in the case $\alpha = 0$ the minimum value J^* of the cost functional fulfills the relation

$$J^* < \frac{1}{2} \int_I \|\hat{u}(t)\|^2_{L^2(\Omega)^n} \, dt.$$

A proof of existence of optimal controls for the elliptic analog to this example utilizing such a condition on the cost functional is given in Vexler [82]. This proof can be transfered directly to the parabolic case considered here.

In these examples, the choice of the spaces is driven by the aim of fulfilling the minimal requirements for stating the optimization problems. To guarantee solvability of the presented problems or the uniqueness of solutions, the spaces potentially have to be restricted. For further details, we refer to the following section and the literature cited therein.

Further examples of nonlinear parabolic control problems motivated from concrete applications can be found for instance in Neittaanmäki and Tiba [64].

2.3 Existence and uniqueness of solutions

There is a number of publications, where the question of existence of solutions to optimization problems as stated above is discussed; see for example the textbooks Lions [53], Fursikov [37], and Tröltzsch [78]. Therein, the authors follow mainly two different approaches:

- The *non-reduced approach*, where the state and the control variables are treated explicitly.
- The *reduced approach*, where the state variable is eliminated and treated implicitly.

In what follows, we sketch how to apply the techniques from the reduced approach to the considered optimization problems.

A key ingredient in proving the solvability of PDE-constrained optimization problems as ($\mathbb{P}$) via the reduced approach is the existence of a solution operator S which maps a given control q to the unique solution $u = S(q)$ of the state equation (2.3). That is, S is characterized by the implicit relation

$$(\partial_t S(q), \varphi)_I + a(q, S(q))(\varphi) + (S(q)(0), \varphi(0)) = (f, \varphi)_I + (u_0(q), \varphi(0)) \quad \forall \varphi \in X. \tag{2.6}$$

Obviously, the existence of a solution operator is implied by the property of unique solvability of the state equation. There are several sets of assumptions on the structure of the bilinear form $a(\cdot,\cdot)(\cdot)$ and its dependence on the control variable ensuring existence and uniqueness of solutions to the considered state equation. We do not rely on a specific set but assume throughout this thesis the existence of a solution operator $S\colon Q \to X$. This assumption is illustrated for the configuration of Example 2.1 at the end of this section.

Provided the assumed existence of the solution operator S, we are able to define the *reduced cost functional* $j\colon Q \to \mathbb{R}$ by

$$j(q) \coloneqq J(q, S(q)).$$

This definition allows us to reformulate problem ($\mathbb{P}$) as the unconstrained optimization problem

$$\text{Minimize } j(q), \quad q \in Q. \tag{$\mathbb{P}^{\text{red}}$}$$

The reduced formulation enables us to apply the classical existence theorem from the calculus of variations to the abstract optimization problem under consideration. We recall here the formulation from Dacorogna [24]:

Theorem 2.1. *Let the reduced cost functional $j\colon Q \to \mathbb{R}$ be weakly lower semicontinuous, that is*

$$\liminf_{i\to\infty} j(q_i) \geq j(q) \quad \textit{whenever} \quad q_i \rightharpoonup q \textit{ in } Q,$$

and let j be coercive over Q, that is

$$j(q) \geq \alpha \|q\|_Q + \beta$$

for every $q \in Q$ and for some $\alpha > 0$, $\beta \in \mathbb{R}$. Then, problem ($\mathbb{P}^{\text{red}}$) has at least one solution $q \in Q$.

Proof. Let $(q_i)_{i\in\mathbb{N}}$ be a minimizing sequence for ($\mathbb{P}^{\text{red}}$), that is

$$j(q_i) \to \inf_{r\in Q} j(r).$$

From the hypothesis of coercivity we may deduce for $i > i_0$, i_0 large enough, that there exists $K > 0$ such that

$$\|q_i\|_Q < K \quad \forall i > i_0.$$

Since Q is a Hilbert space, we can extract a weakly convergent subsequence (also denoted by $(q_i)_{i\in\mathbb{N}}$) such that

$$q_i \rightharpoonup q \text{ in } Q.$$

Then, the weak lower semicontinuity of j implies the result. □

To prove uniqueness of solutions to optimization problem ($\mathbb{P}^{\text{red}}$), stronger requirements on the functional j have to be imposed:

Theorem 2.2. *Let the reduced cost functional j fulfill the hypotheses of Theorem 2.1. If j is additionally strongly convex on Q, that is*

$$j(\lambda q_1 + (1-\lambda)q_2) < \lambda j(q_1) + (1-\lambda) j(q_2)$$

for all $\lambda \in (0,1)$ and all $q_1, q_2 \in Q$ with $q_1 \neq q_2$, then problem ($\mathbb{P}^{\text{red}}$) has a unique solution.

Proof. Assume q_1 and q_2 are two solutions of ($\mathbb{P}^{\text{red}}$). Then, we have for arbitrary $\lambda \in (0, 1)$

$$j(\lambda q_1 + (1-\lambda)q_2) < \lambda j(q_1) + (1-\lambda)j(q_2) = \min_{q \in Q} j(q),$$

which is a contradiction. □

The main difficulty in applying these theorems to concrete PDE-constrained optimization problems is to verify the conditions proposed on the reduced cost functional j. We present here just the construction of proving existence and uniqueness in the abstract setting established in Section 2.2 for the linear-quadratic optimization problem (2.5) considered in Example 2.1.

Remark 2.4. For several concrete configurations, the requirement on the reduced cost functional to be lower semicontinuous can not be met. In many of these situations, the solvability of the optimization problems can still be proven using the non-reduced approach.

The first step is to ensure the unique solvability of the state equation which is here the linear heat equation:

Theorem 2.3. *Let $H = L^2(\Omega)$ and $V = H^1_0(\Omega)$. Then, the linear parabolic equation*

$$\begin{aligned} \partial_t u - \varepsilon \Delta u &= f && \text{in } \Omega \times I, \\ u &= 0 && \text{on } \partial\Omega \times I, \\ u &= u_0 && \text{on } \Omega \times \{0\} \end{aligned}$$

admits for $f \in L^2(I, V^*)$, $u_0 \in H$, and $\varepsilon > 0$ a unique solution $u \in X$. Furthermore, the solution depends continuously on the data, that is the mapping

$$(f, u_0) \mapsto u$$

is continuous from $L^2(I, V^*) \times H$ to X.

Proof. The proof can be found for instance in Lions [53] and Wloka [88]. □

Theorem 2.3 implies the linearity and continuity of the solution operator $S\colon Q \to X$, $q \mapsto u$ of the state equation (2.5b). Then, the reduced cost functional $j\colon Q \to \mathbb{R}$,

$$j(q) = J(q, S(q)) = \frac{1}{2}\int_I \|S(q)(t) - \hat{u}(t)\|^2_{L^2(\Omega)}\,dt + \frac{\alpha}{2}\int_I \|q(t)\|^2_{L^2(\Omega)}\,dt$$

is continuous and convex. Thus, j is weakly lower semicontinuous; cf. Dacorogna [24]. Since j is coercive for $\alpha > 0$, we may apply Theorem 2.1 to obtain, that optimization problem (2.5) admits at least one solution in this case. Moreover, $\alpha > 0$ implies strong convexity of j and, by Theorem 2.2, the solution of (2.5) is unique.

The steps in proving existence and uniqueness of optimal solutions for a wider class of optimization problems, for example for the case of semilinear state equations, are quite similar to the presented linear-quadratic case. However, the proofs are more involved since the nonlinearities have to be treated, too. Furthermore, the solution operator S is nonlinear and thus, the continuity of J does no longer imply weak lower semicontinuity of j. Examples of such proofs for concrete problems can be found for instance in the textbooks cited at the beginning of this section.

Since this topic is not the major purpose of this thesis, we will not go more into detail and assume in what follows that problem ($\mathbb{P}$) admits a (locally) unique solution. Moreover, we assume the existence of a neighborhood $W \subseteq Q \times X$ of the optimal solution, such that the linearized operator $A'_u(q(t), u(t)) \colon V \to V^*$ is an isomorphism for all $(q, u) \in W$ and almost all $t \in I$. This assumption allows all considered linearized and adjoint problems stated in this thesis to be well posed.

2.4 Optimality conditions

In this section, we formulate standard necessary and sufficient optimality conditions for the reformulated unconstrained optimization problem ($\mathbb{P}^{\text{red}}$).

The theorems presented in the previous section ensure the existence and uniqueness of *global* solutions. The optimality conditions presented here are formulated more generally by means of *local* solutions: A control $q \in Q$ is called local solution of the optimization problem ($\mathbb{P}^{\text{red}}$) if there exists a neighborhood $Q_0 \subseteq Q$ containing q such that

$$j(q) \leq j(r) \quad \forall r \in Q_0.$$

Before presenting the optimality conditions, we recall for the convenience of the reader the standard definitions of differentiability in normed vector spaces, which can be found for example in Jahn [47]:

Definition 2.1 (Directional derivative)**.** Let Y and Z be normed vector spaces, Y_0 be a nonempty subset of Y and $f \colon Y_0 \to Z$ be a given mapping. If for two elements $y \in Y_0$ and $\delta y \in Y$ the limit

$$f'(y)(\delta y) := \lim_{\lambda \downarrow 0} \frac{f(y + \lambda \delta y) - f(y)}{\lambda}$$

exists, then $f'(y)(\delta y)$ is called the directional derivative of f at y in direction δy. If this limit exists for all $\delta y \in Y$, then f is called directionally differentiable at y.

Definition 2.2 (Gâteaux derivative)**.** Let Y and Z be normed vector spaces, Y_0 be a nonempty subset of Y. A directionally differentiable mapping $f \colon Y_0 \to Z$ is called Gâteaux differentiable at $y \in Y_0$, if the directional derivative $f'(y)$ is a continuous linear mapping from Y to Z. $f'(y)$ is then called Gâteaux derivative of f at y.

Definition 2.3 (Fréchet derivative)**.** Let Y and Z be normed vector spaces, Y_0 be a nonempty subset of Y and $f \colon Y_0 \to Z$ be a given mapping. Furthermore let an element $y \in Y_0$ be given. If there is a continuous linear mapping $f'(y) \colon Y \to Z$ with the property

$$\lim_{\|\delta y\|_Y \to 0} \frac{\|f(y + \delta y) - f(y) - f'(y)(\delta y)\|_Z}{\|\delta y\|_Y} = 0,$$

then $f'(y)$ is called the Fréchet derivative of f at y and f is called Fréchet differentiable at y.

Remark 2.5. The given definitions of the different kinds of derivatives can directly be extended to higher order derivatives. We give here as example the definition of second order Fréchet derivatives: If $f\colon Y_0 \to Z$ is Fréchet differentiable at all $y \in Y_0$ and the mapping $f'\colon Y_0 \to L(Y, Z)$ is Fréchet differentiable at $y \in Y_0$, then f is called two times Fréchet differentiable at y. The second derivative is denoted by $f''(y) := (f')'(y)$. We call f two times continuously Fréchet differentiable at $y \in Y_0$ if the second derivative f'' is continuous in y.

We are now prepared to state the first and second order necessary and second order sufficient optimality conditions. The proofs follow the classical techniques, see for instance Tröltzsch [78], and are presented here just for completeness.

Theorem 2.4 (First order necessary optimality condition). *Let the reduced cost functional j be Gâteaux differentiable on an open subset $Q_0 \subseteq Q$. If $q \in Q_0$ is a local optimal solution of the optimization problem* ($\mathbb{P}^{\text{red}}$)*, then there holds the first order necessary optimality condition*

$$j'(q)(\delta q) = 0 \quad \forall \delta q \in Q. \tag{2.7}$$

Proof. For given direction $\delta q \in Q$, there exists $\lambda > 0$ such that $q + \lambda\delta q \in Q_0$ and $j(q + \lambda\delta q) \geq j(q)$. Then, we have

$$\frac{j(q + \lambda\delta q) - j(q)}{\lambda} \geq 0.$$

With λ tending to 0, we obtain

$$j'(q)(\delta q) \geq 0.$$

Since the Gâteaux derivative is linear in δq and since with δq also $-\delta q$ is a feasible direction, we achieve the stated condition. □

Remark 2.6. If a convex (not necessary open) subset $Q_0 \subseteq Q$ is considered, the first order necessary optimality condition (2.7) is given as the variational inequality

$$j'(q)(q - \delta q) \geq 0 \quad \forall \delta q \in Q_0.$$

This situation occurs in many application problems where additional constraints on the control variable have to be imposed.

Remark 2.7. If the functional j is additionally convex, that is

$$j(\lambda q_1 + (1 - \lambda)q_2) \leq \lambda j(q_1) + (1 - \lambda)j(q_2)$$

for all $\lambda \in [0, 1]$ and all $q_1, q_2 \in Q$, then condition (2.7) is also sufficient for q to be a solution of ($\mathbb{P}^{\text{red}}$).

Theorem 2.5 (Second order necessary optimality condition). *Let the reduced cost functional j be two times continuously Fréchet differentiable on an open subset $Q_0 \subseteq Q$. If $q \in Q_0$ is a local optimal solution of the optimization problem* ($\mathbb{P}^{\text{red}}$)*, then there holds the second order necessary optimality condition*

$$j''(q)(\delta q, \delta q) \geq 0 \quad \forall \delta q \in Q.$$

Proof. For given $\delta q \in Q$, there is $\lambda > 0$ such that $q + \lambda\delta q \in Q_0$. It holds by Taylor expansion

$$0 \leq j(q + \lambda\delta q) - j(q) = \lambda j'(q)(\delta q) + \frac{\lambda^2}{2} j''(q)(\delta q, \delta q) + r_2^j(q, \lambda\delta q)$$

with the remainder term r_2^j of second order. Applying the first order necessary optimality condition and dividing by $\lambda^2/2$ leads to

$$0 \leq j''(q)(\delta q, \delta q) + \frac{2r_2^j(q, \lambda\delta q)}{\lambda^2}.$$

Tending to the limit $\lambda \downarrow 0$ yields the stated condition. □

Theorem 2.6 (Second order sufficient optimality condition)**.** *Let the reduced cost functional j be two times continuously Fréchet differentiable on a neighborhood $Q_0 \subseteq Q$ of q. Moreover, let the control q fulfill the first order necessary optimality condition*

$$j'(q)(\delta q) = 0 \quad \forall \delta q \in Q$$

and assume the existence of $\gamma > 0$ such that the second order sufficient optimality condition

$$j''(q)(\delta q, \delta q) \geq \gamma \|\delta q\|_Q^2 \quad \forall \delta q \in Q$$

is valid. Then, a constant $\rho > 0$ exists such that the quadratic growth condition

$$j(q + \delta q) \geq j(q) + \frac{\gamma}{4}\|\delta q\|_Q^2$$

holds for all $\delta q \in Q$ with $\|\delta q\|_Q \leq \rho$. Consequently, q is a local solution of the optimization problem ($\mathbb{P}^{\text{red}}$).

Proof. By means of Taylor expansion, we obtain with some $\theta \in (0, 1)$ for ρ chosen small enough such that $q + \delta q \in Q_0$ for $\|\delta q\|_Q \leq \rho$

$$\begin{aligned} j(q + \delta q) &= j(q) + j'(q)(\delta q) + \frac{1}{2} j''(q + \theta\delta q)(\delta q, \delta q) \\ &= j(q) + \frac{1}{2} j''(q + \theta\delta q)(\delta q, \delta q) \\ &= j(q) + \frac{1}{2} j''(q)(\delta q, \delta q) + \frac{1}{2}[j''(q + \theta\delta q) - j''(q)](\delta q, \delta q). \end{aligned}$$

For $\|\delta q\|_Q \leq \rho$ small, the proposed continuity of j'' yields

$$|[j''(q + \theta\delta q) - j''(q)](\delta q, \delta q)| \leq \frac{\gamma}{2}\|\delta q\|_Q^2.$$

In total, we achieve by applying the second assumption

$$j(q + \delta q) \geq j(q) + \frac{\gamma}{2}\|\delta q\|_Q^2 - \frac{\gamma}{4}\|\delta q\|_Q^2 = j(q) + \frac{\gamma}{4}\|\delta q\|_Q^2,$$

which is the stated result. □

Remark 2.8. Often, j is Fréchet differentiable with respect to a "stronger" space $\widetilde{Q} \subset Q$ and the coercivity of j'' can only be shown with respect to Q. In this situation, the so called *two-norm-discrepancy* occurs and the second order sufficient condition has to be formulated using the norms of both spaces $\widetilde{Q}$ and Q; see for instance Tröltzsch [78] for details on this.

2.5 Representation formulas for the derivatives

For evaluating the optimality conditions stated in the previous section and to apply derivative-based optimization algorithms to the optimization problem under consideration, it is necessary to provide computable representations of the first and second derivatives of the reduced cost functional j. Here, "computable" has to be understood in the sense, that similar expressions can be evaluated on the discrete level; see Chapter 3.

In this section, we establish different computable representations of the derivatives and discuss their advantages and disadvantages depending on the concrete configuration of the optimization problem. The presented construction is already published in Becker, Meidner, and Vexler [8]. Similar derivations of the presented concepts for other types of state equations or in terms of operators instead of semilinear forms can be found in Becker [2, 3] and Ulbrich [79].

Throughout this thesis, we assume the semilinear form a, the cost functional J, and the solution operator S to be smooth enough that all required directional derivatives exist; see for instance Tröltzsch [78] for a discussion on Fréchet differentiability of S in some model configurations. We indicate the variables to which the directional derivatives are applied by a subscript. For instance $a'_q(q,u)(\delta q,\varphi)$ denotes the directional derivative of the semilinear form $a(q,u)(\varphi)$ with respect to q in direction δq.

2.5.1 First derivatives

The most obvious approach for calculating the first derivatives of j is the so-called *sensitivity approach.* For $q \in Q$ and a direction $\delta q \in Q$, the chain rule yields with $u = S(q)$ for the directional derivative $j'(q)$ the expression

$$j'(q)(\delta q) = \alpha(q - \hat{q}, \delta q)_Q + \int_I J'_1(u)(\delta u)\,dt + J'_2(u(T))(\delta u(T)). \tag{2.8}$$

Here, the sensitivity $\delta u := S'(q)(\delta q) \in X$ is required to evaluate this expression. By totally differentiating the state equation (2.3) with respect to q in the direction δq (see Becker [2] for a rigorous justification of this procedure), we obtain

$$\begin{aligned}(\partial_t \delta u, \varphi)_I + a'_u(q,u)(\delta u,\varphi) + (\delta u(0), \varphi(0))&\\ = -a'_q(q,u)(\delta q,\varphi) + (u'_0(q)(\delta q), \varphi(0)) \quad \forall \varphi \in X.&\end{aligned} \tag{2.9}$$

Hence, the sensitivity δu is given as the solution of the linearized state equation (2.9). Therefore, to calculate the directional derivative $j'(q)(\delta q)$ for a given direction δq via the sensitivity approach, the following two steps are required (additionally to the solution of the state equation for $q \in Q$):

(i) Compute the sensitivity $\delta u \in X$ by solving (2.9).

(ii) Compute $j'(q)(\delta q)$ via (2.8).

This procedure is expensive if the whole derivative $j'(q)$ respectively the gradient $\nabla j(q)$ is required, as in this case for a whole basis $\{ \delta q_i \}$ of Q, the derivatives $j'(q)(\delta q_i)$ have to be computed. Since each of these evaluations requires the solution of a linear parabolic equation, this procedure rapidly becomes prohibitive for large dimensions of the (discretized) control space.

We now derive a more efficient way to represent the derivative $j'(q)$. For this so-called *adjoint approach*, we make use of the Lagrangian $\mathcal{L}\colon Q \times X \times X \to \mathbb{R}$ of the optimization problem $(\mathbb{P})$ defined by

$$\mathcal{L}(q, u, z) := J(q, u) + (f - \partial_t u, z)_I - a(q, u)(z) + (u_0(q) - u(0), z(0)).$$

Remark 2.9. In the given definition of the Lagrangian, the choice of the Lagrange multiplier $z(0) \in H$ for the initial condition $u_0(q) - u(0)$ seems to be arbitrary. However, choosing a second independent Lagrange multiplier $\tilde{z} \in H$ for coupling the initial condition yields necessarily the condition $\tilde{z} = z(0)$ in stationary points of the Lagrangian. Hence, the a priori choice of $\tilde{z} = z(0)$ constitutes no restriction.

By means of the useful identity

$$j(q) = J(q, u) = \mathcal{L}(q, u, z),$$

which holds true for $u = S(q)$ and arbitrary $z \in X$, we obtain with the abbreviation $\delta u = S'(q)(\delta q)$ the following expression of the directional derivative $j'(q)(\delta q)$ in terms of derivatives of the Lagrangian:

$$j'(q)(\delta q) = \mathcal{L}'_q(q, u, z)(\delta q) + \mathcal{L}'_u(q, u, z)(\delta u).$$

If we now determine the adjoint state $z \in X$ such that the adjoint equation

$$\mathcal{L}'_u(q, u, z)(\varphi) = 0 \quad \forall \varphi \in X$$

is fulfilled, then we obtain the expression

$$\begin{aligned} j'(q)(\delta q) &= \mathcal{L}'_q(q, u, z)(\delta q) \\ &= \alpha(q - \hat{q}, \delta q)_Q - a'_q(q, u)(\delta q, z) + (u'_0(q)(\delta q), z(0)). \end{aligned} \tag{2.10}$$

The *adjoint equation* is given in explicit form as

$$\begin{aligned} &-(\varphi, \partial_t z)_I + a'_u(q, u)(\varphi, z) + (\varphi(T), z(T)) \\ &\qquad = \int_I J'_1(u)(\varphi)\, dt + J'_2(u(T))(\varphi(T)) \quad \forall \varphi \in X. \end{aligned} \tag{2.11}$$

It is obtained by integration by parts with respect to time which is admissible for functions in X; cf. Wloka [88].

For evaluating the directional derivative $j'(q)(\delta q)$ in a given direction δq via the adjoint approach, the following two steps are required:

(i) Compute the adjoint state $z \in X$ by solving (2.11).

(ii) Compute $j'(q)(\delta q)$ via (2.10).

In contrast to the proceeding in the sensitivity approach, here the adjoint equation has to be solved only once—even if the whole derivative $j'(q)$ is needed. This is because the adjoint equation does not depend on the direction δq. If $j'(q)(\delta q_i)$ is required for a basis $\{\, \delta q_i \,\}$ of Q, only expression (2.10) has to be evaluated one by one for each $\delta q \in \{\, \delta q_i \,\}$. Thus, this variant of expressing the first derivatives of the reduced cost functional j is also applicable to optimization problems with high-dimensional (discretized) control space.

Remark 2.10. Due to the regularity assumption on the linearization of the semilinear form, the necessary optimality condition of first order stated in Theorem 2.4 in terms of the reduced functional is equivalent to the existence of a triple $(q, u, z) \in Q \times X \times X$ solving the optimality system of problem ($\mathbb{P}$). It is given by the derivatives of the Lagrangian defined above:

$$\begin{aligned} \mathcal{L}'_z(q,u,z)(\varphi) &= 0 \quad \forall \varphi \in X \qquad \text{(State equation)},\\ \mathcal{L}'_u(q,u,z)(\varphi) &= 0 \quad \forall \varphi \in X \qquad \text{(Adjoint equation)},\\ \mathcal{L}'_q(q,u,z)(\psi) &= 0 \quad \forall \psi \in Q \qquad \text{(Gradient equation)}. \end{aligned} \tag{2.12}$$

2.5.2 Second derivatives

We now turn to the calculation of the second derivatives of the reduced cost functional. They are needed for instance for the evaluation of the second order optimality condition or to apply second order optimization algorithms like Newton's method to solve problem ($\mathbb{P}^{\text{red}}$).

Since the sensitivity approach was not competitive for the computation of the first derivatives, we restrict ourselves to the adjoint approach. Even though, we obtain two different representations of the second derivatives. But in contrast to the two representations of the first derivatives, they are both applicable and have their advantages in different configurations of concrete optimization problems.

For representing the second derivatives, we take into account the implicit dependence of the adjoint solution z on the control due to (2.11): We assume the existence of a sufficiently smooth solution operator $T\colon Q \to X$ with $z = T(q)$. Then, differentiation of

$$j(q) = \mathcal{L}(q,u,z) = \mathcal{L}(q, S(q), T(q))$$

yields for two given directions $\delta q, \tau q \in Q$

$$\begin{aligned} j''(q)(\delta q, \tau q) = \quad & \mathcal{L}''_{qq}(q,u,z)(\delta q,\tau q) + \mathcal{L}''_{qu}(q,u,z)(\delta q,\tau u) + \mathcal{L}''_{qz}(q,u,z)(\delta q,\tau z)\\ &+ \mathcal{L}''_{uq}(q,u,z)(\delta u,\tau q) + \mathcal{L}''_{uu}(q,u,z)(\delta u,\tau u) + \mathcal{L}''_{uz}(q,u,z)(\delta u,\tau z)\\ &+ \mathcal{L}''_{zq}(q,u,z)(\delta z,\tau q) + \mathcal{L}''_{zu}(q,u,z)(\delta z,\tau u)\\ &\qquad\qquad + \mathcal{L}'_u(q,u,z)(\delta\tau u) + \mathcal{L}'_z(q,u,z)(\delta\tau z). \end{aligned} \tag{2.13}$$

The abbreviations of the directions obtained by the chain rule are defined as

$$\begin{aligned} \delta u &:= S'(q)(\delta q), & \tau u &:= S'(q)(\tau q), & \delta\tau u &:= S''(q)(\delta q,\tau q),\\ \delta z &:= T'(q)(\delta q), & \tau z &:= T'(q)(\tau q), & \delta\tau z &:= T''(q)(\delta q,\tau q). \end{aligned}$$

Since u is assumed to be a solution of the state equation (2.3) and z to be a solution of the adjoint equation (2.11), the two terms in the last line of (2.13) vanish. Consequently, we obtain the representation

$$\begin{aligned} j''(q)(\delta q, \tau q) = \quad & \mathcal{L}''_{qq}(q,u,z)(\delta q, \tau q) + \mathcal{L}''_{qu}(q,u,z)(\delta q, \tau u) + \mathcal{L}''_{qz}(q,u,z)(\delta q, \tau z) \\ & + \mathcal{L}''_{uq}(q,u,z)(\delta u, \tau q) + \mathcal{L}''_{uu}(q,u,z)(\delta u, \tau u) + \mathcal{L}''_{uz}(q,u,z)(\delta u, \tau z) \\ & + \mathcal{L}''_{zq}(q,u,z)(\delta z, \tau q) + \mathcal{L}''_{zu}(q,u,z)(\delta z, \tau u). \end{aligned} \tag{2.14}$$

Here, the boxes indicate the two different possibilities to express the second derivatives, which we discuss in what follows:

(I) We collect all terms containing the directions τz and τu and require them to be zero for all possible directions by choosing $\delta u \in X$ and $\delta z \in X$ such that

$$\mathcal{L}''_{qz}(q,u,z)(\delta q, \varphi) + \mathcal{L}''_{uz}(q,u,z)(\delta u, \varphi) = 0 \quad \forall \varphi \in X, \tag{2.15a}$$

$$\mathcal{L}''_{qu}(q,u,z)(\delta q, \varphi) + \mathcal{L}''_{uu}(q,u,z)(\delta u, \varphi) + \mathcal{L}''_{zu}(q,u,z)(\delta z, \varphi) = 0 \quad \forall \varphi \in X. \tag{2.15b}$$

These terms are the parts of (2.14) boxed by solid lines. Then, the remainder parts are the expression of the second derivative:

$$j''(q)(\delta q, \tau q) = \mathcal{L}''_{qq}(q,u,z)(\delta q, \tau q) + \mathcal{L}''_{uq}(q,u,z)(\delta u, \tau q) + \mathcal{L}''_{zq}(q,u,z)(\delta z, \tau q). \tag{2.16}$$

(II) We collect all terms containing the directions τz and δz and require them to be zero for all possible directions by choosing $\delta u \in X$ and $\tau u \in X$ such that

$$\mathcal{L}''_{qz}(q,u,z)(\delta q, \varphi) + \mathcal{L}''_{uz}(q,u,z)(\delta u, \varphi) = 0 \quad \forall \varphi \in X, \tag{2.17a}$$

$$\mathcal{L}''_{zq}(q,u,z)(\varphi, \tau q) + \mathcal{L}''_{zu}(q,u,z)(\varphi, \tau u) = 0 \quad \forall \varphi \in X. \tag{2.17b}$$

These are the parts of (2.14) boxed by dashed lines. Then, the remainder parts are the expression of the second derivative:

$$\begin{aligned} j''(q)(\delta q, \tau q) = \mathcal{L}''_{qq}(q,u,z)(\delta q, \tau q) + \mathcal{L}''_{qu}(q,u,z)(\delta q, \tau u) \\ + \mathcal{L}''_{uq}(q,u,z)(\delta u, \tau q) + \mathcal{L}''_{uu}(q,u,z)(\delta u, \tau u). \end{aligned} \tag{2.18}$$

Before discussing the advantages and disadvantages of the two derived representations, we first present the concrete form of the expressions formulated in terms of the Lagrangian above. First, we note that the equations (2.15a), (2.17a), and (2.17b) are identical if we are allowed to change the order of taking derivatives. Taking the concrete form of the Lagrangian into account, we obtain the explicit formulation of these three equations for $\delta u \in X$:

$$(\partial_t \delta u, \varphi)_I + a'_u(q,u)(\delta u, \varphi) + (\delta u(0), \varphi(0)) = -a'_q(q,u)(\delta q, \varphi) + (u'_0(q)(\delta q), \varphi(0)) \quad \forall \varphi \in X.$$

This is the already known linearized state or *tangent equation* (2.9).

The explicit formulation of (2.15b) is given by an *additional adjoint equation* for $\delta z \in X$:

$$\begin{aligned} -(\varphi, \partial_t \delta z)_I + a'_u(q,u)(\varphi, \delta z) + (\varphi(T), \delta z(T)) = -a''_{uu}(q,u)(\delta u, \varphi, z) \\ - a''_{qu}(q,u)(\delta q, \varphi, z) + \int_I J''_1(u)(\delta u, \varphi)\, dt + J''_2(u(T))(\delta u(T), \varphi(T)) \quad \forall \varphi \in X. \end{aligned} \tag{2.19}$$

Let us now summarize the necessary computational steps to assess the second derivative $j''(q)(\delta q, \tau q)$ in two given directions δq and τq. We therefor assume that the state u and adjoint state z are already computed for the given control q.

(I)
- (i) Compute the solution $\delta u \in X$ of the tangent equation (2.9) for the direction δq.
- (ii) Compute the solution $\delta z \in X$ of the additional adjoint equation (2.19).
- (iii) Compute $j''(q)(\delta q, \tau q)$ via:

$$\begin{aligned} j''(q)(\delta q, \tau q) = \alpha(\delta q, \tau q)_Q - a''_{qq}(q,u)(\delta q, \tau q, z) - a''_{uq}(q,u)(\delta u, \tau q, z) \\ - a'_q(q,u)(\tau q, \delta z) + (u'_0(q)(\tau q), \delta z(0)) + (u''_0(q)(\delta q, \tau q), z(0)). \end{aligned} \tag{2.20}$$

(II)
- (i) Compute the solution $\delta u \in X$ of the tangent equation (2.9) for the direction δq.
- (ii) Compute the solution $\tau u \in X$ of the tangent equation (2.9) for the direction τq.
- (iii) Compute $j''(q)(\delta q, \tau q)$ via:

$$\begin{aligned} j''(q)(\delta q, \tau q) = \alpha(\delta q, \tau q)_Q + \int_I J''_1(u)(\delta u, \tau u)\, dt + J''_2(u(T))(\delta u(T), \tau u(T)) \\ - a''_{qq}(q,u)(\delta q, \tau q, z) - a''_{uq}(q,u)(\delta u, \tau q, z) - a''_{qu}(q,u)(\delta q, \tau u, z) \\ - a''_{uu}(q,u)(\delta u, \tau u, z) + (u''_0(q)(\delta q, \tau q), z(0)). \end{aligned} \tag{2.21}$$

If the derivative $j''(q)(\delta q, \tau q)$ is required only once for two given directions δq and τq, the approaches (I) and (II) have the same effort. However, if $j''(q)(\delta q, \tau q)$ has to be evaluated for one δq and τq running through a basis $\{\, \tau q_i \,\}$ of Q, approach (I) is much more efficient than approach (II) since in (I) the auxiliary solutions do not depend on $\tau q \in \{\, \tau q_i \,\}$ and have to be computed only once. Conversely, if $j''(q)(\tau q_i, \tau q_j)$ is required for a basis $\{\, \tau q_i \,\}$, it is cheaper to use approach (II) because here we only need to solve the tangent equation (2.9) for all $\delta q \in \{\, \tau q_i \,\}$ and do not need to compute the solutions of the additional adjoint equation (2.19). A criterion indicating in which situations one should use which approach inside a Newton-based optimization loop is given in Chapter 4.

Finally, we collect the representations of the three auxiliary equations for comparing them later on with the corresponding discrete equations established in Chapter 3. First, we recall the formulations in terms of derivatives of the Lagrangian:

Adjoint: Find $z \in X$ such that

$$\mathcal{L}'_u(q,u,z)(\varphi) = 0 \quad \forall \varphi \in X.$$

Tangent: Find $\delta u \in X$ such that

$$\mathcal{L}''_{qz}(q,u,z)(\delta q, \varphi) + \mathcal{L}''_{uz}(q,u,z)(\delta u, \varphi) = 0 \quad \forall \varphi \in X.$$

Additional Adjoint: Find $\delta z \in X$ such that

$$\mathcal{L}''_{qu}(q,u,z)(\delta q, \varphi) + \mathcal{L}''_{uu}(q,u,z)(\delta u, \varphi) + \mathcal{L}''_{zu}(q,u,z)(\delta z, \varphi) = 0 \quad \forall \varphi \in X.$$

Explicitly, the equations are given as follows:

Adjoint: Find $z \in X$ such that

$$
\begin{aligned}
&-(\varphi, \partial_t z)_I + a'_u(q,u)(\varphi, z) + (\varphi(T), z(T)) \\
&\qquad = \int_I J'_1(u)(\varphi)\, dt + J'_2(u(T))(\varphi(T)) \quad \forall \varphi \in X.
\end{aligned} \tag{2.11}
$$

Tangent: Find $\delta u \in X$ such that

$$
\begin{aligned}
&(\partial_t \delta u, \varphi)_I + a'_u(q,u)(\delta u, \varphi) + (\delta u(0), \varphi(0)) \\
&\qquad = -a'_q(q,u)(\delta q, \varphi) + (u'_0(q)(\delta q), \varphi(0)) \quad \forall \varphi \in X.
\end{aligned} \tag{2.9}
$$

Additional Adjoint: Find $\delta z \in X$ such that

$$
\begin{aligned}
&-(\varphi, \partial_t \delta z)_I + a'_u(q,u)(\varphi, \delta z) + (\varphi(T), \delta z(T)) = -a''_{uu}(q,u)(\delta u, \varphi, z) \\
&\quad - a''_{qu}(q,u)(\delta q, \varphi, z) + \int_I J''_1(u)(\delta u, \varphi)\, dt + J''_2(u(T))(\delta u(T), \varphi(T)) \quad \forall \varphi \in X.
\end{aligned} \tag{2.19}
$$

3 Space-Time Finite Element Discretization

In this chapter, we discuss suitable discretizations of the optimization problem ($\mathbb{P}$). To this end, we use Galerkin finite element methods separately in space and time for discretizing the state equation. This allows us to give a natural computable representation of the discrete gradient and Hessian in the same manner as shown in Section 2.5 for the continuous problem. The use of exact discrete derivatives is important for the convergence of the optimization algorithms given in Chapter 4. Moreover, our systematic approach to a priori and a posteriori error estimation (cf. the Chapters 5 and 6) relies on the usage of Galerkin discretizations. In addition, the proposed Galerkin discretizations exhibit the property that the *discretize-then-optimize* approach and the *optimize-then-discretize* approach lead to the same discrete systems.

Section 3.1 is devoted to the semidiscretization in time by *continuous Galerkin* (cG) and *discontinuous Galerkin* (dG) methods. Section 3.2 deals with the space discretization of the semidiscrete problems arising from time discretization. This is done by means of continuous Galerkin finite element methods. The discretization of the control space Q is treated in Section 3.3. Since this part of the discretization depends strongly on the concrete choice of the control space Q, it is kept rather abstract by choosing a finite-dimensional subspace $Q_d \subseteq Q$. Nevertheless, possible concretizations are discussed on the basis of the examples given in Section 2.2. Two concrete variants of the proposed discretizations, which are equivalent to some time stepping schemes, are presented in Section 3.4. Finally, in Section 3.5, we discuss a possibility of numerically proving the crucial property of exactness of the computed discrete derivatives of the reduced cost functional. We close this chapter by substantiating the representation formulas for the derivatives by numerical experiments.

3.1 Time discretization of the state variable

We consider two Galerkin finite element methods for the time discretization of the state equation (2.3) of optimization problem ($\mathbb{P}$). A more detailed introduction and motivation of the concepts presented in the sequel can be found for instance in the textbook of Eriksson, Estep, Hansbo, and Johnson [29].

The first type of discretization we consider, is defined using discontinuous trial and test functions of degree r. We call this method *discontinuous Galerkin method of degree r* or simply dG(r) method; see Section 3.1.1. The second method considered is defined using continuous trial functions of degree r and discontinuous test functions of degree $r-1$. This method is called *continuous Galerkin method of degree r* or cG(r) method; see Section 3.1.2.

To define the proposed semidiscretizations in time, let us partition the time interval $\bar{I} = [0, T]$ as

$$\bar{I} = \{\, 0 \,\} \cup I_1 \cup I_2 \cup \cdots \cup I_{M-1} \cup I_M$$

with left open and right closed subintervals $I_m = (t_{m-1}, t_m]$ of size $k_m := t_m - t_{m-1}$ and time points

$$0 = t_0 < t_1 < \cdots < t_{M-1} < t_M = T.$$

The discretization parameter k is defined as piecewise constant function by setting $k\big|_{I_m} := k_m$ for $m = 1, 2, \ldots, M$. Especially in the theoretical analysis presented in Chapter 5, we use the symbol k also for the maximal length of a subinterval, that is

$$k := \max_{m=1,2,\ldots,M} k_m.$$

By means of the subintervals I_m, we define for $r \in \mathbb{N}_0$ the two semidiscrete spaces X_k^r and $\widetilde{X}_k^r$ by

$$X_k^r := \left\{\, v_k \in C(\bar{I}, H) \;\middle|\; v_k\big|_{I_m} \in \mathcal{P}_r(I_m, V),\ m = 1, 2, \ldots, M \,\right\},$$
$$\widetilde{X}_k^r := \left\{\, v_k \in L^2(I, H) \;\middle|\; v_k\big|_{I_m} \in \mathcal{P}_r(I_m, V),\ m = 1, 2, \ldots, M \text{ and } v_k(0) \in H \,\right\}.$$

Here, $\mathcal{P}_r(I_m, V)$ denotes the space of polynomials up to order r defined on I_m with values in V. Thus, X_k^r consist of functions which are continuous and piecewise polynomials with respect to time. This space is used as trial space in the continuous Galerkin method. The functions in $\widetilde{X}_k^r$ do not have to be continuous, they may have discontinuities at the borders of the subintervals I_m. This space is used as test space in the continuous Galerkin method and as trial and test space in the discontinuous Galerkin method.

Remark 3.1. By construction, we have the inclusion $X_k^r \subseteq X$. However, such an inclusion does not hold for $\widetilde{X}_k^r$ since we have $X \subseteq C(\bar{I}, H)$ and $\widetilde{X}_k^r \not\subseteq C(\bar{I}, H)$.

3.1.1 Discontinuous Galerkin methods

When using dG(r) methods, we define the solution u_k in the space $\widetilde{X}_k^r$ of piecewise polynomials of degree r. To account for discontinuities of functions $v_k \in \widetilde{X}_k^r$ at the time nodes t_m, we introduce the notations

$$v_{k,m}^+ := \lim_{t \downarrow 0} v_k(t_m + t), \quad v_{k,m}^- := \lim_{t \downarrow 0} v_k(t_m - t) = v_k(t_m), \quad \text{and} \quad [v_k]_m := v_{k,m}^+ - v_{k,m}^-.$$

That is, $v_{k,m}^+$ is the limit “from above”, $v_{k,m}^-$ is the limit “from below”, and $[v_k]_m$ is the “jump” in $v_k(t)$ at time t_m. This notation is depicted in Figure 3.1.

Then, the dG(r) semidiscretization of the state equation (2.3) reads: Find for given control $q_k \in Q$ a state $u_k \in \widetilde{X}_k^r$ such that

$$\sum_{m=1}^{M} (\partial_t u_k, \varphi)_{I_m} + a(q_k, u_k)(\varphi) + \sum_{m=0}^{M-1} ([u_k]_m, \varphi_m^+) + (u_{k,0}^-, \varphi_0^-)$$
$$= (f, \varphi)_I + (u_0(q_k), \varphi_0^-) \quad \forall \varphi \in \widetilde{X}_k^r. \quad (3.1)$$

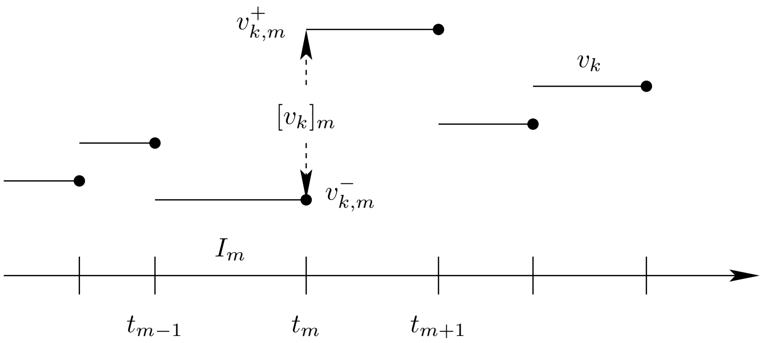


Figure 3.1. Notation for the dG(r) method in the case $r = 0$

Here, $(v, w)_{I_m}$ is defined on I_m correspondingly to the definition of $(v, w)_I$ on I by

$$(v, w)_{I_m} := \int_{I_m} (v(t), w(t))\, dt.$$

Remark 3.2. Many authors prefer the formulation

$$\sum_{m=1}^{M} (\partial_t u_k, \varphi)_{I_m} + a(q_k, u_k)(\varphi) + \sum_{m=2}^{M} ([u_k]_{m-1}, \varphi_{m-1}^+) + (u_{k,0}^+, \varphi_0^+) \\ = (f, \varphi)_I + (u_0(q_k), \varphi_0^+) \quad \forall \varphi \in \widetilde{X}_k^r. \quad (3.2)$$

of the dG(r) method and eliminate the value $v_k(0)$ in the definition of $\widetilde{X}_k^r$. The equivalence of (3.1) and (3.2) can be seen by subtracting the two equations obtaining the implication

$$(u_{k,0}^-, \varphi_0^- - \varphi_0^+) = (u_0(q), \varphi_0^- - \varphi_0^+) \quad \Longrightarrow \quad (3.1) \Leftrightarrow (3.2).$$

The prerequisite is fulfilled either directly due to formulation (3.1) since the terms containing φ_0^- can be separated from the remainder, or by defining the undefined value $u_{k,0}^-$ in the case of considering formulation (3.2). In the major parts of this thesis, we stay at representation (3.1), since it has advantages especially for implementational reasons. This is due to the fact, that then for the dG(0) method the same data structures can be used as for the cG(1) formulation introduced in the next subsection. However, in Chapter 5, we use formulation (3.2) since this representation has advantages for the theoretical analysis of the dG(r) schemes.

In many cases, the unique existence of solutions $u_k \in \widetilde{X}_k^r$ to (3.1) can be obtained for instance by means of the decoupling method shown in Schötzau [73]: When doing so, (3.1) is equivalent to an elliptic system of $r + 1$ equations which can be chosen upper triangular. Under standard assumptions on the semilinear form $a(\cdot, \cdot)(\cdot)$, the unique solvability of these equations is ensured. In the case of a linear state equation, the unique existence of solutions can also be proven by means of Fourier analysis; see Thomée [76].

Then, the semidiscrete optimization problem for the dG(r) time discretization has the form

$$\text{Minimize } J(q_k, u_k) \text{ subject to the state equation (3.1)}, \ (q_k, u_k) \in Q \times \widetilde{X}_k^r. \qquad (\widetilde{\mathbb{P}}_k)$$

We pose the Lagrangian $\widetilde{\mathcal{L}}\colon Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r \to \mathbb{R}$ associated with the dG(r) time discretization of the state equation as

$$\begin{aligned}\widetilde{\mathcal{L}}(q_k, u_k, z_k) \coloneqq J(q_k, u_k) + (f, z_k)_I - \sum_{m=1}^{M} (\partial_t u_k, z_k)_{I_m} \\ - a(q_k, u_k)(z_k) - \sum_{m=0}^{M-1} ([u_k]_m, z_{k,m}^+) + (u_0(q_k) - u_{k,0}^-, z_{k,0}^-).\end{aligned}$$

Remark 3.3. Here, as in the continuous case, the initial condition is coupled to the Lagrangian by $z_{k,0}^- = z_k(0)$. This constitutes no restriction because also in this semidiscrete situation, we would immediately obtain $\tilde{z}_k = z_k(0)$ if we couple the initial condition via a separate Lagrange multiplier $\tilde{z}_k$.

Following the lines of the continuous case, we introduce a semidiscrete solution operator $S_k\colon Q \to \widetilde{X}_k^r$ such that $u_k = S_k(q_k)$ fulfills for $q_k \in Q$ the semidiscrete state equation (3.1). Similar to Section 2.3, we define the semidiscrete reduced cost functional $j_k\colon Q \to \mathbb{R}$ as

$$j_k(q_k) \coloneqq J(q_k, S_k(q_k)),$$

and reformulate the optimization problem $(\widetilde{\mathbb{P}}_k)$ as unconstrained problem:

$$\text{Minimize } j_k(q_k), \quad q_k \in Q. \tag{$\widetilde{\mathbb{P}}_k^{\text{red}}$}$$

With these preliminaries, we obtain similar expressions for the three auxiliary equations in terms of the semidiscrete Lagrangian as stated in Section 2.5. However, the derivation of the explicit representations for the auxiliary equations requires some care due to the special form of the Lagrangian $\widetilde{\mathcal{L}}$ for the dG(r) discretization.

By computing derivatives, we arrive at the following three auxiliary equations expressed in terms of derivatives of the modified Lagrangian $\widetilde{\mathcal{L}}$:

Adjoint for dG(r): Find $z_k \in \widetilde{X}_k^r$ such that

$$\widetilde{\mathcal{L}}'_u(q_k, u_k, z_k)(\varphi) = 0 \quad \forall \varphi \in \widetilde{X}_k^r.$$

Tangent for dG(r): Find $\delta u_k \in \widetilde{X}_k^r$ such that

$$\widetilde{\mathcal{L}}''_{qz}(q_k, u_k, z_k)(\delta q_k, \varphi) + \widetilde{\mathcal{L}}''_{uz}(q_k, u_k, z_k)(\delta u_k, \varphi) = 0 \quad \forall \varphi \in \widetilde{X}_k^r.$$

Additional Adjoint for dG(r): Find $\delta z_k \in \widetilde{X}_k^r$ such that

$$\widetilde{\mathcal{L}}''_{qu}(q_k, u_k, z_k)(\delta q_k, \varphi) + \widetilde{\mathcal{L}}''_{uu}(q_k, u_k, z_k)(\delta u_k, \varphi) + \widetilde{\mathcal{L}}''_{zu}(q_k, u_k, z_k)(\delta z_k, \varphi) = 0 \quad \forall \varphi \in \widetilde{X}_k^r.$$

Calculating the derivatives and applying interval-wise integration by parts to the adjoint equations lead to the explicit form of the auxiliary equations in the case of dG(r) semidiscretization in time:

Adjoint for dG(r): Find $z_k \in \widetilde{X}_k^r$ such that

$$\begin{aligned} -\sum_{m=1}^{M} (\varphi, \partial_t z_k)_{I_m} + a'_u(q_k, u_k)(\varphi, z_k) - \sum_{m=0}^{M-1} (\varphi_m^-, [z_k]_m) \\ + (\varphi_M^-, z_{k,M}^-) = \int_I J'_1(u_k)(\varphi)\, dt + J'_2(u_{k,M}^-)(\varphi_M^-) \quad \forall \varphi \in \widetilde{X}_k^r. \end{aligned} \tag{3.3a}$$

Tangent for dG(r): Find $\delta u_k \in \widetilde{X}_k^r$ such that

$$\begin{aligned} \sum_{m=1}^{M} (\partial_t \delta u_k, \varphi)_{I_m} + a'_u(q_k, u_k)(\delta u_k, \varphi) + \sum_{m=0}^{M-1} ([\delta u_k]_m, \varphi_m^+) + (\delta u_{k,0}^-, \varphi_0^-) \\ = -a'_q(q_k, u_k)(\delta q_k, \varphi) + (u'_0(q_k)(\delta q_k), \varphi_0^-) \quad \forall \varphi \in \widetilde{X}_k^r. \end{aligned} \tag{3.3b}$$

Additional Adjoint for dG(r): Find $\delta z_k \in \widetilde{X}_k^r$ such that

$$\begin{aligned} -\sum_{m=1}^{M} (\varphi, \partial_t \delta z_k)_{I_m} + a'_u(q_k, u_k)(\varphi, \delta z_k) - \sum_{m=0}^{M-1} (\varphi_m^-, [\delta z_k]_m) \\ + (\varphi_M^-, \delta z_{k,M}^-) = -a''_{uu}(q_k, u_k)(\delta u_k, \varphi, z_k) - a''_{qu}(q_k, u_k)(\delta q_k, \varphi, z_k) \\ + \int_I J''_1(u_k)(\delta u_k, \varphi)\, dt + J''_2(u_{k,M}^-)(\delta u_{k,M}^- \varphi_M^-) \quad \forall \varphi \in \widetilde{X}_k^r. \end{aligned} \tag{3.3c}$$

The representation formulas for the first and second derivatives stated in Section 2.5 can now be translated directly to the semidiscrete level: We obtain exactly the same expressions as given in (2.10), (2.20), and (2.21), but with q, u, z, δq, τq, δu, and δz replaced by q_k, u_k, z_k, δq_k, τq_k, δu_k, and δz_k.

3.1.2 Continuous Galerkin methods

Using the semidiscrete spaces defined at the beginning of this section, the cG(r) formulation of the state equation can be stated directly as follows: Find for given control $q_k \in Q$ a state $u_k \in X_k^r$ such that

$$(\partial_t u_k, \varphi)_I + a(q_k, u_k)(\varphi) + (u_k(0), \varphi_0^-) = (f, \varphi)_I + (u_0(q_k), \varphi_0^-) \quad \forall \varphi \in \widetilde{X}_k^{r-1}. \tag{3.4}$$

Remark 3.4. Again, we use here the subscript k to indicate the semidiscretization of the state in time. This eases the notation and is unproblematic since we always tell explicitly which time discretization is considered.

Remark 3.5. In the formulation of the cG(r) method, the polynomial degree of the test functions is reduced by one compared to the degree used for the trial functions. This is necessary to obtain a quadratical system of equations since we have to compensate the additional degrees of freedom in $\widetilde{X}_k^r$ due to the allowed discontinuity of its elements.

The corresponding semidiscretized optimization problem reads as follows:

$$\text{Minimize } J(q_k, u_k) \text{ subject to the state equation (3.4)}, \ (q_k, u_k) \in Q \times X_k^r. \tag{$\mathbb{P}_k$}$$

Since the state equation semidiscretized by the cG(r) method has the same form as in the continuous setting, the corresponding Lagrangian is analogously defined on $Q \times X_k^r \times \widetilde{X}_k^{r-1}$ by

$$\mathcal{L}(q_k, u_k, z_k) := J(q_k, u_k) + (f - \partial_t u_k, z_k)_I - a(q_k, u_k)(z_k) + (u_0(q_k) - u_{k,0}, z_{k,0}^-).$$

To ease the notation, we use here additionally to the notation introduced for the dG(r) discretization the abbreviation $v_{k,m} := v_k(t_m)$.

Remark 3.6. The coupling of the initial condition is done here like it was done for the dG(r) discretization and the statement of Remark 3.3 applies also in this case.

Using the same notation as in the previous subsection, we define the semidiscrete reduced cost functional $j_k \colon Q \to \mathbb{R}$ as

$$j_k(q_k) := J(q_k, S_k(q_k)),$$

and reformulate the optimization problem ($\mathbb{P}_k$) as unconstrained problem:

$$\text{Minimize } j_k(q_k), \quad q_k \in Q. \tag{$\mathbb{P}_k^{\text{red}}$}$$

We again base on the representation of the auxiliary equations in terms of derivatives of the Lagrangian, which are formally identical to the continuous case considered in Section 2.5:

Adjoint for cG(r): Find $z_k \in \widetilde{X}_k^{r-1}$ such that

$$\mathcal{L}'_u(q_k, u_k, z_k)(\varphi) = 0 \quad \forall \varphi \in X_k^r.$$

Tangent for cG(r): Find $\delta u_k \in X_k^r$ such that

$$\mathcal{L}''_{qz}(q_k, u_k, z_k)(\delta q_k, \varphi) + \mathcal{L}''_{uz}(q_k, u_k, z_k)(\delta u_k, \varphi) = 0 \quad \forall \varphi \in \widetilde{X}_k^{r-1}.$$

Additional Adjoint for cG(r): Find $\delta z_k \in \widetilde{X}_k^{r-1}$ such that

$$\mathcal{L}''_{qu}(q_k, u_k, z_k)(\delta q_k, \varphi) + \mathcal{L}''_{uu}(q_k, u_k, z_k)(\delta u_k, \varphi) + \mathcal{L}''_{zu}(q_k, u_k, z_k)(\delta z_k, \varphi) = 0 \quad \forall \varphi \in X_k^r.$$

Proceeding as for the dG(r) discretization leads to the explicit form of the three auxiliary equations in the setting of the cG(r) semidiscretization in time:

Adjoint for cG(r): Find $z_k \in \widetilde{X}_k^{r-1}$ such that

$$\begin{aligned} -\sum_{m=1}^{M} (\varphi, \partial_t z_k)_{I_m} + a'_u(q_k, u_k)(\varphi, z_k) - \sum_{m=0}^{M-1} (\varphi_m, [z_k]_m) \\ + (\varphi_M, z_{k,M}^-) = \int_I J'_1(u_k)(\varphi)\, dt + J'_2(u_{k,M})(\varphi_M) \quad \forall \varphi \in X_k^r. \end{aligned} \tag{3.5a}$$

Tangent for cG(r): Find $\delta u_k \in X_k^r$ such that

$$\begin{aligned}(\partial_t \delta u_k, \varphi)_I + a_u'(q_k, u_k)(\delta u_k, \varphi) + (\delta u_{k,0}, \varphi_0^-)& \\ = -a_q'(q_k, u_k)(\delta q_k, \varphi) + (u_0'(q_k)(\delta q_k), \varphi_0^-) \quad \forall \varphi \in \widetilde{X}_k^{r-1}.& \qquad (3.5\text{b})\end{aligned}$$

Additional Adjoint for cG(r): Find $\delta z_k \in \widetilde{X}_k^{r-1}$ such that

$$\begin{aligned}-\sum_{m=1}^{M} (\varphi, \partial_t \delta z_k)_{I_m} + a_u'(q_k, u_k)(\varphi, \delta z_k) - \sum_{m=0}^{M-1} (\varphi_m, [\delta z_k]_m)& \\ + (\varphi_M, \delta z_{k,M}^-) = -a_{uu}''(q_k, u_k)(\delta u_k, \varphi, z_k) - a_{qu}''(q_k, u_k)(\delta q_k, \varphi, z_k)& \\ + \int_I J_1''(u_k)(\delta u_k, \varphi)\, dt + J_2''(u_{k,M})(\delta u_{k,M}, \varphi_M) \quad \forall \varphi \in X_k^r.& \qquad (3.5\text{c})\end{aligned}$$

By inspection of the auxiliary equations of the dG(r) and cG(r) methods, one may recognize that the adjoint equation for dG(r) (3.3a) and the adjoint equation for cG(r) (3.5a) are quite similar. However, the main difference lies in the selection of the test space. In contrast to the dG(r) method, where the test functions are discontinuous in time, the cG(r) method uses continuous test functions. This fact has to be incorporated especially when computing the concrete time stepping schemes as done in Section 3.4.

As for the dG(r) discretization, the representation for the derivatives of the reduced functional derived in Section 2.5 carries over from the continuous level to the level of cG(r) semidiscretization just by adding the subscript k to all arising variables.

3.2 Space discretization of the state variable

Up to now we have considered only semidiscretization in time, that is, the introduced spaces X_k^r and $\widetilde{X}_k^r$ still contain the continuous spatial space V in their definitions. The current section is devoted to the space discretization of the semidiscrete equations from the previous section. This is done by choosing finite-dimensional subspaces $V_h^s \subseteq V$ consisting of finite elements up to order s. Moreover, we allow different space discretizations in each time interval I_m. Details on this construction are given in Section 3.2.2.

3.2.1 Triangulations and finite element spaces

In this subsection, we describe the finite element triangulations of the computational domain $\Omega \subseteq \mathbb{R}^n$ for $n \in \{2, 3\}$ and the construction of the corresponding finite element spaces. For simplicity, we assume the boundary $\partial\Omega$ to be polygonal. The case of non-polygonally bounded domains is not considered here; details on this can be found for instance in Braess [17].

Depending on the dimension, the domain Ω is partitioned into open quadrilaterals or hexahedrals K—in the sequel denoted as *cells*. The resulting triangulation is denoted by $\mathcal{T}_h = \{ K \}$.

The mesh parameter h is defined as a cellwise constant function describing the diameter of the cell by

$$h\big|_K := h_K := \operatorname{diam}(K).$$

Additionally, we use the symbol h also for the maximal cell diameter, that is

$$h := \max_{K \in \mathcal{T}_h} h_K.$$

The maximal straight parts which make up the boundary ∂K of a cell K are called *faces*.

Following the literature as Ciarlet [22] or Braess [17], we propose the following definition:

Definition 3.1 (Regularity)**.** A triangulation $\mathcal{T}_h = \{\, K \,\}$ is called regular if the following conditions are fulfilled:

(i) $\bar{\Omega} = \bigcup_{K \in \mathcal{T}_h} \bar{K}$.

(ii) For each distinct cells $K_1, K_2 \in \mathcal{T}_h$, one has $K_1 \cap K_2 = \emptyset$.

(iii) Any face of any cell K_1 in the triangulation $\mathcal{T}_h$ is either a subset of the boundary $\partial\Omega$ or a face of another cell $K_2 \in \mathcal{T}_h$.

To allow local mesh refinement without using connecting elements, we weaken the last condition of Definition 3.1 and introduce *hanging nodes*: Cells are allowed to have nodes which lie on midpoints of faces of neighboring cells. At most one hanging node is permitted on each face (cf. Figure 3.2).

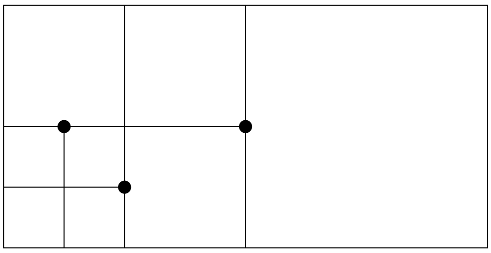


Figure 3.2. A two-dimensional triangulation with three hanging nodes

In addition, we require the triangulation $\mathcal{T}_h$ to be organized in a patch-wise manner. This means that it results from a global uniform refinement of a coarser triangulation $\mathcal{T}_{2h}$. By a *patch* of cells, we denote a group of four cells in $\mathcal{T}_h$ which results from a refinement of a common coarser cell in $\mathcal{T}_{2h}$. We make use of this construction in the context of a posteriori error estimation in Section 6.4. An example of this construction is given in Figure 3.3.

Following Ciarlet [22], Brenner and Scott [18], or Johnson [48], we construct continuous V-conforming finite element spaces V_h^s by

$$V_h^s := \left\{\, v \in C(\bar{\Omega}) \cap V \;\middle|\; v\big|_K \in \mathcal{Q}_s(K),\ K \in \mathcal{T}_h \,\right\},$$

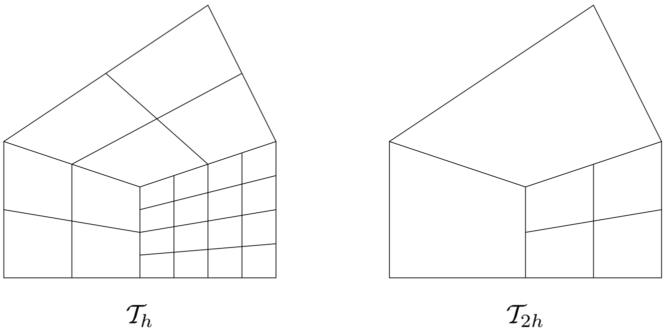


Figure 3.3. A two-dimensional triangulation with patch structure and hanging nodes (left) resulting from a coarser regular triangulation (right) by global uniform refinement

where $\mathcal{Q}_s(K)$ denotes a suitable space of polynomial-like functions on the cell $K \in \mathcal{T}_h$. To define the spaces $\mathcal{Q}_s(K)$, we introduce the polynomial spaces $\hat{\mathcal{Q}}_s(\hat{K})$ on the reference cell $\hat{K} = (0,1)^n$ given by

$$\hat{\mathcal{Q}}_s(\hat{K}) := \operatorname{span}\left\{ \prod_{i=1}^{n} x_i^{\alpha_i} \;\middle|\; \alpha_i \in \{ 0, 1, \dots, s \} \right\}.$$

Then, the spaces $\mathcal{Q}_s(K)$ are obtained using the transformations $T_K \colon \hat{K} \to K$ (cf. Figure 3.4) as

$$\mathcal{Q}_s(K) = \left\{ v \colon K \to \mathbb{R} \;\middle|\; v \circ T_K \in \hat{\mathcal{Q}}_s(\hat{K}) \right\}.$$

If we additionally have $T_K \in \hat{\mathcal{Q}}_s(\hat{K})^n$, then we call the resulting finite element space isoparametric.

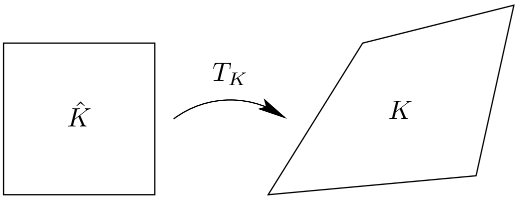


Figure 3.4. Transformation T_K from the reference cell $\hat{K}$ to a computational cell K

The case of hanging nodes requires some additional remarks: There are no degrees of freedom corresponding to these irregular nodes. The values of the finite element functions at such nodes are determined by point-wise interpolation. This implies continuity and therefore global conformity. For details on the implementation see for instance Carey and Oden [19].

By means of Cea's lemma, it is possible to estimate the approximation error of finite elements by an interpolation error. The interpolation error of the point-wise interpolation for continuous functions $i_h \colon C(\bar{\Omega}) \to V_h^s$ can be estimated by the following lemma:

Lemma 3.1. *Let $\mathcal{T}_h$ be a regular triangulation of the domain Ω and V_h^s be a space of (isoparametric) finite elements of order s. Then, there exists a constant C depending only on Ω and s*

such that there holds for each cell $K \in \mathcal{T}_h$ and $u \in H^m(\Omega)$ with $2 \le m \le s+1$ and $0 \le k \le m$:

$$|u - i_h u|_{H^k(K)} \le C \frac{h_K^m}{\rho_K^k} |u|_{H^m(K)}.$$

Here, ρ_K denotes the diameter of the biggest ball inscribed in the cell K.

Proof. The proof is done using the Bramble-Hilbert lemma and can be found for example in Braess [17]. □

Remark 3.7. A family $\{ \mathcal{T}_h \mid h \downarrow 0 \}$ of regular triangulations is called to be quasi uniform, if there exists a constant θ such that for all $K \in \bigcup_h \mathcal{T}_h$ the condition

$$\frac{h_K}{\rho_K} \le \theta$$

is fulfilled. Under the assumption of quasi uniformity, the assertion of Lemma 3.1 can be formulated as

$$|u - i_h u|_{H^k(\Omega)} \le C h^{m-k} |u|_{H^m(\Omega)}.$$

Especially in Chapter 5, we use this and related estimates for deriving a priori convergence estimates for the space-time finite element discretization of the state equation formulated in the following subsection.

3.2.2 Discretization on dynamic meshes

In this subsection, we construct the fully discrete versions of the semidiscrete equations derived in Section 3.1 and introduce also the concept of dynamic meshes. This is done in the same way as in Schmich and Vexler [72]. We allow dynamic mesh changes in time whereas the time steps k_m are kept constant in space. Therefore, we associate with each time point t_m a triangulation $\mathcal{T}_h^m$ and a corresponding finite element space $V_h^{s,m} \subseteq V$ which is used as spatial trial and test space in the adjacent time interval I_m.

By means of this choice, we define the fully discrete space-time finite element space

$$\widetilde{X}_{k,h}^{r,s} := \left\{ v_{kh} \in L^2(I,H) \;\middle|\; v_{kh}\big|_{I_m} \in \mathcal{P}_r(I_m, V_h^{s,m}),\ m = 1,2,\dots,M \text{ and } v_{kh}(0) \in V_h^{s,0} \right\}.$$

Due to the conformity of $V_h^{s,m}$, we have the inclusion $\widetilde{X}_{k,h}^{r,s} \subseteq \widetilde{X}_k^r$.

Thus, the so-called cG(s)dG(r) discretization of the state equation (3.6) is obtained from the dG(r) semidiscretization by adding the supplementary index h to the variables and by replacing the semidiscrete space $\widetilde{X}_k^r$ by $\widetilde{X}_{k,h}^{r,s}$: Find for given control $q_{kh} \in Q$ a state $u_{kh} \in \widetilde{X}_{k,h}^{r,s}$ such that

$$\sum_{m=1}^{M} (\partial_t u_{kh}, \varphi)_{I_m} + a(q_{kh}, u_{kh})(\varphi) + \sum_{m=0}^{M-1} ([u_{kh}]_m, \varphi_m^+) + (u_{kh,0}^-, \varphi_0^-) \\ = (f,\varphi)_I + (u_0(q_{kh}), \varphi_0^-) \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s}. \quad (3.6)$$

Remark 3.8. The notation cG(s)dG(r) (and also the notation cG(s)cG(r) used below) is taken from Eriksson, Estep, Hansbo, and Johnson [29] and describes continuous discretizations in space of order s combined with (dis-)continuous discretizations in time of order r.

The semidiscrete optimization problem for the cG(s)dG(r) discretization has the form

$$\text{Minimize } J(q_{kh}, u_{kh}) \text{ subject to the state equation (3.6)}, \ (q_{kh}, u_{kh}) \in Q \times \widetilde{X}^{r,s}_{k,h}. \qquad (\widetilde{\mathbb{P}}_{kh})$$

Utilizing the reduced cost functional $j_{kh} \colon Q \to \mathbb{R}$ defined by means of the discrete solution operator $S_{kh} \colon Q \to \widetilde{X}^{r,s}_{k,h}$, we obtain the reduced optimization problem

$$\text{Minimize } j_{kh}(q_{kh}), \quad q_{kh} \in Q. \qquad (\widetilde{\mathbb{P}}^{\text{red}}_{kh})$$

By following the recipe of adding the index h, we also obtain the fully discrete versions of the auxiliary equations (3.3). We skip here the repetition of these equations since we give a realization for a concrete choice of discretizations of the control variable in Section 3.4.

The formulation of the cG(s)cG(r) discretization is more involved since we have to ensure global continuity in time of functions in the trial space. To this end, we describe an approach similar to the one presented in Becker [2]: Let $\{ \tau_0, \tau_1, \dots, \tau_r \}$ be a basis of $\mathcal{P}_r(I_m, \mathbb{R})$ with the property

$$\tau_0(t_{m-1}) = 1, \quad \tau_0(t_m) = 0, \quad \text{and} \quad \tau_i(t_{m-1}) = 0, \ i = 1, 2, \dots, r.$$

By means of the spaces $X^{r,s,m}_{k,h} \subseteq \mathcal{P}_r(I_m, V)$, given as

$$X^{r,s,m}_{k,h} := \operatorname{span}\left\{ \tau_i v_i \;\middle|\; v_0 \in V^{s,m-1}_h, \ v_i \in V^{s,m}_h, \ i = 1, 2, \dots, r \right\},$$

we define the trial space for the cG(s)cG(r) formulation by

$$X^{r,s}_{k,h} := \left\{ v_{kh} \in C(\bar{I}, H) \;\middle|\; v_{kh}\big|_{I_m} \in X^{r,s,m}_{k,h}, \ m = 1, 2, \dots, M \right\}.$$

The definition of $X^{r,s,m}_{k,h}$ ensures the continuity in time of all functions in $X^{r,s}_{k,h}$. This is due to the fact that the spatial degrees of freedom which vanish when stepping from $V^{s,m-1}_h$ to $V^{s,m}_h$ are only coupled to the temporal basis function τ_0 which is zero at the right boundary of the subinterval I_m. Vice versa, the degrees of freedom in $V^{s,m}_h$ which appear when coming from $V^{s,m-1}_h$ are only coupled to the basis functions τ_i, $i = 1, 2, \dots, r$ which are zero at the left boundary of I_m.

Remark 3.9. In the case $r = 1$, we can choose the Lagrange basis of $\mathcal{P}_1(I_m, \mathbb{R})$ given by

$$\tau_0(t) = \frac{t_m - t}{k_m} \quad \text{and} \quad \tau_1(t) = \frac{t - t_{m-1}}{k_m}.$$

This basis fulfills the proposed requirements.

Remark 3.10. If all spatial triangulations and consequently all spatial finite element spaces are the same (that is $V_h^{s,m} = V_h^s$ for $m = 0, 1, \dots, M$), we have the identity

$$X_{k,h}^{r,s,m} = \mathcal{P}_r(I_m, V_h^s),$$

and thus the definition of the space $X_{k,h}^{r,s}$ coincides with the more familiar one

$$X_{k,h}^{r,s} = \left\{ v_{kh} \in C(\bar{I}, H) \;\middle|\; v_{kh}\big|_{I_m} \in \mathcal{P}_r(I_m, V_h^s),\ m = 1, 2, \dots, M \right\}$$

of the usual trial space for the cG(s)cG(r) discretization.

Using these spaces, we formulate the cG(s)cG(r) discretized state equation as: Find for given control $q_{kh} \in Q$ a state $u_{kh} \in X_{k,h}^{r,s}$ such that

$$(\partial_t u_{kh}, \varphi)_I + a(q_{kh}, u_{kh})(\varphi) + (u_{kh}(0), \varphi_0^-) = (f, \varphi)_I + (u_0(q_{kh}), \varphi_0^-) \quad \forall \varphi \in \widetilde{X}_{k,h}^{r-1,s}. \tag{3.7}$$

Similar to the procedure mentioned during the presentation of the cG(s)dG(r) discretization, the fully discrete state equation (3.7) is obtained by adding the index h to the variables and by replacing the spaces X_k^r and $\widetilde{X}_k^r$ by $X_{k,h}^{r,s}$ and $\widetilde{X}_{k,h}^{r,s}$, respectively. The three auxiliary equations are obtained from the semidiscrete ones (3.5) by doing so, too. A concrete realization of these equations is presented in Section 3.4 after discussing possible discretizations of the control variable. Also the expressions for the derivatives of the discrete reduced cost functional j_{kh} are directly obtained from the continuous level by replacing the continuous solutions by the discrete ones.

3.3 Discretization of the control variable

In this section, we discuss possible discretizations of the control space which was kept undiscretized up to now. Before doing so, we remark that it is possible to solve optimization problems numerically even if an infinite-dimensional control space is not discretized explicitly. Details of this approach for linear-quadratic elliptic optimization problems can be found in Hinze [44]. Since this approach exhibits computational difficulties in situations where the control enters the state equation nonlinearly, we stay here with the classical approach of discretizing the controls and present an a priori analysis for it in Chapter 5.

We recall that the Hilbert space of controls Q was in general characterized by the inclusion

$$Q \subseteq L^2(I, R)$$

with a spatial Hilbert space R. In the sequel, we describe possible Galerkin-type discretizations of the control spaces Q chosen in the Examples 2.1, 2.2, 2.3, and 2.4 given in Section 2.2.

Example 3.1 (Concerning Example 2.1)**.** Here, we have

$$R = L^2(\varOmega) \quad \text{and} \quad Q = L^2(I, R) = L^2(I, L^2(\varOmega)).$$

For the time discretization of Q, we choose the dG(r_d) method as presented in Section 3.1.1 for the state space X, but with polynomials of order $r_d \in \mathbb{N}_0$. In general, we allow the time discretization of Q to be coarser than the one for X. That is, we always enforce the set of time points used for the control discretization to be a subset of the time points of the state discretization.

The spatial part of Q is either discretized as the state space by a cG(s_d) method, or we discretize it by cellwise constant functions. We call this discretization dG(0) method as for the time discretization. Our theoretical results to be developed in Chapter 5 also include the possibility of a coarser mesh with cell size h_d which is again constructed by coarsening of the triangulation used for discretizing the states. However, for computational reasons, in the presented numerical computations we always use $h_d = h$ and $s_d = s$.

Summarizing, we choose a finite-dimensional subset $Q_d \subseteq Q$ defined by the cG(s_d)dG(r_d) or by the dG(0)dG(r_d) method. These discretizations lead in combination with the discretization of the state space to the fully discrete optimization problem.

Example 3.2 (Concerning Example 2.2)**.** Here, we have

$$R = L^2(\partial\Omega) \quad \text{and} \quad Q = L^2(I, R) = L^2(I, L^2(\partial\Omega)).$$

For the time discretization of Q, we proceed as in Example 3.1. The space discretization of R is done by traces $\gamma(v_h) \in C(\partial\Omega)$ of functions $v_h \in V_h^{s_d}$ constructed by means of a cG(s_d) finite element discretization on Ω.

Example 3.3 (Concerning Example 2.3)**.** Here, we have

$$R = L^2(\Omega) \quad \text{and} \quad Q = \mathcal{P}_0(\bar{I}, R) = \mathcal{P}_0(\bar{I}, L^2(\Omega)).$$

Since here the temporal component of Q is already discrete, it needs not to be discretized. The spatial discretization of R can be done as described in Example 3.1.

Example 3.4 (Concerning Example 2.4)**.** Here, we have

$$R = \mathbb{R} \quad \text{and} \quad Q = \mathcal{P}_0(\bar{I}, R) = \mathcal{P}_0(\bar{I}, \mathbb{R}).$$

Thus, the control space Q is already finite-dimensional and we choose $Q_d = Q$.

Since all presented choices of Q_d lead to conforming discretizations of Q, the discrete state and auxiliary equations stated in the section before can be transfered directly to the level of discrete controls. The solution variables as q and u on this level are denoted by q_{khd} and u_{khd}, respectively. We abbreviate the indices "khd", which symbolize the space and time discretization of the state and the discretization of the control, by "σ".

Finally, we state the fully discrete optimization problem with cG(s)dG(r) discretization of the state space and discretized control space Q_d as

$$\text{Minimize } J(q_\sigma, u_\sigma) \text{ subject to the state equation (3.6), } (q_\sigma, u_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{r,s}. \qquad (\widetilde{\mathbb{P}}_\sigma)$$

Especially for the a priori analysis given in Section 5.3, we make use of this problem for the precise formulation of the error estimates to be derived.

3.4 Time stepping schemes

In what follows, we present one concrete time-stepping scheme for the cG(s)dG(r) and the cG(s)cG(r) discretizations both combined with the cG(s)dG(r) discretization of the control space. These schemes correspond to the widely-used implicit Euler and Crank-Nicolson schemes.

To solve the individual time steps of the discrete schemes presented in the sequel, we employ a Newton solver for treating the nonlinearities. The arising linear subproblems are then solved by a multigrid iteration with an ILU decomposition as smoother. Details on the construction of a multigrid solver on adaptively refined meshes are given in Becker and Braack [4].

3.4.1 Implicit Euler scheme

To obtain the well-known implicit Euler scheme as a special case of the dG(r) time discretization, we choose $r = 0$ and approximate the temporal integrals arising by the box rule. We define for brevity

$$\begin{aligned} Q_m &:= q^-_{\sigma,m}, & U_m &:= u^-_{\sigma,m}, & Z_m &:= z^-_{\sigma,m}, \\ \Delta Q_m &:= \delta q^-_{\sigma,m}, & \Delta U_m &:= \delta u^-_{\sigma,m}, & \Delta Z_m &:= \delta z^-_{\sigma,m} \end{aligned}$$

for $m = 0, 1, \ldots, M$. With this, we obtain the following schemes for the cG(s)dG(0)-discretized state and auxiliary equations combined with a cG(s)dG(0) discretization of the control space, which all should be fulfilled for every $\psi \in V_h^{s,m}$:

State for cG(s)dG(0):

$m = 0$:

$$(U_0, \psi) = (u_0(Q_0), \psi)$$

$m = 1, 2, \ldots, M$:

$$(U_m, \psi) + k_m \bar{a}(Q_m, U_m)(\psi) = (U_{m-1}, \psi) + k_m(f(t_m), \psi)$$

Adjoint for cG(s)dG(0):

$m = M$:

$$(\psi, Z_M) + k_M \bar{a}'_u(Q_M, U_M)(\psi, Z_M) = k_M J'_1(U_M)(\psi) + J'_2(U_M)(\psi)$$

$m = M-1, M-2, \ldots, 1$:

$$(\psi, Z_m) + k_m \bar{a}'_u(Q_m, U_m)(\psi, Z_m) = (\psi, Z_{m+1}) + k_m J'_1(U_m)(\psi)$$

$m = 0$:

$$(\psi, Z_0) = (\psi, Z_1)$$

Tangent for cG(s)dG(0):

***m* = 0:**

$$(\mathit{\Delta}U_0, \psi) = (u_0'(Q_0)(\mathit{\Delta}Q_0), \psi)$$

***m* = 1, 2, …, *M*:**

$$(\mathit{\Delta}U_m, \psi) + k_m\bar{a}_u'(Q_m, U_m)(\mathit{\Delta}U_m, \psi) = (\mathit{\Delta}U_{m-1}, \psi) - k_m\bar{a}_q'(Q_m, U_m)(\mathit{\Delta}Q_m, \psi)$$

Additional Adjoint for cG(s)dG(0):

***m* = *M*:**

$$\begin{aligned}(\psi, \mathit{\Delta}Z_M) + k_M\bar{a}_u'(Q_M, U_M)(\psi, \mathit{\Delta}Z_M) = \\ - k_M\bar{a}_{uu}''(Q_M, U_M)(\mathit{\Delta}U_M, \psi, Z_M) - k_M\bar{a}_{qu}''(Q_M, U_M)(\mathit{\Delta}Q_M, \psi, Z_M) \\ + k_M J_1''(U_M)(\mathit{\Delta}U_M, \psi) + J_2''(U_M)(\mathit{\Delta}U_M, \psi)\end{aligned}$$

***m* = *M* − 1, *M* − 2, …, 1:**

$$\begin{aligned}(\psi, \mathit{\Delta}Z_m) + k_m\bar{a}_u'(Q_m, U_m)(\psi, \mathit{\Delta}Z_m) = \\ (\psi, \mathit{\Delta}Z_{m+1}) - k_m\bar{a}_{uu}''(Q_m, U_m)(\mathit{\Delta}U_m, \psi, Z_m) - k_m\bar{a}_{qu}''(Q_m, U_m)(\mathit{\Delta}Q_m, \psi, Z_m) \\ + k_m J_1''(U_m)(\mathit{\Delta}U_m, \psi)\end{aligned}$$

***m* = 0:**

$$(\psi, \mathit{\Delta}Z_0) = (\psi, \mathit{\Delta}Z_1)$$

The implicit Euler scheme is known to be a first order strongly A-stable method. The resulting schemes for the auxiliary equations have basically the same structure and lead consequently to a first order approximation in time, too. However, the precise a priori error analysis for the optimization problem requires more care and depends on the given structure of the problem under consideration; see Chapter 5 for the analysis in the case of a linear-quadratic optimal control problem. Furthermore, the approximation of the integrals by the box rule has disadvantages especially in the case of long time integration. In Eriksson and Johnson [32], the authors demonstrate this actuality in the case of a scalar linear ordinary differential equation. In such cases, the utilization of a quadrature rule of higher order is profitable.

Since the obtained time stepping scheme for the adjoint equation is identical to the implicit Euler scheme applied to the continuous adjoint equation, we note that even when using numerical integration, the implicit Euler scheme (as well as all dG(r) schemes) exhibits the property that the presented *discretize-then-optimize* approach leads to the same time stepping scheme as the *optimize-then-discretize* approach.

However, this is only the case when the time stepping equation is formulated as

$$(U_m, \psi) + k_m\bar{a}(Q_m, U_m)(\psi) = (U_{m-1}, \psi) + k_m(f(t_m), \psi).$$

As shown, this leads to

$$(\psi, Z_m) + k_m \bar{a}'_u(Q_m, U_m)(\psi, Z_m) = (\psi, Z_{m+1}) + k_m J'_1(U_m)(\psi) \tag{3.8}$$

as adjoint time stepping scheme. However, when using the equivalent standard formulation

$$\frac{1}{k_m}(U_m, \psi) + \bar{a}(Q_m, U_m)(\psi) = \frac{1}{k_m}(U_{m-1}, \psi) + (f(t_m), \psi)$$

of the implicit Euler scheme for the state equation, the discretize-then-optimize-approach produces the adjoint scheme

$$\frac{1}{k_m}(\psi, \widetilde{Z}_m) + \bar{a}'_u(Q_m, U_m)(\psi, \widetilde{Z}_m) = \frac{1}{k_{m+1}}(\psi, \widetilde{Z}_{m+1}) + k_m J'_1(U_m)(\psi).$$

In contrast, the optimize-then-discretize approach, that is the application of this variant of the implicit Euler scheme to the continuous adjoint equation gives

$$\frac{1}{k_m}(\psi, Z_m) + \bar{a}'_u(Q_m, U_m)(\psi, Z_m) = \frac{1}{k_m}(\psi, Z_{m+1}) + J'_1(U_m)(\psi),$$

which is obviously equivalent to (3.8), the formulation obtained by the Galerkin ansatz. Hence, the adjoint $\widetilde{Z}$ (obtained by the discretize-then-optimize-approach) is related to the adjoint Z (obtained either by the optimize-then-discretize approach or by the Galerkin ansatz) via

$$\widetilde{Z}_m = k_m Z_m.$$

Due to the construction of Z, it is an approximation of the continuous adjoint state z. Thus, we have $Z \to z$ for $k \to 0$ (cf. Section 5.3.1 for the precise formulation). Consequently, we obtain $\widetilde{Z} = kZ \to 0$ for $k \to 0$. Although the usage of $\widetilde{Z}$ also leads to correct computations of the discrete derivatives if the representations of j'_{kh} and j''_{kh} are adjusted correspondingly, the behavior of $\widetilde{Z}$ for small k is unfavorable from a computational point of view.

3.4.2 Crank-Nicolson scheme

The Crank-Nicolson scheme can be obtained in the context of the cG(r) time discretizations by choosing $r = 1$ and approximating the temporal integrals arising by the trapezoidal rule. Using the representation of the Crank-Nicolson scheme as a cG(r) scheme allows us to give directly the concrete form of the auxiliary equations leading to the exact computation of the discrete gradient and Hessian.

We set here for brevity

$$\begin{aligned} Q_m &:= q^-_{\sigma,m}, & U_m &:= u_{\sigma,m}, & Z_m &:= z^-_{\sigma,m}, \\ \mathit{\Delta} Q_m &:= \delta q^-_{\sigma,m}, & \mathit{\Delta} U_m &:= \delta u_{\sigma,m}, & \mathit{\Delta} Z_m &:= \delta z^-_{\sigma,m} \end{aligned}$$

for $m = 0, 1, \ldots, M$. With this, we obtain the following schemes for the cG(s)cG(1)-discretized state and auxiliary equations combined with a cG(s)dG(0) discretization of the control space, which all should be fulfilled for every $\psi \in V_h^{s,m}$:

State for cG(s)cG(1):

$\boldsymbol{m = 0}$**:**

$$(U_0,\psi) = (u_0(Q_0),\psi)$$

$\boldsymbol{m = 1, 2, \ldots, M}$**:**

$$\begin{aligned}(U_m,\psi) + \frac{k_m}{2}\bar{a}(Q_m,U_m)(\psi) = (U_{m-1},\psi)&\\ - \frac{k_m}{2}\bar{a}(Q_m,U_{m-1})(\psi) + \frac{k_m}{2}(f(t_{m-1}),\psi) + \frac{k_m}{2}(f(t_m),\psi)&\end{aligned}$$

Adjoint for cG(s)cG(1):

$\boldsymbol{m = M}$**:**

$$(\psi,Z_M) + \frac{k_M}{2}\bar{a}'_u(Q_M,U_M)(\psi,Z_M) = \frac{k_M}{2}J'_1(U_M)(\psi) + J'_2(U_M)(\psi)$$

$\boldsymbol{m = M-1, M-2, \ldots, 1}$**:**

$$\begin{aligned}(\psi,Z_m) + \frac{k_m}{2}\bar{a}'_u(Q_m,U_m)(\psi,Z_m) = (\psi,Z_{m+1})&\\ - \frac{k_{m+1}}{2}\bar{a}'_u(Q_{m+1},U_m)(\psi,Z_{m+1}) + \frac{k_m+k_{m+1}}{2}J'_1(U_m)(\psi)&\end{aligned}$$

$\boldsymbol{m = 0}$**:**

$$(\psi,Z_0) = (\psi,Z_1) - \frac{k_1}{2}\bar{a}'_u(Q_1,U_0)(\psi,Z_1) + \frac{k_1}{2}J'_1(U_0)(\psi)$$

Tangent for cG(s)cG(1):

$\boldsymbol{m = 0}$**:**

$$(\mathit{\Delta} U_0,\psi) = (u'_0(Q_0)(\mathit{\Delta} Q_0),\psi)$$

$\boldsymbol{m = 1, 2, \ldots, M}$**:**

$$\begin{aligned}(\mathit{\Delta} U_m,\psi) + \frac{k_m}{2}\bar{a}'_u(Q_m,U_m)(\mathit{\Delta} U_m,\psi) =&\\ (\mathit{\Delta} U_{m-1},\psi) - \frac{k_m}{2}\bar{a}'_u(Q_m,U_{m-1})(\mathit{\Delta} U_{m-1},\psi)&\\ - \frac{k_m}{2}\bar{a}'_q(Q_m,U_{m-1})(\mathit{\Delta} Q_m,\psi) - \frac{k_m}{2}\bar{a}'_q(Q_m,U_m)(\mathit{\Delta} Q_m,\psi)&\end{aligned}$$

Additional Adjoint for cG(s)cG(1):

***m* = *M*:**

$$
\begin{aligned}
(\psi, \mathit{\Delta} Z_M) + \frac{k_M}{2} \bar{a}'_u(Q_M, U_M)(\psi, \mathit{\Delta} Z_M) &= \\
&- \frac{k_M}{2} \bar{a}'_{uu}(Q_M, U_M)(\mathit{\Delta} U_M, \psi, Z_M) - \frac{k_M}{2} \bar{a}'_{qu}(Q_M, U_M)(\mathit{\Delta} Q_M, \psi, Z_M) \\
&+ \frac{k_M}{2} J''_1(U_M)(\mathit{\Delta} U_M, \psi) + J''_2(U_M)(\mathit{\Delta} U_M, \psi)
\end{aligned}
$$

***m* = *M* − 1, *M* − 2, . . . , 1:**

$$
\begin{aligned}
(\psi, \mathit{\Delta} Z_m) + \frac{k_m}{2} \bar{a}'_u(Q_m, U_m)(\psi, \mathit{\Delta} Z_m) &= (\psi, \mathit{\Delta} Z_{m+1}) \\
&- \frac{k_{m+1}}{2} \bar{a}'_u(Q_{m+1}, U_m)(\psi, \mathit{\Delta} Z_{m+1}) - \frac{k_m}{2} \bar{a}''_{uu}(Q_m, U_m)(\mathit{\Delta} U_m, \psi, Z_m) \\
&- \frac{k_{m+1}}{2} \bar{a}''_{uu}(Q_{m+1}, U_m)(\mathit{\Delta} U_m, \psi, Z_{m+1}) - \frac{k_m}{2} \bar{a}''_{qu}(Q_m, U_m)(\mathit{\Delta} Q_m, \psi, Z_m) \\
&- \frac{k_{m+1}}{2} \bar{a}''_{qu}(Q_{m+1}, U_m)(\mathit{\Delta} Q_m, \psi, Z_{m+1}) + \frac{k_m + k_{m+1}}{2} J''_1(U_m)(\mathit{\Delta} U_m, \psi)
\end{aligned}
$$

***m* = 0:**

$$
\begin{aligned}
(\psi, \mathit{\Delta} Z_0) = (\psi, \mathit{\Delta} Z_1) &- \frac{k_1}{2} \bar{a}'_u(Q_1, U_0)(\psi, Z_1) - \frac{k_1}{2} \bar{a}''_{uu}(Q_1, U_0)(\mathit{\Delta} U_0, \psi, Z_1) \\
&- \frac{k_1}{2} \bar{a}''_{qu}(Q_1, U_0)(\mathit{\Delta} Q_0, \psi, Z_1) + \frac{k_1}{2} J''_1(U_0)(\mathit{\Delta} U_0, \psi)
\end{aligned}
$$

The resulting Crank-Nicolson scheme is known to be of second order. However, in contrast to the implicit Euler scheme, this method does not possess the property of strong A-stability. The structure of the time steps for the adjoint equations is quite unusual since in the first and in the last steps "half steps" occur, and in the other steps, terms containing the sizes of two adjacent time intervals k_m and k_{m+1} appear. This complicates the a priori error analysis for the adjoint schemes, which is discussed for instance in Becker [2].

Even if the schemes for the state and adjoint equations differ, also this type of discretization exhibits in a certain sense the property of interchanging of discretization and optimization: Discretization of the *weakly* formulated optimality system and building the discrete optimality system using the discretized state equation lead to the same discrete time stepping schemes.

3.5 Numerical results

Since the solution algorithms which we present in the following Chapter 4 strongly depend on the exactness of the computed derivatives of the reduced cost functional j_{kh}, we discuss in this section a possibility of validating the correctness of the derivatives numerically and present results on this for the four example configurations given in Section 2.2.

The main idea thereby is to compare the derivatives computed by the adjoint approaches from the Sections 2.5, 3.1, and 3.2 with difference quotients. For the *numerically evaluated* first and second central difference quotient for j_{kh}, we obtain under the assumption of sufficient regularity the error representations

$$\begin{aligned} e_1(\varepsilon) &:= \left| \frac{j_{kh}(q+\varepsilon\delta q) - j_{kh}(q-\varepsilon\delta q)}{2\varepsilon} - j'_{kh}(q)(\delta q) \right| \approx C\varepsilon^2 j'''_{kh}(r) + \frac{c}{\varepsilon}, \\ e_2(\varepsilon) &:= \left| \frac{j_{kh}(q+\varepsilon\delta q) - 2j_{kh}(q) + j_{kh}(q-\varepsilon\delta q)}{\varepsilon^2} - j''_{kh}(q)(\delta q, \delta q) \right| \approx C\varepsilon^2 j''''_{kh}(r) + \frac{c}{\varepsilon^2}, \end{aligned}$$

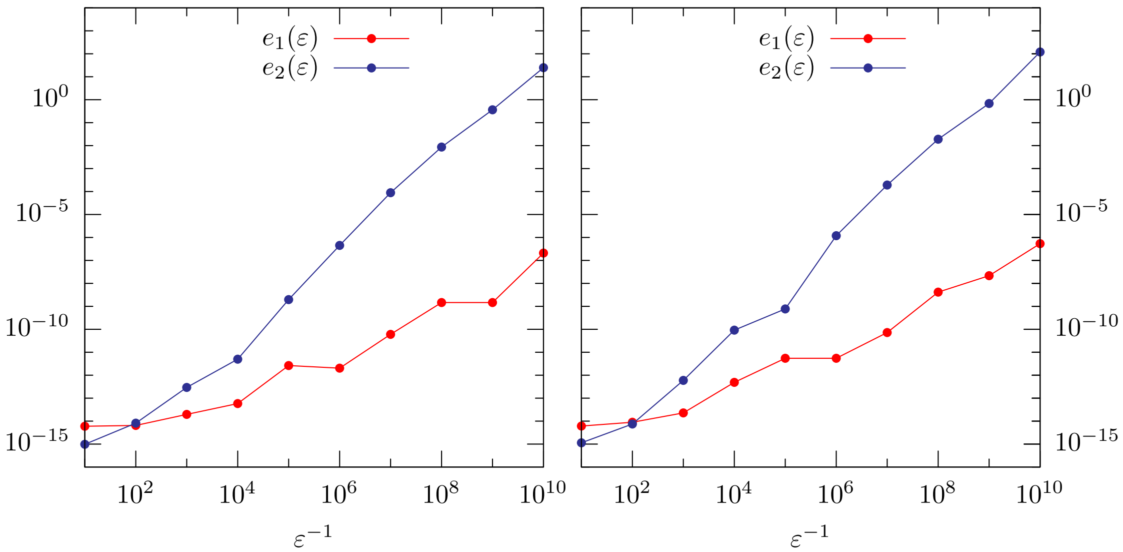
where $r \in (q - \varepsilon\delta q, q + \varepsilon\delta q)$ is an intermediate point and the constants c and C are independent of ε. Thereby, the parts of the error representation containing the positive exponents of ε are obtained by standard convergence analysis (Taylor expansion), whereas the parts with negative exponents come from an analysis of the truncation error for small ε.

That is, if the derivatives obtained by the adjoint approach are computed correctly, one may observe for $\varepsilon \to 0$ primarily quadratical convergence of the difference quotients to the numerically determined derivatives. But if ε is small enough, the truncation errors dominate and the difference quotients diverge. Since a reliable determination of a suitable value of ε is virtually impossible, the usage of derivatives computed by difference quotients is prohibitive for optimization algorithms.

The Figures 3.5, 3.6, 3.7, and 3.8 depict the errors $e_1(\varepsilon)$ and $e_2(\varepsilon)$ between the values of the derivatives computed by means of the difference quotients above and by mens of the adjoint approach presented in the preceding chapters for the configurations of the four examples presented in Section 2.2. Thereby, as discretization for the state space, the cG(1)dG(0) and the cG(1)cG(1) were examined. The control space was discretized as discussed in the Examples 3.1, 3.2, 3.3, and 3.4 in Section 3.3.

Except for Figure 3.5, we find in all figures quadratic convergence of the errors e_1 and e_2 which then moves to divergence when ε is getting too small. Due to the linear-quadratic structure of Example 2.1, which implies $j'''_{kh} = 0$, we observe in Figure 3.5 only the divergence of the difference quotient originated by the truncation error.

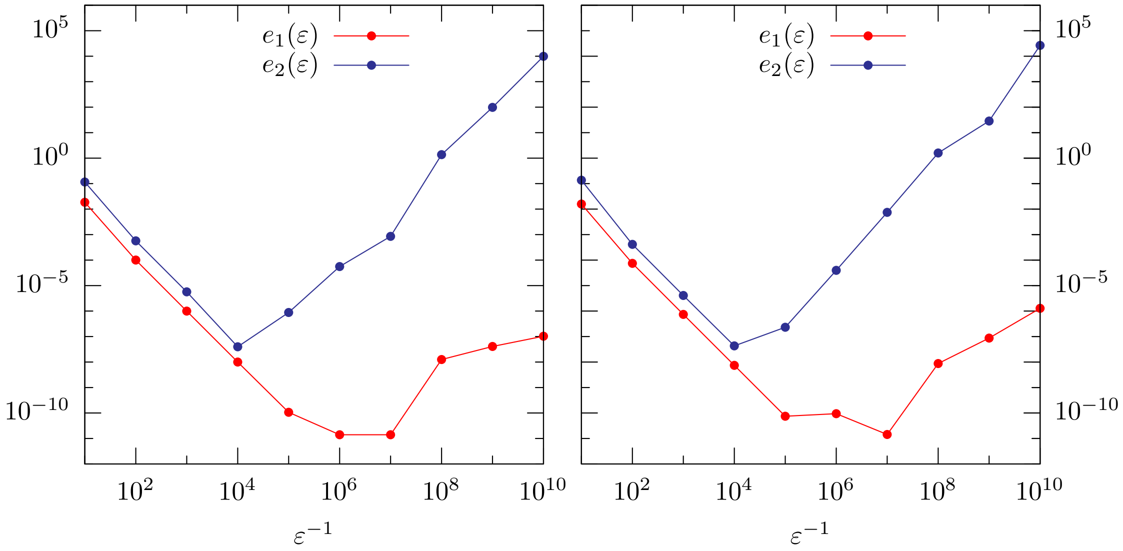
Thus, these tests confirm the correctness of the computed derivatives of first and second order. Especially, they verify the correctness of the linearized and adjoint time stepping schemes derived in Section 3.4.



(a) cG(1)dG(0)

(b) cG(1)cG(1)

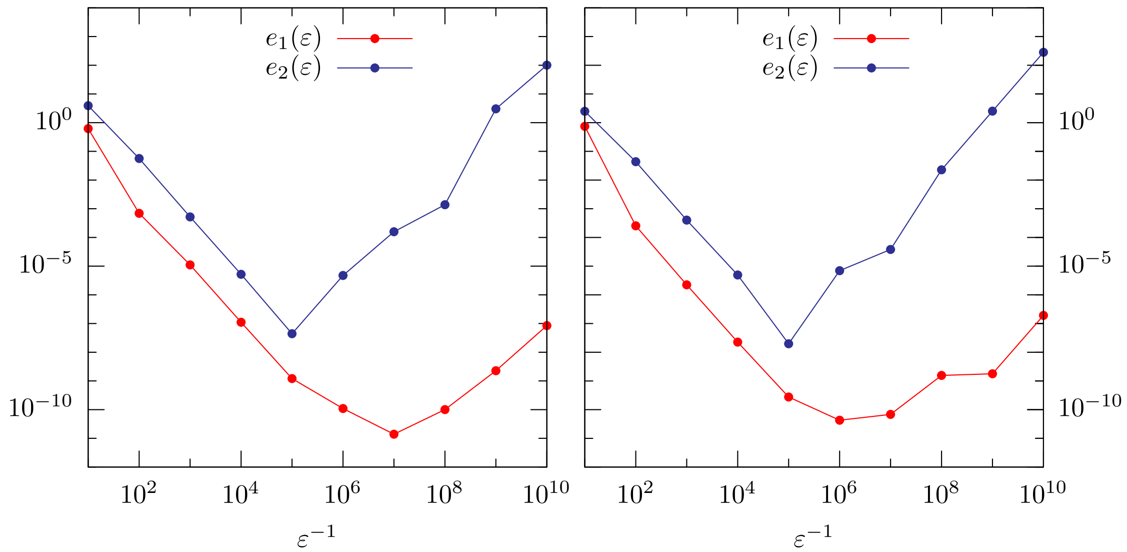
Figure 3.5. Convergence of the difference quotients for the reduced cost functional for Example 2.1



(a) cG(1)dG(0)

(b) cG(1)cG(1)

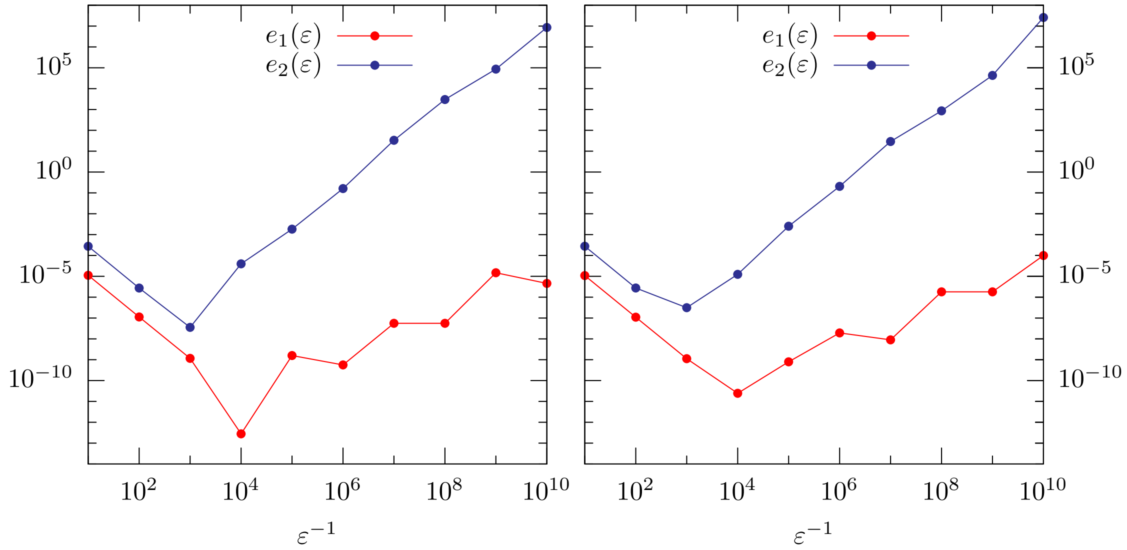
Figure 3.6. Convergence of the difference quotients for the reduced cost functional for Example 2.2



(a) cG(1)dG(0)

(b) cG(1)cG(1)

Figure 3.7. Convergence of the difference quotients for the reduced cost functional for Example 2.3



(a) cG(1)dG(0)

(b) cG(1)cG(1)

Figure 3.8. Convergence of the difference quotients for the reduced cost functional for Example 2.4

4 Algorithmic Aspects of Numerical Optimization

In this chapter, we present algorithmic aspects of numerical methods for solving the prototypical PDE-constrained optimizations problem ($\mathbb{P}$) after reformulation as unconstrained optimization problem ($\mathbb{P}^{\text{red}}$) and discretization as described in Chapter 3. In the first section, we present two variants of Newton-based optimization loops. In Section 4.2, we discuss aspects of solving the linear systems arising from the application of such (exact and inexact) Newton methods and present possible globalizations techniques. Thereby, we focus in particular on matrix-free linear solvers and on globalization by line search and trust-region methods. Section 4.3 is devoted to the discussion of storage reduction techniques, the so-called *checkpointing techniques*, which provide the possibility of reducing the size of memory required for executing the proposed algorithms in the context of nonstationary optimization. Finally, we close this chapter with the presentation of a numerical example in Section 4.4.

Some results of Section 4.3 on the application of storage reduction techniques in the context of nonstationary optimization problems are already published in Becker, Meidner, and Vexler [8].

Throughout this chapter, we consider the discretized control space Q_d of finite dimension with a basis

$$\{ \tau q_i \mid i = 1, 2, \dots, \dim Q_d \} . \tag{4.1}$$

Furthermore, we skip for simplicity the subscript σ at the arising solution variables. Since we only consider discrete states and controls here, this does not cause any misunderstandings.

4.1 Newton-type methods

As announced, we present in this section two variants of Newton's method for solving the reduced optimization problem ($\mathbb{P}^{\text{red}}$) respectively its discrete analogs. The two presented algorithms differ in the way of computing the update. In general, Newton-type methods are successfully used for solving optimization problem governed by time-dependent partial differential equations; see for example Hinze and Kunisch [45] and Tröltzsch [77].

The well known motivation (see for instance Deuflhard [26]) of the classical ordinary Newton method for solving a nonlinear operator equation

$$f(y) = 0,$$

with a continuously Fréchet differentiable mapping $f\colon Y \to Z$ on Banach spaces Y and Z is done by means of the Taylor expansion

$$0 = f(y + \delta y) = f(y) + f'(y)(\delta y) + r_1^f(y, \delta y)$$

for a given point $y \in Y$ and a direction $\delta y \in Y$. By dropping the higher order term r_1^f, we arrive at the equation determining the Newton update δy:

$$f'(y)(\delta y) = -f(y).$$

The next iterate y^+ of Newton's Method is then defined by $y^+ = y + \delta y$.

Under some assumptions on the derivatives of f, Newton's method is known to be quadratically convergent if it is started with an initial guess lying in a neighborhood of the solution. This assertion is proven in abstract spaces by the classical *Newton-Kantorovich* and *Newton-Mysovskikh* theorems; see for instance Kantorovich and Akilov [49] and Mysovskikh [63].

For unconstrained optimization, Newton's method is employed to find a control $q \in Q_d$ fulfilling the first order necessary optimality condition for the discrete reduced cost functional j_{kh}, that is

$$j'_{kh}(q)(\tau q) = 0 \quad \forall \tau q \in Q_d.$$

Applying Newton's method to this equation, each performed step requires the solution of the linear system

$$j''_{kh}(q)(\delta q, \tau q) = -j'_{kh}(q)(\tau q) \quad \forall \tau q \in Q_d. \tag{4.2}$$

However, in the context of numerical optimization, Newton's method is usually introduced differently to the motivation given above: We note, that the linear system (4.2) is the first order necessary optimality condition of the linear-quadratic subproblem

$$\text{Minimize } m(q, \delta q) := j_{kh}(q) + j'_{kh}(q)(\delta q) + \frac{1}{2} j''_{kh}(q)(\delta q, \delta q), \quad \delta q \in Q_d. \tag{4.3}$$

Thus, if δq is a solution of (4.3), it solves the linear system (4.2), too. Moreover, if the second derivatives $j''_{kh}(q)$ are positive definite, also the reversal of this assertion holds true.

To keep the algorithms presented in the sequel as general as possible, we consider additional restrictions to the unconstrained subproblem (4.3). We impose the constraint $\|\delta q\|_Q \leq \mu$ for given $\mu \in \mathbb{R} \cup \{ +\infty \}$ and formulate the constrained subproblem as

$$\text{Minimize } m(q, \delta q), \quad \delta q \in Q_d, \ \|\delta q\|_Q \leq \mu. \tag{4.4}$$

This offers the possibility to incorporate *line search* as well as *trust-region* globalization techniques in the formulation of Newton's method; see Section 4.2.2 for details on these techniques.

For stating the optimization loops based on Newton's method, we have to represent the derivatives of the reduced cost functional used in the formulations (4.2) and (4.3) in terms of vectors and matrices in $\mathbb{R}^{\dim Q_d}$ and $\mathbb{R}^{\dim Q_d \times \dim Q_d}$. Therefore, we introduce as first step the gradient $\nabla j_{kh}(q) \in Q_d$ and the Hessian $\nabla^2 j_{kh}(q) \colon Q_d \to Q_d$ defined as usual by the Hilbert space identifications

$$\begin{aligned} (\nabla j_{kh}(q), \tau q)_Q &= j'_{kh}(q)(\tau q) && \forall \tau q \in Q_d, \\ (\nabla^2 j_{kh}(q)\delta q, \tau q)_Q &= j''_{kh}(q)(\delta q, \tau q) && \forall \delta q, \tau q \in Q_d. \end{aligned}$$

By means of these representations, the key equation (4.2) determining the Newton update can be written as

$$(\nabla^2 j_{kh}(q)\delta q, \tau q_i)_Q = -(\nabla j_{kh}(q), \tau q_i)_Q, \quad i = 1, 2, \dots, \dim Q_d. \tag{4.5}$$

We now formulate this system of equations in terms of coefficient vectors and matrices: Let us first consider the term $(\nabla j_{kh}(q), \tau q_i)_Q$ on the right-hand side of (4.5). We express $\nabla j_{kh}(q) \in Q_d$ by means of its coefficient vector $\boldsymbol{f} \in \mathbb{R}^{\dim Q_d}$ with respect to the basis (4.1) and obtain

$$(\nabla j_{kh}(q), \tau q_i)_Q = \sum_{j=1}^{\dim Q_d} \boldsymbol{f}_j (\tau q_j, \tau q_i)_Q.$$

Hence, $\boldsymbol{f}$ is determined as solution of

$$\boldsymbol{G}\boldsymbol{f} = \Big((\nabla j_{kh}(q), \tau q_i)_Q\Big)_{i=1}^{\dim Q_d} = \Big(j'_{kh}(q)(\tau q_i)\Big)_{i=1}^{\dim Q_d},$$

where $\boldsymbol{G}$ is the Gramian matrix of the basis (4.1) defined by $\boldsymbol{G}_{ij} := (\tau q_j, \tau q_i)_Q$.

Remark 4.1. The concrete form of the Gramian matrix depends on the discrete control space Q_d. If, for instance, Q_d origins from a finite element discretization, $\boldsymbol{G}$ equals the mass matrix. In contrast, if $Q_d = Q$ is a finite-dimensional space of parameters, then $\boldsymbol{G}$ usually equals the identity matrix.

Concerning the left-hand side of (4.5), we represent δq by means of its coefficient vector $\boldsymbol{d} \in \mathbb{R}^{\dim Q_d}$. Then,

$$(\nabla^2 j_{kh}(q)\delta q, \tau q_i)_Q = \sum_{j=1}^{\dim Q_d} \boldsymbol{d}_j (\nabla^2 j_{kh}(q)\tau q_j, \tau q_i)_Q$$

implies that $\boldsymbol{d}$ fulfills

$$\boldsymbol{K}\boldsymbol{d} = \Big((\nabla^2 j_{kh}(q)\delta q, \tau q_i)_Q\Big)_{i=1}^{\dim Q_d} = \Big(j''_{kh}(q)(\delta q, \tau q_i)\Big)_{i=1}^{\dim Q_d},$$

where the matrix $\boldsymbol{K}$ is given by $\boldsymbol{K}_{ij} := (\nabla^2 j_{kh}(q)\tau q_j, \tau q_i)_Q = j''_{kh}(q)(\tau q_j, \tau q_i)$.

Consequently, the Newton equation (4.5) is equivalent to the following linear system for the coefficient vectors:

$$\boldsymbol{H}\boldsymbol{d} = -\boldsymbol{f}.$$

Here, the coefficient matrix $\boldsymbol{H}$ of the Hessian $\nabla^2 j_{kh}(q)$ is given in terms of the regular Gramian matrix $\boldsymbol{G}$ by $\boldsymbol{H} := \boldsymbol{G}^{-1}\boldsymbol{K}$.

Especially if $\dim Q_d$ is large, the computation of the whole matrix $\boldsymbol{H}$ is very costly and should be avoided. In such situations, it is reasonable to compute only the coefficient vector $\boldsymbol{h}$ of the product $\nabla^2 j_{kh}(q)\delta q \in Q_d$ in order to use it within an iterative solver. Similar as before, we obtain

$$(\nabla^2 j_{kh}(q)\delta q, \tau q_i)_Q = \sum_{j=1}^{\dim Q_d} \boldsymbol{h}_j (\tau q_j, \tau q_i)_Q,$$

and $\boldsymbol{h}$ is given as solution of

$$\boldsymbol{G}\boldsymbol{h} = \Big((\nabla^2 j_{kh}(q)\delta q, \tau q_i)_Q \Big)_{i=1}^{\dim Q_d} = \Big(j''_{kh}(q)(\delta q, \tau q_i) \Big)_{i=1}^{\dim Q_d}.$$

For stating the algorithms, we employ the following notations for coefficient vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^{\dim Q_d}$:

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle \coloneqq \boldsymbol{a}^T \boldsymbol{G} \boldsymbol{b} \qquad \text{and} \qquad |\boldsymbol{a}| \coloneqq \langle \boldsymbol{a}, \boldsymbol{a} \rangle^{\frac{1}{2}}.$$

Via these definitions, we have that $\mathbb{R}^{\dim Q_d}$ equipped with $|\cdot|$ is isometric isomorphic to Q_d equipped with $\|\cdot\|_Q$. Furthermore, we can rewrite the linear-quadratic subproblem from (4.3) in terms of the introduced coefficient vectors as

$$m(q, \boldsymbol{d}) = j_{kh}(q) + \langle \boldsymbol{f}, \boldsymbol{d} \rangle + \frac{1}{2} \langle \boldsymbol{H}\boldsymbol{d}, \boldsymbol{d} \rangle.$$

Now, we are prepared to state the announced two versions of Newton's method. In both algorithms, the required information on the first derivative j'_{kh} to obtain $\boldsymbol{f}$ is computed using representation (2.10). However, the two algorithms differ in the way how they solve the linear-quadratic subproblem (4.4) to obtain a correction δq for the current control q. If problem (4.4) is solved exactly by a direct solver, the resulting algorithm belongs to the class of *exact Newton methods*, whereas it is called to be an *inexact Newton method* if the linear systems are solved only approximatively, that is by an iterative solver as for instance the methods of conjugate gradients; see Section 4.2.1 for a detailed discussion.

Remark 4.2. For many concrete optimization problems, the inverting of $\boldsymbol{G}$ to compute $\boldsymbol{f}$ and $\boldsymbol{h}$ can be avoided. In the configuration of Example 2.1, we have for instance

$$\nabla j_{kh}(q) = \alpha(q - \hat{q}) + z_{kh} \qquad \text{and} \qquad \nabla^2 j_{kh}(q)\delta q = \alpha \delta q + \delta z_{kh}.$$

Consequently, here $\boldsymbol{f}$ and $\boldsymbol{h}$ can be expressed directly in terms of the coefficient vectors for q, $\hat{q}$, z_{kh}, δq, and δz_{kh}.

4.1.1 Optimization loop without assembling the Hessian

Algorithm 4.1 treats the computation of a solution to (4.4) by an iterative solver, which only requires products of the Hessian with given vectors and does not necessitate the whole Hessian matrix. A widely used solver which fulfills these requirements of matrix-freeness is the already mentioned conjugate gradient method. Thus, when using this approach, we always end up with an inexact Newton method.

Algorithm 4.1. Optimization loop *without* assembling the Hessian

```
Choose an initial $q^0 \in Q_d$, $\mu_0 \in \mathbb{R} \cup \{ +\infty \}$, and set $l = 0$.
repeat
    Compute $u^l$, i.e., solve the discrete state equation.
    Compute $z^l$, i.e., solve the discrete adjoint equation.
    Assemble the coefficient vector $\boldsymbol{f}$ of the gradient $\nabla j_{kh}(q^l)$. For doing so, evaluate the
    right-hand side of representation (2.10) for $\tau q = \tau q_i$, $i = 1, 2, \dots, \dim Q_d$ and solve the
    linear system
        $$\boldsymbol{G}\boldsymbol{f} = \Big( j'_{kh}(q^l)(\tau q_i) \Big)_{i=1}^{\dim Q_d}.$$
    Solve the problem
        $$\text{Minimize } m(q^l, \boldsymbol{d}), \quad \boldsymbol{d} \in \mathbb{R}^{\dim Q_d},\ |\boldsymbol{d}| = \|\delta q\|_Q \le \mu_l,$$
    approximately by use of a solver which only requires matrix-vector products of the
    Hessian computed by Algorithm 4.2.
    Choose $\mu_{l+1}$ and $\nu_l$ depending on the behavior of the algorithm.
    Set $q^{l+1} = q^l + \nu_l \delta q$.
    Increment $l$.
until $|\boldsymbol{f}| = \|\nabla j_{kh}(q^l)\|_Q <$ TOL
```

Algorithm 4.2. Computation of the product $\nabla^2 j(q^l)\delta q$

```
Require: $u^l$ and $z^l$ are already computed for the given $q^l$.
Compute $\delta u^l$, i.e., solve the discrete tangent equation.
Compute $\delta z^l$, i.e., solve the discrete additional adjoint equation.
Assemble the coefficient vector $\boldsymbol{h}$ of the product $\nabla^2 j_{kh}(q^l)\delta q$. For doing so, evaluate the
right-hand side of representation (2.20) for $\tau q = \tau q_i$, $i = 1, 2, \dots, \dim Q_d$ and solve the
linear system
    $$\boldsymbol{G}\boldsymbol{h} = \Big( j''_{kh}(q)(\delta q, \tau q_i) \Big)_{i=1}^{\dim Q_d}.$$
```

In Algorithm 4.1, several steps have to be concretized. Possibilities therefor are presented in Section 4.2.

The computation of the required matrix-vector products can be done using representation (2.20) and is described in Algorithm 4.2. We note that in order to obtain the product of the Hessian with a given vector, we have to solve one tangent equation and one additional adjoint equation. This has to be done in each step of the linear solver.

4.1.2 Optimization loop with assembling the Hessian

In contrast to Algorithm 4.1, Algorithm 4.3 assembles the whole Hessian respectively its representation as coefficient matrix. Consequently, one may use every (direct or iterative) linear solver for solving (4.4) or respectively the linear system (4.2). To compute the coefficient matrix $\boldsymbol{H}$ of the Hessian $\nabla^2 j_{kh}(q)$, we employ the representation of the second derivatives of the reduced cost functional given by (2.21). Thus, in each Newton step we have to solve the tangent equation for each basis vector in (4.1).

Algorithm 4.3. Optimization loop *with* assembling the Hessian

```
Choose an initial $q^0 \in Q_d$, $\mu_0 \in \mathbb{R} \cup \{ +\infty \}$, and set $l = 0$.
repeat
    Compute $u^l$, i.e., solve the discrete state equation.
    Compute $\{ \tau u_i^l \mid i = 1, 2, \dots, \dim Q_d \}$ for the chosen basis of $Q_d$, i.e. solve the discrete
    tangent equation for each of the basis vectors $\tau q_i$ in (4.1).
    Compute $z^l$, i.e., solve the discrete adjoint equation.
    Assemble the coefficient vector $\boldsymbol{f}$ of the gradient $\nabla j_{kh}(q^l)$. For doing so, evaluate the
    right-hand side of representation (2.10) for $\tau q = \tau q_i$, $i = 1, 2, \dots, \dim Q_d$ and solve the
    linear system
        $$\boldsymbol{G}\boldsymbol{f} = \Big( j'_{kh}(q^l)(\tau q_i) \Big)_{i=1}^{\dim Q_d}.$$
    Assemble the coefficient matrix $\boldsymbol{H}$ of the Hessian $\nabla^2 j_{kh}(q^l)$. For doing so, evaluate the
    right-hand side of representation (2.21) for $\delta q = \tau q_j$ $\tau q = \tau q_i$, $\delta u = \tau u_j$, and $\tau u = \tau u_i$,
    $i = 1, 2, \dots, \dim Q_d$ and solve the matrix equation
        $$\boldsymbol{G}\boldsymbol{H} = \Big( j''_{kh}(q^l)(\tau q_i, \tau q_j) \Big)_{i,j=1}^{\dim Q_d}.$$
    Solve the problem
        $$\text{Minimize } m(q^l, \boldsymbol{d}), \quad \boldsymbol{d} \in \mathbb{R}^{\dim Q_d},\ |\boldsymbol{d}| = \|\delta q\|_Q \le \mu_l$$
    exactly or approximately by means of a (linear) solver.
    Choose $\mu_{l+1}$ and $\nu_l$ depending on the behavior of the algorithm.
    Set $q^{l+1} = q^l + \nu_l \delta q$.
    Increment $l$.
until $|\boldsymbol{f}| = \|\nabla j_{kh}(q^l)\|_Q < \text{TOL}$
```

As noted for Algorithm 4.1, concretizations for the vaguely formulated steps of Algorithm 4.3 are presented in Section 4.2.

4.1.3 Comparison of the presented optimization loops

We now compare the efficiency of the two presented algorithms under the assumption of using the conjugate gradient method for solving the linear system in the Algorithms 4.1 and 4.3. Then, for one step of Newton's method, Algorithm 4.1 requires the solution of two linear problems (tangent equation and additional adjoint equation) per step of the CG-iteration, whereas for Algorithm 4.3, it is necessary to solve $\dim Q_d$ many tangent equations for assembling the Hessian matrix.

Thus, if we have to perform n_{CG} steps of the CG method per Newton step (a number, which can hardly be determined a priori), we should favor Algorithm 4.3, if and only if

$$\frac{\dim Q_d}{2} \le n_{\text{CG}}. \tag{4.6}$$

In Section 4.3, we discuss a comparison of these two algorithms in the context of storage reduction techniques.

4.2 Extensions and concretizations of Newton methods

In this section, we give concretizations of the vaguely formulated steps of the Algorithms 4.1 and 4.3. That is, we present an iterative method for solving the subproblems (4.3) and (4.4) and two globalization techniques extending the region of convergence of Newton's method.

4.2.1 Linear solvers

Since algorithms for solving the unconstrained as well as the constrained subproblems (4.3) and (4.4) are well-known for cases where the whole Hessian is available (cf. Nocedal and Wright [65]), we focus here on a matrix-free algorithm for solving these subproblems. As one possibility, we present the classical *Steihaug conjugate gradient method* (cf. Steihaug [74]). This algorithm is designed for solving the constrained subproblem (4.4) approximatively to obtain the Newton update δq respectively its coefficient vector $\boldsymbol{d}$. Since for $\mu = +\infty$, the constrained problem coincides with the unconstrained one, the algorithm presented in the sequel can be applied to both subproblems.

Algorithm 4.4. Steihaug conjugate gradient method

```
Set $\boldsymbol{p}^0 = 0$, $\boldsymbol{r}^0 = -\boldsymbol{f}$, $\boldsymbol{g}^0 = \boldsymbol{r}^0$, and $i = 0$.
loop
    Compute the coefficient vector $\boldsymbol{h}$ of the product $\nabla^2 j(q^l)\boldsymbol{g}^i$ by means of Algorithm 4.2.
    Set $\gamma = \langle \boldsymbol{h}, \boldsymbol{g}^i \rangle$.
    if $\gamma \le 0$ then
        if $\mu_l < \infty$ then
            Compute $\xi > 0$ such that $|\boldsymbol{p}^i + \xi \boldsymbol{g}^i| = \mu_l$.
            Set $\boldsymbol{d} = \boldsymbol{p}^i + \xi \boldsymbol{g}^i$.
        else
            Set $\boldsymbol{d} = \boldsymbol{p}^{i-1}$ or $\boldsymbol{d} = \boldsymbol{p}^0$ if $i = 0$.
        break (Negative curvature found.)
    Compute $\alpha = {}^{|\boldsymbol{r}^i|^2}/_{\gamma}$.
    Set $\boldsymbol{p}^{i+1} = \boldsymbol{p}^i + \alpha \boldsymbol{g}^i$.
    if $|\boldsymbol{p}^{i+1}| > \mu_l$ then
        Compute $\xi > 0$ such that $|\boldsymbol{p}^i + \xi \boldsymbol{g}^i| = \mu_l$.
        Set $\boldsymbol{d} = \boldsymbol{p}^i + \xi \boldsymbol{g}^i$
        break (Norm of approximation too large.)
    Compute $\boldsymbol{r}^{i+1} = \boldsymbol{r}^i - \alpha \boldsymbol{h}$
    if ${}^{|\boldsymbol{r}^{i+1}|}/_{|\boldsymbol{r}^0|} < \mathrm{TOL}$ then
        Set $\boldsymbol{d} = \boldsymbol{p}^{i+1}$.
        break (Approximation good enough.)
    Compute $\beta = {}^{|\boldsymbol{r}^{i+1}|^2}/_{|\boldsymbol{r}^i|^2}$.
    Set $\boldsymbol{g}^{i+1} = \boldsymbol{r}^{i+1} + \beta \boldsymbol{g}^i$.
    Increment $i$.
```

Algorithm 4.4 includes three different termination rules: We terminate ...

- ... in step 21, if we have a sufficiently good approximation to the Newton step (4.2).
- ... in step 17, if the norm of the approximation is too large with respect to the bound μ_l. Then, we take a linear combination of the previous iterate and the current one.
- ... in step 11, if we encounter a direction of negative curvature. Then, we move to the boundary given by μ_l if finite or take in the unconstrained case the previous iterate.

The last termination rule extends the classical conjugate gradient (CG) method to the cases where the Hessian is not necessarily positive definite.

It is crucial to start Algorithm 4.4 with the initial guess $\boldsymbol{p}^0 = 0$, because then, the computed directions δq are always descent directions. Additionally, we obtain the following properties of the iterates $\boldsymbol{p}^i$ which are necessary to show that all proposed termination rules are reasonable:

$$\begin{gathered}
0 = |\boldsymbol{p}^0| < \cdots < |\boldsymbol{p}^i| < |\boldsymbol{p}^{i+1}| < \cdots < |\boldsymbol{d}| = \|\delta q\|_Q \leq \mu_l, \\
m(q^l, \boldsymbol{p}^0) > \cdots > m(q^l, \boldsymbol{p}^i) > m(q^l, \boldsymbol{p}^{i+1}) > \cdots > m(q^l, \boldsymbol{d}) = m(q^l, \delta q).
\end{gathered}$$

Proofs of these assertions can be found for instance in Steihaug [74] and Nocedal and Wright [65].

The unpreconditioned CG method as described here can be inefficient when the Hessian is ill-conditioned and may even fail to reach the desired accuracy. Hence, it is important to introduce preconditioning techniques into the CG method. It is possible to modify Algorithm 4.4 such that it solves directly the preconditioned system. Details on this and a rigorous convergence analysis of the whole algorithm can again be found in Steihaug [74].

4.2.2 Globalization techniques

In this subsection, we give two possible concretizations of how to choose μ and ν in the Algorithms 4.1 and 4.3 for enlarging the region of convergence of Newton's method. Such globalization techniques are necessary since the classical Newton method does not necessarily converge for every initial guess. That means, it is in general not globally convergent. The resulting algorithms are then—depending on the choice of the globalization approach—called *line search Newton-CG method* or *trust-region Newton-CG method.*

In what follows, we just give an overview over two of these techniques (line search and trust-region methods) from a practical point of view. A detailed discussion can be found in the standard literature as Nocedal and Wright [65] and Conn, Gould, and Toint [23].

Line search methods

For using line search, we set $\mu_0 = +\infty$ and keep it constant during the Algorithms 4.1 and 4.3. That is, we have only to solve the unconstrained subproblem (4.3). Thus, as crucial part remains the proper choice of $\nu_l > 0$ to maximize the reduction of the cost functional in the computed direction δq. Here, one can find several possibilities in the literature. The best possibility would be

$$\nu_l = \arg\min_{\nu > 0} j_{kh}(q^l + \nu \delta q).$$

However, in most cases this *exact line search* is too expensive. A popular simplification is the *Armijo backtracking*: Let $\beta, \gamma \in (0, 1)$ (often $\beta = 0.5$, $\gamma = 0.01$) be chosen constants. Determine the largest step size $\nu_l \in \{ 1, \beta, \beta^2, \dots \}$ fulfilling

$$j_{kh}(q^l + \nu_l \delta q) \leq j_{kh}(q^l) + \gamma \nu_l \nabla j_{kh}(q^l) \delta q.$$

This choice leads under some assumption on j_{kh} to a globally convergent Newton method which has the same local convergence properties as the classical Newton method.

Trust-region methods

In trust-region methods, ν_l is always set as 0 or 1, but now a finite μ is chosen. In a prototypical trust-region Newton algorithm (cf. Nocedal and Wright [65]), we choose constants $\mu_{\max} > 0$, $\mu_0 \in (0, \mu_{\max})$, and $\gamma \in [0, 0.25)$. The determination of μ_{l+1} is then depending on the ratio

$$\rho_l := \frac{j_{kh}(q^l) - j_{kh}(q^l + \delta q)}{m(q^l, 0) - m(q^l, \delta q)},$$

which measures how good the model function m, which is minimized when solving the subproblem (4.4), approximates the functional j_{kh}. We then choose μ_{l+1} and ν_l by means of Algorithm 4.5.

Algorithm 4.5. Determination of μ_{l+1} and ν_l

```
if $\rho_l < 0.25$ then
    Set $\mu_{l+1} = 0.25\|\delta q\|_Q$.
else if $\rho_l > 0.75$ and $\|\delta q\|_Q = \mu_l$ then
    Set $\mu_{l+1} = \min(2\mu_l, \mu_{\max})$.
else
    Set $\mu_{l+1} = \mu_l$.
if $\rho_l > \gamma$ then
    Set $\nu_l = 1$.
else
    Set $\nu_l = 0$.
```

Under standard assumptions on j_{kh}, also this trust-region Newton method is globally convergent and converges locally like the classical Newton method.

4.3 Storage reduction techniques

When computing the gradient of the reduced cost functional as described in the algorithms in the previous sections, we need to have access to the solution u of the state equation at all points in space and time while computing the adjoint solution z. Similarly, we need the solutions of the state (u), tangent (δu), and adjoint (z) equations to solve the additional adjoint equation for δz when computing matrix-vector products with the Hessian of the reduced cost functional. If all data are stored, the storage grows linearly with respect to the number of time intervals in the time discretization and the dimension of the space discretization $\dim V_h^s$. For large problems, especially in three space dimensions, storing all the necessary data might be impossible. To overcome this difficulty, storage reduction techniques have been developed in Griewank [41], Berggren, Glowinski, and Lions [14], and Walther and Griewank [86]. All the techniques presented there exhibit the property of reducing the storage when performing M time steps from $\mathcal{O}(M)$ to $\mathcal{O}(\log_2 M)$ at the cost of $\mathcal{O}(M \log_2 M)$ additional time steps. For the so-called *binomial checkpointing* proposed for instance in [41] and [86], optimal complexity was proven.

In this section, we present an approach, which relies on ideas from Berggren, Glowinski, and Lions [14]. We analyze the complexity of this algorithm and prove that the required storage grows only logarithmic with respect to the number of time intervals. The main purpose of this section is to discuss this storage reduction technique in the context of the optimization algorithms described in Section 4.1. Due to its structure, we call the presented approach *multi-level windowing.*

In the last years, the capacities of main memory and hard discs have been growing rapidly. Hence, such storage reduction techniques can often be superseded by storing all data. Although the costs of writing all data on the hard disc are much higher than for keeping them in the main memory, storing of data on hard discs could be advantageous in view of run-time trade-offs in the range of $\mathcal{O}(\log M)$ which are implicated by the checkpointing approaches. At the end of Section 4.4, we discuss this drawback in more detail.

4.3.1 Abstract algorithm

First, we consider the following abstract setting: Let the two time stepping schemes

$$\begin{aligned} \mathbf{x}_{m-1} &\mapsto \mathbf{x}_m \qquad &&\text{for } m = 1, 2, \dots, M, \\ (\mathbf{y}_{m+1}, \mathbf{x}_m) &\mapsto \mathbf{y}_m \qquad &&\text{for } m = M-1, M-2, \dots, 0 \end{aligned}$$

be given together with an initial value $\mathbf{x}_0$ and the mapping $\mathbf{x}_M \mapsto \mathbf{y}_M$ prescribing the terminal condition for $\mathbf{y}$. The time stepping schemes coming from the dG(0) and cG(1) semidiscretizations derived in Section 3.4 are concrete realizations of these abstract schemes.

Additionally, we assume the solutions $\mathbf{x}_m$ as well as $\mathbf{y}_m$ to be of the same size for all $m = 0, 1, \dots, M$. However, if this is not the case, the checkpointing technique presented in the sequel can be applied to clusters of time steps similar in size instead of single time steps. Such clustering is for instance important when using dynamical meshes, since in this case, the amount of storage for a solution $\mathbf{x}_m$ depends on the mesh currently used.

The trivial approach of performing the forward and backward iterations is to compute and store the whole forward solution $(\mathbf{x}_m)_{m=0}^M$, and use these values to compute the backward solution $(\mathbf{y}_m)_{m=0}^M$. The required amount of storage to do this is $M+1$ in terms of the size of one forward solution $\mathbf{x}_m$. The number of forward steps necessary to compute the whole backward solution is M, the number of backward steps is M, too.

The aim of the following checkpointing algorithms is to reduce the needed storage by performing some additional forward steps. To introduce the checkpointing, we assume that we can factorize the number of given time steps M as $M = PR$ with positive integers P and R. With this, we can separate the set of time points $\{\,0, 1, \dots, M\,\}$ in P slices each containing $R-1$ time steps and $P+1$ sets containing one element as

$$\begin{aligned}\{\,0,\dots,M\,\} = \{\,0\,\} \cup \{\,1,\dots,R-1\,\} \cup \{\,R\,\} \cup \cdots \\ \cdots \cup \{\,(P-1)R\,\} \cup \{\,(P-1)R+1,\dots,PR-1\,\} \cup \{\,PR\,\}\,.\end{aligned}$$

The algorithm works as follows: First, we compute the forward solution $\mathbf{x}_m$ for $m = 1, 2, \dots, M$ and store the $P+1$ samples $\{\,\mathbf{x}_0, \mathbf{x}_R, \dots, \mathbf{x}_{PR}\,\}$. Additionally, we store the $R-1$ values of $\mathbf{x}$ in the last slice $\{\,\mathbf{x}_{(P-1)R+1}, \mathbf{x}_{(P-1)R+2}, \dots, \mathbf{x}_{PR-1}\,\}$. Now, we have the necessary information on $\mathbf{x}$ to compute $\mathbf{y}_m$ for $m = M, M-1, \dots, (P-1)R+1$. After doing so, the values of $\mathbf{x}$ in the last slice are no longer needed. We can replace them with the values of $\mathbf{x}$ in the next-last slice, which we can compute directly using the forward time stepping scheme since we stored the value $\mathbf{x}_{(P-2)R}$ in the first run. Thereby, we can compute $\mathbf{y}_m$ for $m = (P-1)R, (P-1)R-1, \dots, (P-2)R+1$. This can now be done iteratively till we have computed $\mathbf{y}$ in the first slice and finally obtain the value $\mathbf{y}_0$. The so called *one-level windowing* is presented in detail in Algorithm 4.6.

Algorithm 4.6. `OneLevelWindowing`(P, R, M)

```
Require: $M = PR$.
Store $\mathbf{x}_0$.
Take $\mathbf{x}_0$ as initial value for $\mathbf{x}$.
for $m = 1$ to $(P-1)R$ do
    Compute $\mathbf{x}_m$.
    if $m$ is a multiple of $R$ then
        Store $\mathbf{x}_m$.
for $i = (P-1)R$ downto $0$ step $R$ do
    Take $\mathbf{x}_i$ as initial value for $\mathbf{x}$.
    for $m = i+1$ to $i+R-1$ do
        Compute $\mathbf{x}_m$.
        Store $\mathbf{x}_m$.
    if $i = M-R$ then
        Compute $\mathbf{x}_M$.
        Store $\mathbf{x}_M$.
    for $m = i+R$ downto $i+1$ do
        Compute $\mathbf{y}_m$ in virtue of $\mathbf{x}_m$.
        Delete $\mathbf{x}_m$ from memory.
    if $i = 0$ then
        Compute $\mathbf{y}_0$.
        Delete $\mathbf{x}_0$ from memory.
```

During the Execution of Algorithm 4.6, the needed amount of memory is not exceeding $(P+1)+(R-1)$ forward solutions. Each of the backward solutions $\mathbf{y}_m$ is computed exactly once, so we need like in the direct approach M solving steps to obtain the whole solution $\mathbf{y}$. To compute the necessary values of $\mathbf{x}_m$, we have to solve $M + (P-1)(R-1)$ forward steps, since we have to compute each of the values of $\mathbf{x}$ once additionally in the first $P-1$ slices. We have in total

$$S(\{\,P, R\,\}) = P + R \qquad \text{and} \qquad W(\{\,P, R\,\}) = 2M - P - R + 1,$$

where S denotes the required amount of memory in terms of the size of one forward solution and W denotes the number of time steps to provide the forward solution $\mathbf{x}$ needed to compute the whole backward solution $\mathbf{y}$. Each of them depends on the used factorization $\{\,P, R\,\}$.

The presented approach can be extended to factorizations $\mathcal{F}_M = \{\,M_0, M_1, \dots, M_L\,\}$ of M in $|\mathcal{F}_M| = L + 1$ factors for $L \in \mathbb{N}_0$. This extension can be obtained via the following inductive argumentation: Assuming $M = M_0 M_1 \cdots M_L$ with positive integers M_l, we can apply the algorithm described above to the factorization $M = PR$ with $P = M_0$ and $R = M_1 M_2 \cdots M_L$, and then recursively to each of the P slices. This so called *multi-level windowing* is described in Algorithm 4.7. It has to be initiated by the call `MultiLevelWindowing`$(0, 0, \mathcal{F}_M, M)$.

Algorithm 4.7. `MultiLevelWindowing`$(s, l, \mathcal{F}_M, M)$

```
Require: $M = \prod_{M_j \in \mathcal{F}_M} M_j$.
Set $L = |\mathcal{F}_M| - 1$, $P = M_l$, and $R = M_{l+1} \cdots M_L$.
if $l = 0$ and $s = 0$ then
    Store $\mathbf{x}_0$.
Take $\mathbf{x}_s$ as initial value for $\mathbf{x}$.
for $m = 1$ to $(P-1)R$ do
    Compute $\mathbf{x}_{s+m}$.
    if $m$ is a multiple of $R$ then
        Store $\mathbf{x}_{s+m}$.
for $i = (P-1)R$ downto $0$ step $R$ do
    if $l + 1 < L$ then
        Call MultiLevelWindowing$(s + i, l + 1, \mathcal{F}_M, M)$.
    else
        Take $\mathbf{x}_{s+i}$ as initial value for $\mathbf{x}$.
        for $m = i + 1$ to $i + R - 1$ do
            Compute $\mathbf{x}_{s+m}$.
            Store $\mathbf{x}_{s+m}$.
        if $s + i = M - R$ then
            Compute $\mathbf{x}_M$.
            Store $\mathbf{x}_M$.
        for $m = i + R$ downto $i + 1$ do
            Compute $\mathbf{y}_{s+m}$ in virtue of $\mathbf{x}_{s+m}$.
            Delete $\mathbf{x}_{s+m}$ from memory.
        if $s + i = 0$ then
            Compute $\mathbf{y}_0$.
            Delete $\mathbf{x}_0$ from memory.
```

Of course, there holds by construction

$$\texttt{OneLevelWindowing}(P, R, M) = \texttt{MultiLevelWindowing}(0, 0, \{\,P, R\,\}, M).$$

Remark 4.3. The presented approach can be extended to cases where a suitable factorization $M = M_0 M_1 \cdots M_L$ does not exist. Then, we consider a representation of M as $M = (M_0 - 1)R_0 + \tilde{R}_0$ with positive integers M_0, R_0, and $\tilde{R}_0$ fulfilling $R_0 \leq \tilde{R}_0 < 2R_0$ and apply this idea recursively to the generated subintervals of length R_0 or $\tilde{R}_0$. This can easily be done, since by construction, the reminder interval of length $\tilde{R}_0$ has at least the same length as the regular subintervals.

In the following theorem, we calculate the necessary amount of storage and the number of forward steps necessary to perform the multi-level windowing described in Algorithm 4.7 for a given factorization $\mathcal{F}_M = \{ M_0, M_1, \dots, M_L \}$ of length $|\mathcal{F}_M| = L + 1$:

Theorem 4.1. *For given $L \in \mathbb{N}_0$ and a factorization $\mathcal{F}_M = \{ M_0, M_1, \dots, M_L \}$ of the number of time steps M with $M_l \in \mathbb{N}$, the required amount of memory of the multi-level windowing algorithm to perform all backward solution steps is*

$$S(\mathcal{F}_M) = \sum_{M_l \in \mathcal{F}_M} (M_l - 1) + 2.$$

To achieve this storage reduction, the number of performed forward steps enhances to

$$W(\mathcal{F}_M) = |\mathcal{F}_M| M - \sum_{M_l \in \mathcal{F}_M} \frac{M}{M_l} + 1.$$

Proof. We prove the theorem by mathematical induction:

***L* = 0:** Here we use the trivial approach where the entire forward solution $\mathbf{x}$ is saved. As considered in the beginning of this section, we then have $S(\mathcal{F}_M) = M + 1$ and $W(\mathcal{F}_M) = M$ for $\mathcal{F}_M = \{ M \}$.

***L* − 1 ⇝ *L*:** We consider the factorization $\mathcal{F}_M^* = \{ M_0, M_1, \dots, M_{L-2}, M_{L-1}M_L \}$ of length L additionally to the given factorization $\mathcal{F}_M$ of length $L + 1$. Then, we obtain in the same way as for the one-level windowing, where we have reduced the storage mainly from $PR - 1$ to $(P - 1) + (R - 1)$, the identity

$$S(\mathcal{F}_M) = S(\mathcal{F}_M^*) - (M_{L-1}M_L - 1) + (M_{L-1} - 1) + (M_L - 1).$$

In virtue of the induction hypothesis for $S(\mathcal{F}_M^*)$, it follows

$$\begin{aligned} S_L(\mathcal{F}_M) &= \sum_{M_l \in \mathcal{F}_M^*} (M_l - 1) - (M_{L-1}M_L - 1) + (M_{L-1} - 1) + (M_L - 1) + 2 \\ &= \sum_{M_l \in \mathcal{F}_M} (M_l - 1) + 2. \end{aligned}$$

Now, we prove the assertion for W. To this end, we justify the equality

$$W(\mathcal{F}_M) = W(\mathcal{F}_M^*) + \frac{M}{M_{L-1}M_L}(M_{L-1} - 1)(M_L - 1).$$

The asserted identity follows immediately from the fact that we divide each of the $\frac{M}{M_{L-1}M_L}$ slices

$$\{\, s+1, s+2, \dots, s+M_{L-1}M_L - 1 \,\}\,, \quad s = 0, M_{L-1}M_L, \dots, \left(\frac{M}{M_{L-1}M_L} - 1\right) M_{L-1}M_L$$

of length $M_{L-1}M_L - 1$ as

$$\begin{aligned}\{\, s+1, \dots, s+M_{L-1}M_L - 1 \,\} = \{\, s+1, \dots, s+M_L - 1 \,\} \cup \{\, s+M_L \,\} \cup \cdots \\ \cdots \cup \{\, s+(M_{L-1}-1)M_L \,\} \cup \{\, s+(M_{L-1}-1)M_L + 1, \dots, s+M_{L-1}M_L - 1 \,\}\,.\end{aligned}$$

Since we just need to compute the forward solution in the first $M_{L-1} - 1$ subslices when we change from the factorization of length L to the one of length $L+1$, the additional work equals

$$\frac{M}{M_{L-1}M_L}(M_{L-1} - 1)(M_L - 1)$$

as stated. Then, we obtain in virtue of the induction hypothesis for $W(\mathcal{F}_M^*)$

$$\begin{aligned} W(\mathcal{F}_M) &= |\mathcal{F}_M^*|M + M - \sum_{M_l \in \mathcal{F}_M^*} \frac{M}{M_l} + \frac{M}{M_{L-1}M_L} - \frac{M}{M_{L-1}} - \frac{M}{M_L} + 1 \\ &= |\mathcal{F}_M|M - \sum_{M_l \in \mathcal{F}_M} \frac{M}{M_l} + 1. \end{aligned}$$ □

If $M^{\frac{1}{L+1}} \in \mathbb{N}$, the minimal storage S_L of all possible factorizations of length $L+1$ is

$$S_L \coloneqq S(\{M^{\frac{1}{L+1}}, \dots, M^{\frac{1}{L+1}}\}) = (L+1)(M^{\frac{1}{L+1}} - 1) + 2.$$

The numbers of forward steps for the memory-optimal factorization then results in

$$W_L \coloneqq W(\{M^{\frac{1}{L+1}}, \dots, M^{\frac{1}{L+1}}\}) = (L+1)(M - M^{\frac{L}{L+1}}) + 1.$$

If we choose additionally $L \approx \log_2 M$, we obtain for the optimal factorization from above the proposed logarithmic growth of the necessary amount of storage and the corresponding number of forward steps

$$S_L = \mathcal{O}(\log_2 M) \qquad \text{and} \qquad W_L = \mathcal{O}(M \log_2 M).$$

In the following subsections, we consider the multi-level windowing described here in the context of nonstationary optimization. We give a detailed estimate for the number of steps and the amount of memory required to perform one Newton step for a given number of levels $L \in \mathbb{N}_0$.

4.3.2 Optimization loop without assembling the Hessian

First, we treat the variant of the optimization algorithm, which does not assemble the entire Hessian of the reduced cost functional and is given in Algorithm 4.1. As stated in this algorithm, it is necessary to compute the value of the reduced cost functional and the gradient once per Newton step. To apply the derived checkpointing techniques, we set $\mathbf{x} = u$, $\mathbf{y} = z$ and note, that Algorithm 4.7 can easily be extended to compute the necessary terms for evaluating the functional and the gradient during the forward or backward computation, respectively. Thus, the total number of times steps needed to do this, is $W^{\text{grad}} = W(\mathcal{F}_M) + M$. The required amount of memory is $S^{\text{grad}} = S(\mathcal{F}_M)$.

Additionally to the gradient, we need to compute one matrix-vector product of the Hessian times a given vector in each of the n_{CG} steps of the conjugate gradient method (cf. Algorithm 4.4). This is done as described in Algorithm 4.2. For avoiding the storage of u or z in all time steps, we have to recompute u, δu, z, and δz again in every CG step. Consequently, we set here $\mathbf{x} = (u, \delta u)$ and $\mathbf{y} = (z, \delta z)$. We obtain $W^{\text{hess}} = 2(W(\mathcal{F}_M) + M)$ and $S^{\text{hess}} = 2S(\mathcal{F}_M)$.

In total we achieve

$$\begin{aligned} W^{(1)} &= W^{\text{grad}} + n_{\text{CG}} W^{\text{hess}} = (1 + 2n_{\text{CG}})(W(\mathcal{F}_M) + M) \quad \text{and} \\ S^{(1)} &= \max(S^{\text{grad}}, S^{\text{hess}}) = 2S(\mathcal{F}_M). \end{aligned}$$

Remark 4.4. The checkpointing algorithm (Algorithm 4.7) can be modified to reduce the necessary forward steps under acceptance of increasing the needed amount of storage as follows: We do not delete u while computing z at the initial checkpoints where u is saved before starting the computation of z. Additionally, we store z at these checkpoints. These saved values of u and z can be used to reduce the necessary number of forward steps to provide the values of u and δu for computing one matrix-vector product with the Hessian. Of course, when saving additional samples of u and z, the needed amount of storage increases. For one Newton step we obtain the total work $\widetilde{W}^{(1)}$ and storage $\widetilde{S}^{(1)}$ as

$$\widetilde{W}^{(1)} = W^{(1)} - 2n_{\text{CG}} \min(S(\mathcal{F}_M), M) \quad \text{and} \quad \widetilde{S}^{(1)} = S^{(1)} + 2S(\mathcal{F}_M) - M_0 - 2.$$

Due to this modification, the algorithm includes the case of not using checkpointing at all for $L = 0$, while the original form of the algorithm deletes u during the computation of z also for $L = 0$.

4.3.3 Optimization loop with assembling the Hessian

For using Algorithm 4.3, it is necessary to compute u, τu_i $(i = 1, 2, \dots, \dim Q_d)$, and z. Again, the evaluation of the reduced cost functional can be done during the first forward computation, and the evaluation of the gradient and the Hessian can be done during the computation of z. So, we set $\mathbf{x} = (u, \tau u_1, \tau u_2, \dots, \tau u_{\dim Q_d})$ and $\mathbf{y} = z$. Thus, the required number of steps and the needed amount of memory are

$$W^{(2)} = (1 + \dim Q_d) W(\mathcal{F}_M) + M \qquad \text{and} \qquad S^{(2)} = (1 + \dim Q_d) S(\mathcal{F}_M).$$

4.3.4 Comparison of the presented optimization loops

We obtain directly $S^{(2)} \geq S^{(1)}$, since we have obviously $\dim Q_d \geq 1$. The relation between $W^{(1)}$ and $W^{(2)}$ depends on the factorization of M. A simple calculation leads to the following condition:

$$W^{(2)} \leq W^{(1)} \quad \Longleftrightarrow \quad \frac{\dim Q_d}{2} \leq n_{\mathrm{CG}} \Big(1 + \frac{M}{W(\mathcal{F}_M)} \Big).$$

If we choose $\mathcal{F}_M$ and L such that $W(\mathcal{F}_M) \approx M \log_2 M$, we can express the condition above just in terms of M as

$$W^{(2)} \lesssim W^{(1)} \quad \Longleftrightarrow \quad \frac{\dim Q_d}{2} \lesssim n_{\mathrm{CG}} \Big(1 + \frac{1}{\log_2 M} \Big). \tag{4.7}$$

This implies, that even though the required memory for the second algorithm with assembling the entire Hessian is greater, this algorithm requires only then fewer steps than the first one, if condition (4.7) is fulfilled. Note, that condition (4.7) is the extension of criterion (4.6) to the case when applying checkpointing.

Remark 4.5. If we apply globalization techniques such as line search or trust-region methods (cf. Section 4.2.2) to one of the presented optimization algorithms, we have to compute the solution of the state equation and the value of the cost functional several times without computing the gradient or the Hessian. The direct approach for doing this, is to compute the state, evaluate it and delete it afterwards. This might not be optimal, since for the following computation of the gradient (and the Hessian) via checkpointing, the needful preparations are not done. So, the better way of doing this is to run Algorithm 4.7 until line 19 and break consequently after completing the forward solution. If after that the value of the gradient is needed, it is possible to restart directly on line 20 with the computation of the backward solutions. If we consider the version with assembling the Hessian, we have to compute the tangent solutions in an extra forward run in which we can also use the stored values of the state solution.

4.4 Numerical results

In this section, we examine the behavior of the two types of optimization algorithms described in the Sections 4.1 and 4.2 as well as the checkpointing technique presented in Section 4.3 in the situation of a given optimization problem with finite-dimensional control. To this end, we consider an optimal control problem with terminal observation where the control variable $q \in Q = \mathbb{R}^8$ enters the initial condition of the nonlinear state equation. We choose as spatial domain $\Omega = (0,1)^3$, the final time $T = 1$ and pose the state equation as

$$\begin{aligned} \partial_t u - \varepsilon \Delta u + u^2 &= 0 && \text{in } \Omega \times I, \\ \varepsilon \partial_n u &= 0 && \text{on } \partial\Omega \times I, \\ u &= g_0 + \sum_{i=1}^{8} g_i q_i && \text{on } \Omega \times \{ 0 \}. \end{aligned} \tag{4.8}$$

Here, g_i $(i = 1, 2, \ldots, 8)$ are given shape functions. The desired state $\hat{u}$ (see Figure 4.1) is given for $x = (x_1, x_2, x_3)^T$ as

$$\hat{u}(x) = \frac{1}{6}(3 + x_1 + x_2 + x_3),$$

and the cost functional to be minimized is chosen as

$$J(q,u) = \frac{1}{2}\|u(T) - \hat{u}\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\sum_{i=1}^{8} q_i^2.$$

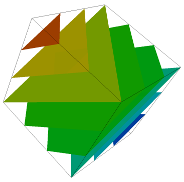


Figure 4.1. Isosurfaces of the desired state $\hat{u}$

The parameters ε and α are selected as $\varepsilon = 10^{-1}$ and $\alpha = 10^{-4}$, and the state space X is given in virtue of the choices $V = H^1(\Omega)$ and $H = L^2(\Omega)$. For discretizing the state space, we employ the cG(1)dG(0) and cG(1)cG(1) schemes performing 100 time steps on a mesh consisting of 4,096 hexahedral cells with diameter $h = 0.0625$. Since $Q = \mathbb{R}^8$, the control space needs not to be discretized. Thus, we set $Q_d = Q$.

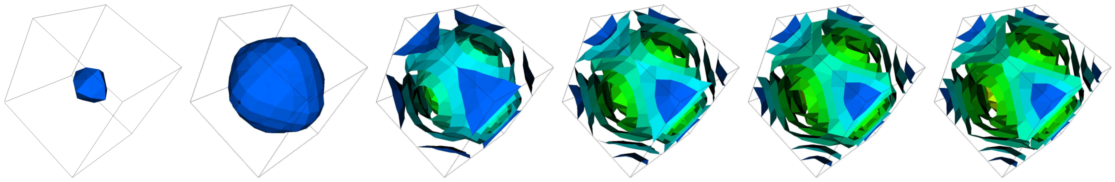
In Table 4.1, we show the progression of the norm of the gradient of the reduced functional $\|\nabla j_{hk}\|_Q$ and the reduction of the values of the cost functional j_{hk} during the executions of Newton's method applied to the optimization problem. Since condition (4.6) is not fulfilled in this example, one should prefer rather Algorithm 4.1 which makes only use of matrix-vector products of the Hessian than the alternative Algorithm 4.3.

Table 4.1. Results of the optimization loop with dG(0) and cG(1) time discretization starting with initial guess $q^0 = (0, 0, \dots, 0)^T$

	cG(1)dG(0)			cG(1)cg(1)		
Newton step	n_{CG}	$\|\nabla j_{hk}\|_Q$	j_{hk}	n_{CG}	$\|\nabla j_{hk}\|_Q$	j_{hk}
0	—	$1.21 \cdot 10^{-01}$	$2.76 \cdot 10^{-01}$	—	$1.21 \cdot 10^{-01}$	$2.76 \cdot 10^{-01}$
1	2	$4.99 \cdot 10^{-02}$	$1.34 \cdot 10^{-01}$	2	$4.98 \cdot 10^{-02}$	$1.34 \cdot 10^{-01}$
2	2	$2.00 \cdot 10^{-02}$	$6.28 \cdot 10^{-02}$	2	$1.99 \cdot 10^{-02}$	$6.33 \cdot 10^{-02}$
3	3	$7.61 \cdot 10^{-03}$	$2.94 \cdot 10^{-02}$	3	$7.62 \cdot 10^{-03}$	$3.00 \cdot 10^{-02}$
4	3	$2.55 \cdot 10^{-03}$	$1.64 \cdot 10^{-02}$	3	$2.57 \cdot 10^{-03}$	$1.70 \cdot 10^{-02}$
5	3	$6.03 \cdot 10^{-04}$	$1.32 \cdot 10^{-02}$	3	$6.21 \cdot 10^{-04}$	$1.37 \cdot 10^{-02}$
6	3	$5.72 \cdot 10^{-05}$	$1.29 \cdot 10^{-02}$	3	$6.18 \cdot 10^{-05}$	$1.34 \cdot 10^{-02}$
7	3	$6.37 \cdot 10^{-07}$	$1.29 \cdot 10^{-02}$	3	$7.62 \cdot 10^{-07}$	$1.34 \cdot 10^{-02}$
8	3	$1.75 \cdot 10^{-10}$	$1.29 \cdot 10^{-02}$	3	$1.21 \cdot 10^{-10}$	$1.34 \cdot 10^{-02}$

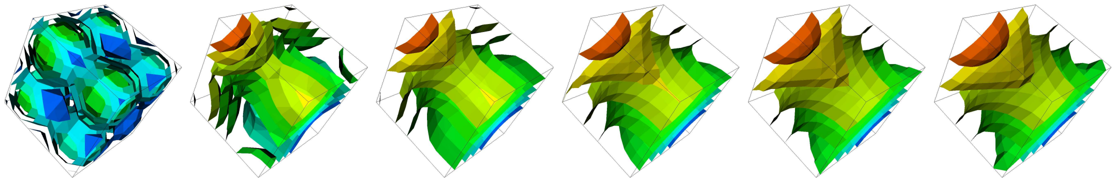
In the Figures 4.2 and 4.3 we show isosurfaces of the initial control q^0 and the optimal control q^8 obtained after eight Newton steps of the proposed algorithm. Figure 4.3(f) demonstrates the good qualitative agreement of the optimal solution with the desired state depicted in Figure 4.1.

We now consider the behavior of the presented checkpointing technique described earlier in this chapter when applied to the considered optimization problem. Table 4.2 demonstrates the



(a) $t = 0.0$ (b) $t = 0.2$ (c) $t = 0.4$ (d) $t = 0.6$ (e) $t = 0.8$ (f) $t = 1.0$

Figure 4.2. Isosurfaces of the state corresponding to the initial control q^0



(a) $t = 0.0$ (b) $t = 0.2$ (c) $t = 0.4$ (d) $t = 0.6$ (e) $t = 0.8$ (f) $t = 1.0$

Figure 4.3. Isosurfaces of the state corresponding to the optimal control q^8

reduction in storage requirements as proposed in Section 4.3. We achieve a storage reduction about the factors 30 and 45 for the two variants of the optimization loop. Thereby, the total number of time steps grows about the factor 3.2 for the algorithm with, and about the factor 4.0 for the algorithm without assembling the Hessian. Comparable run-time trade-offs are obtained in Sternberg [75], where the binomial checkpointing routine introduced in Griewank [41] was examined in the context of optimal control.

Table 4.2. Reduction of the storage requirement due to windowing for 500 time steps in the cG(1)dG(0) discretization

	With Hessian		Without Hessian	
Factorization	#Checkpoints	#Time steps	#Checkpoints	#Time steps
500	4509	45000	1503	35000
$5 \cdot 100$	945	80640	210	87948
$10 \cdot 50$	540	84690	120	90783
$2 \cdot 2 \cdot 5 \cdot 25$	288	120582	64	118503
$5 \cdot 10 \cdot 10$	216	114174	48	113463
$4 \cdot 5 \cdot 5 \cdot 5$	153	136512	34	130788
$2 \cdot 2 \cdot 5 \cdot 5 \cdot 5$	144	146646	32	138663

We remark, that although the factorization $2 \cdot 2 \cdot 5 \cdot 25$ consists of more factors than the factorization $5 \cdot 10 \cdot 10$, both the storage requirement and the total number of time steps are greater for the first factorization than for the second one. The reason for this is the imbalance of the size of the different factors in $2 \cdot 2 \cdot 5 \cdot 25$. As shown in Section 4.3, in the optimal factorization all factors are identical. So it is plausible that a factorization as for instance $5 \cdot 10 \cdot 10$ is more efficient than one where the size of the factors varies much.

Table 4.2 also proves the asserted dependence on condition (4.7) which states when to use which variant of the optimization loop on the considered factorization of M. For the factorizations $5 \cdot 100$ and $10 \cdot 50$, the variant with assembling the Hessian needs less forward steps than the other variant without assembling the Hessian. However, for the remaining factorizations the situation is vice versa.

For a concrete chosen spatial mesh size of for example to 32,768 cells per time step, application of the checkpointing routine would reduce the necessary amount of memory from initially 1,236 MB to 39 MB when assembling the Hessian and from 412 MB to 9 MB otherwise. However, these numbers result only from theoretical investigations based on the memory consumption of one solution sample on the considered mesh.

A practical examination of the behavior of the windowing technique seems only possible on coarse discretizations since even run-time trade-offs in the region of 3 make computations on finer discretizations extremely time consuming. Thus, from a practical point of view, the run-time trade-off of checkpointing compared to the basic approach (storing all solutions) limits its usage to situations where the numerical recomputation of solutions is rather fast. Usually, this is only the case if the size of the solutions is small. On the other hand, in view of the rapidly growing capacities of main memory in modern compute servers, larger and larger solutions can be stored. Thus, the need for storage reduction techniques becomes only necessary for highly memory consuming systems originating from fine discretizations especially in three space dimensions. An example of such a configuration, namely the simulation of a three-dimensional flow around a cylinder, is investigated in Heuveline and Walther [43].

But even in such large scale computations, the limitations of main memory can be avoided by storing all the data on hard disk. Even if the access times to hard disc are much larger than for accessing the main memory, this approach can be competitive since the checkpointing procedure needs in practice at least three times the run-time of the approach with storing everything in main memory. It is arguable whether this trade-off can totally be consumed by reading and writing access to hard disc. Since the capacity of hard discs is virtually unlimited, this approach constitutes a serious alternative to the checkpointing routines.

5 A Priori Error Analysis

In this chapter, we derive a priori estimates for the error caused by discretizing the optimization problem in space and time. In particular, we estimate the error due to the cG(s)dG(r) discretization of the state and the discretization of the control concerning a linear-quadratic parabolic model problem with distributed control.

While the a priori error analysis for finite element discretizations of optimal control problems governed by elliptic equations is discussed in many publications, see for example Falk [34], Geveci [40], Arada, Casas, and Tröltzsch [1], Meyer and Rösch [62], Casas, Mateos, and Tröltzsch [21], Hinze [44], Becker and Vexler [13], and Rösch and Vexler [71], there are only a few published results on this topic for parabolic problems. Amongst others, there are the following articles presenting error estimates for Galerkin type discretizations of parabolic optimal control problems:

- In McNight and Bosarge [58], the authors consider a general class of parabolic optimal control problems and proof an estimate which assesses the error caused by discretization of the control, state, and adjoint state in space only.

- In Winther [87], an optimal control problem with Neumann boundary control and terminal observation is considered. For this configuration, the author shows estimates in $L^\infty(I, L^2(\Omega))$ for the error in the state and the adjoint state variable when discretizing the state by a backward discretization in time and linear finite elements in space. Since the control variable is eliminated from the optimality system, no estimates for the control are given.

- The objective of Lasiecka and Malanowski [51] are (control-constrained) optimal control problems with control by right-hand side and a cost functional which is distributed over space and time. As discretization scheme, the *discrete-time Ritz-Galerkin* scheme for the state and the control variable is chosen. There, the state is discretized by linear finite elements in space and the control discretization uses piecewise constant polynomials in space. With respect to time, the discrete-time Ritz-Galerkin method utilizes for both variables the *θ-scheme* which includes implicit Euler ($\theta = 0$) and Crank-Nicolson ($\theta = 1/2$) schemes. An estimate for the error in the control variable is proven which is optimal with respect to the parameters of the control discretization in the case $\theta = 0$. Based on this result, the authors show the same order of convergence for the error in the state and adjoint state variables.

- In Malanowski [57], the author considers (control-constrained) optimal control problems with control via right-hand side or via boundary conditions of Neumann type. The state variable is again discretized by the discrete-time Ritz-Galerkin scheme with linear finite elements in space. For the control discretization, the discrete-time Ritz-Galerkin

> scheme combined with either linear or constant finite elements in space is examined. An estimate for the error in terms of the control variable of optimal order in the case $\theta = 0$ is presented under the restriction of a prescribed coupling of the temporal and spatial discretization parameters; that is under the condition $k \approx h$.

Our a priori analysis differs from the approaches used in the presented literature: For the discretization error between the solution (q, u) of the continuous optimization problem and the optimal solution (q_σ, u_σ) of the Galerkin-discretized problem, we prove optimal error estimates of the structure

$$\|q - q_\sigma\|_{L^2(I,L^2(\Omega))} \le C_1(u,z)\, k^{r+1} + C_2(u,z)\, h^{s+1} + C_3(q)\, k_d^{r_d+1} + C_4(q)\, h_d^{s_d+1},$$

where r, r_d are the highest degrees of polynomials used in the time discretization of the state and the control variable, respectively, and s, s_d are the highest degree of polynomials used in the space discretization of the state and the control variable. The constants $C_1(u, z)$ and $C_2(u, z)$ depend on the temporal and spatial regularity of the optimal state u and the corresponding adjoint state z. The temporal and spatial regularity of the optimal control q determines the constants $C_3(q)$ and $C_4(q)$.

Based on this result for the error in the control variable, estimates of optimal order for the error in the state and adjoint state variable and also in terms of the cost functional are proven. This extends the results presented in [57] in the following directions: Firstly, we consider not only the lowest order discretization cG(1)dG(0) (which corresponds to the investigated discrete-time Ritz-Galerkin scheme in the case $\theta = 0$) but also higher order cG(s)dG(r) schemes and secondly, we strictly separate the influences of the temporal and spatial regularities of the solutions and also the influences of the time and space discretizations. In particular, the discretization parameters of all involved discretizations can be chosen independently of each other.

In the following section, we give the precise formulation of the optimal control problem investigated in this chapter. Furthermore, we recall results on existence, uniqueness, and regularity of solutions to the considered optimal control problem and concretize the dG(r) semilinear form. Based on the stability estimates to be developed in Section 5.2, we provide an a priori error analysis for the state equation in Section 5.3. The main results on the error analysis for the considered optimal control problem are given in Section 5.4. In this section, error estimates for the error in the control, state, and adjoint state variables are developed. Furthermore, we derive an a priori estimate for the error in terms of the cost functional. In the last section, we present numerical results illustrating our theoretical predictions.

The estimates developed here are also collected in Meidner and Vexler [60] and the proposed techniques are successfully employed in Meidner and Vexler [61] for the development of an a priori error analysis for linear-quadratic optimal control problems with pointwise inequality constraints on the control variable.

5.1 Continuous optimal control problem

As state equation, we consider the linear heat equation

$$\begin{aligned} \partial_t u - \Delta u &= f + q \qquad &&\text{in } \Omega \times I, \\ u &= u_0 &&\text{on } \Omega \times \{ 0 \} \end{aligned} \tag{5.1}$$

combined with either homogeneous Dirichlet or homogeneous Neumann boundary conditions on $\partial\Omega \times I$. Throughout this chapter, the spatial domain Ω is assumed to be polygonally bounded and convex. By means of a given desired state $\hat{u}$, the cost functional J is chosen to be of tracking type:

$$J(q,u) = \frac{1}{2} \int_I \|u(t) - \hat{u}(t)\|^2_{L^2(\Omega)}\, dt + \frac{\alpha}{2} \int_I \|q(t)\|^2_{L^2(\Omega)}\, dt. \tag{5.2}$$

Then, the optimal control problem considered in this chapter is given as concretization of the abstract optimization problem ($\mathbb{P}$) by

$$\text{Minimize } J(q,u) \text{ subject to (5.1)}, \ (q,u) \in Q \times X. \tag{P}$$

As already discussed in Example 2.1, we choose here

$$H = L^2(\Omega), \quad V = H^1_0(\Omega) \text{ or } V = H^1(\Omega), \quad \text{and} \quad Q = L^2(I, H) \tag{5.3}$$

for embedding this control problem in the abstract setting of Chapter 2. The right-hand side f and the desired state $\hat{u}$ are assumed to be in $L^2(I,H)$ and the initial condition u_0 to be in V. Under these assumptions, there exists a solution $u \in X$ to (5.1) which is of even higher regularity:

Theorem 5.1. *Let V and H be chosen accordingly to* (5.3)*. Then, there exists for fixed control $q \in Q$, $f \in L^2(I,H)$, and $u_0 \in V$ a unique solution $u \in X$ of problem* (5.1) *equipped with homogeneous Dirichlet or homogeneous Neumann boundary conditions. Moreover, the solution u exhibits the improved regularity*

$$u \in L^2(I, H^2(\Omega) \cap V) \cap H^1(I, L^2(\Omega)) \hookrightarrow C(\bar{I}, V).$$

It holds the stability estimate

$$\|\partial_t u\|_I + \|\Delta u\|_I \leq C\{\|f + q\|_I + \|\nabla u_0\|_I\}.$$

Proof. The proof of existence and uniqueness is given in Lions [53] and Wloka [88]. The improved regularity is proven in Evans [33], and the embedding of $L^2(I, H^2(\Omega) \cap V) \cap H^1(I, L^2(\Omega))$ into $C(\bar{I}, V)$ can be found for instance in Dautray and Lions [25]. □

The improved regularity of the state carries over to the regularity of the optimal control:

Theorem 5.2. *For given $f, \hat{u} \in L^2(I,H)$, $u_0 \in V$, and $\alpha > 0$, the optimal control problem* (P) *admits a unique solution $(q,u) \in Q \times X$. The optimal control q possesses the regularity*

$$q \in L^2(I, H^2(\Omega) \cap V) \cap H^1(I, L^2(\Omega)).$$

Proof. The existence of a unique optimal solution $(q, u) \in Q \times X$ can be shown here like in Section 2.3 for the optimal control problem from Example 2.1. Then, the first order necessary optimality condition (2.7) respectively the optimality system (2.12) and Theorem 5.1 (applied to the adjoint solution z) imply the stated regularity of q. □

In contrast to the remaining chapters of this thesis, we use here for the a priori analysis the formulation (3.2) of the dG(r) discretization of the state equation without the jump term at t_0. The equivalence of this scheme to the scheme (3.1) used before was proven in Remark 3.2 for the dG(r) semidiscretization and holds true also for space-time discretizations on a fixed spatial mesh. Hence, we restrict ourselves to the case of fixed spatial discretizations, that is

$$V_h^{s,m} = V_h^s, \quad m = 0, 1, \ldots, M.$$

However, the results of the analysis presented in the Sections 5.1 and 5.3.1 also hold true in the case when dynamical changes of the space discretization are allowed. The results of Section 5.3.2 can not be applied directly when dynamical meshes are used.

We abbreviate the u-dependent part of the left-hand side of the dG(r) semilinear form (3.2) concretized for the linear-quadratic problem (P) by B defined for $u_k, \varphi \in \widetilde{X}_k^r$ as

$$B(u_k, \varphi) := \sum_{m=1}^{M} (\partial_t u_k, \varphi)_{I_m} + (\nabla u_k, \nabla \varphi)_I + \sum_{m=2}^{M} ([u_k]_{m-1}, \varphi_{m-1}^+) + (u_{k,0}^+, \varphi_0^+). \tag{5.4}$$

We note that with integration by parts, it also holds

$$B(u_k, \varphi) = -\sum_{m=1}^{M} (u_k, \partial_t \varphi)_{I_m} + (\nabla u_k, \nabla \varphi)_I - \sum_{m=1}^{M-1} (u_{k,m}^-, [\varphi]_m) + (u_{k,M}^-, \varphi_M^-). \tag{5.5}$$

Since we have left out the parts of the dG(r) formulation that depend on the control q, the bilinear form B represents also the left-hand side of the dG(r) method for the uncontrolled problem, that is for (5.1) in the case $q = 0$. Thus, we may employ B for both the analysis of the uncontrolled case derived in the Sections 5.2 and 5.3 and for the analysis of the optimal control problem (P) developed in Section 5.4.

In what follows, we use the abbreviations

$$\|v\| := \|v\|_{L^2(\Omega)}, \quad \|v\|_I := \|v\|_{L^2(I,L^2(\Omega))}, \quad \text{and} \quad \|v\|_{I_m} := \|v\|_{L^2(I_m,L^2(\Omega))}$$

to shorten the notation. They are defined analogously to those of the inner products $(\cdot,\cdot)$, $(\cdot,\cdot)_I$, and $(\cdot,\cdot)_{I_m}$ already used in the previous chapters.

5.2 Stability estimates for the state and adjoint state

The first step in proving the desired a priori estimates is to show stability estimates for the solution of the semidiscrete and the fully discretized state equation (5.1) in the case $q = 0$. The state equation for the dG(r) discretization of (5.1) reads by means of the bilinear form B for given right-hand side f and initial condition u_0 as

$$B(u_k, \varphi) = (f + q, \varphi)_I + (u_0, \varphi_0^+) \quad \forall \varphi \in \widetilde{X}_k^r. \tag{5.6}$$

Remark 5.1. Since we use here the modified dG(r) formulation (3.2) which does not contain conditions for values at time t_0^-, we redefine $\widetilde{X}_k^r$ and $\widetilde{X}_{k,h}^{r,s}$ in the sense that its elements do not necessarily possess values at t_0^-.

Remark 5.2. Using a density argument, it is possible to show that the continuous solution $u \in X$ of (5.1) satisfies also the identity

$$B(u, \varphi) = (f + q, \varphi)_I + (u_0, \varphi_0^+) \quad \forall \varphi \in \widetilde{X}_k^r.$$

Thus, we have here the property of *Galerkin orthogonality*

$$B(u - u_k, \varphi) = 0 \quad \forall \varphi \in \widetilde{X}_k^r,$$

although the dG(r) semidiscretization is a nonconforming Galerkin method ($\widetilde{X}_k^r \not\subseteq X$).

The fully cG(s)dG(r)-discretized formulation (cf. Section 3.2) of the state equation (5.1) aims at the determination of $u_{kh} \in \widetilde{X}_{k,h}^{r,s}$ such that

$$B(u_{kh}, \varphi) = (f + q, \varphi)_I + (u_0, \varphi_0^+) \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s}. \tag{5.7}$$

Now, we are going to proof a first stability estimate for the semidiscrete uncontrolled state equation, that is for (5.6) in the case $q = 0$. A similar estimate is shown in Eriksson and Johnson [30, 31]. However, it is proven therein only for the case $f = 0$. This would not be applicable to the considered control problem (P) where the control acts as right-hand side of the state equation.

Theorem 5.3. *For the solution $u_k \in \widetilde{X}_k^r$ of the uncontrolled* dG(r)*-semidiscretized state equation* (5.6) *with right-hand side $f \in L^2(I, H)$ and initial condition $u_0 \in V$, the stability estimate*

$$\sum_{m=1}^{M} \|\partial_t u_k\|_{I_m}^2 + \|\Delta u_k\|_I^2 + \sum_{m=1}^{M} k_m^{-1} \|[u_k]_{m-1}\|^2 \le C\{\|f\|_I^2 + \|\nabla u_0\|^2\}$$

holds. The constant C only depends on the polynomial degree r and the domain Ω. The jump term $[u_k]_0$ at $t = 0$ is defined as $u_{k,0}^+ - u_0$.

Proof. We first note that by means of the definition $[u_k]_0 = u_{k,0}^+ - u_0$, the solution $u_k \in \widetilde{X}_k^r$ of (5.6) in the case $q = 0$ fulfills also for all $\varphi \in \mathcal{P}_r(I_m, V)$ the following system of equations:

$$(\partial_t u_k, \varphi)_{I_m} + (\nabla u_k, \nabla \varphi)_{I_m} + ([u_k]_{m-1}, \varphi_{m-1}^+) = (f, \varphi)_{I_m} \quad m = 1, 2, \ldots, M. \tag{5.8}$$

The proof of the estimate consist of three steps—one for each term of its left-hand side. The steps base on consecutively testing with $\varphi = -\Delta u_k$, $\varphi = (t - t_{m-1})\partial_t u_k$, and $\varphi = [u_k]_{m-1}$.

(i) At first, we want to choose $\varphi = -\Delta u_k$. For applying integration by parts in space to (5.8), it is necessary to prove $\Delta u_k\big|_{I_m} \in \mathcal{P}_r(I_m, H)$. This assertion follows immediately from applying elliptic regularity theory (cf. Evans [33]) to the transformed time stepping equation

$$(\nabla u_k, \nabla \varphi)_{I_m} = (f - \partial_t u_k, \varphi)_{I_m} - ([u_k]_{m-1}, \varphi_{m-1}^+).$$

The fact that $u_k\big|_{I_m}$ is polynomial in time with values in $V \subseteq H$ implies that the right-hand side is in H for almost all $t \in I_m$. Thus, $\Delta u_k\big|_{I_m}$ is also in H for almost all $t \in I_m$, and since $u_k\big|_{I_m}$ is polynomial with respect to time, this yields $\Delta u_k\big|_{I_m} \in \mathcal{P}_r(I_m, H)$.

Consequently, it is feasible to integrate (5.8) by parts in space to obtain the formulation

$$(\partial_t u_k, \varphi)_{I_m} - (\Delta u_k, \varphi)_{I_m} + ([u_k]_{m-1}, \varphi_{m-1}^+) = (f, \varphi)_{I_m} \quad m = 1, 2, \dots, M. \tag{5.9}$$

The arising boundary terms vanish for both homogeneous Neumann or homogeneous Dirichlet boundary conditions.

Since there are no spatial derivatives on the test function φ anymore, formulation (5.9) holds not only for all $\varphi \in \mathcal{P}_r(I_m, V)$ but by the density of V in H also for all $\varphi \in \mathcal{P}_r(I_m, H)$. Hence, we may choose $\varphi = -\Delta u_k$ as test function and get by applying integration by parts in space a second time

$$(\partial_t \nabla u_k, \nabla u_k)_{I_m} + (\Delta u_k, \Delta u_k)_{I_m} + ([\nabla u_k]_{m-1}, \nabla u_{k,m-1}^+) = (f, -\Delta u_k)_{I_m}.$$

Again, the arising boundary terms vanish due to the prescribed homogeneous boundary conditions of Neumann or Dirichlet type.

By means of the identities

$$(\partial_t v, v)_{I_m} = \frac{1}{2}\|v_m^-\|^2 - \frac{1}{2}\|v_{m-1}^+\|^2, \tag{5.10a}$$

$$([v]_{m-1}, v_{m-1}^+) = \frac{1}{2}\|v_{m-1}^+\|^2 + \frac{1}{2}\|[v]_{m-1}\|^2 - \frac{1}{2}\|v_{m-1}^-\|^2, \tag{5.10b}$$

we achieve

$$\frac{1}{2}\|\nabla u_{k,m}^-\|^2 + \frac{1}{2}\|[\nabla u_k]_{m-1}\|^2 - \frac{1}{2}\|\nabla u_{k,m-1}^-\|^2 + \|\Delta u_k\|_{I_m}^2 = (f, -\Delta u_k)_{I_m}.$$

Summation of the equations for $m = 1, 2, \dots, M$ leads to

$$\frac{1}{2}\|\nabla u_{k,M}^-\|^2 + \frac{1}{2}\sum_{m=1}^{M}\|[\nabla u_k]_{m-1}\|^2 + \|\Delta u_k\|_I^2 = (f, -\Delta u_k)_I + \frac{1}{2}\|\nabla u_0\|^2.$$

Using Young's inequality on the right-hand side, we obtain the first intermediary result

$$\|\Delta u_k\|_I^2 \le \|f\|_I^2 + \|\nabla u_0\|^2. \tag{5.11}$$

(ii) To bound the time derivative $\partial_t u_k$, we use the inverse estimate

$$\|v_k\|_{I_m}^2 \le C k_m^{-1} \int_{I_m} (t - t_{m-1})\|v_k\|^2 \, dt, \tag{5.12}$$

which holds true for all functions $v_k \in \mathcal{P}_r(I_m, V)$ and is obtained by a transformation argument. We choose $\varphi = (t - t_{m-1})\partial_t u_k$ and obtain from (5.9) utilizing the fact that $\varphi_{m-1}^+ = 0$:

$$\begin{aligned}\int_{I_m} (t - t_{m-1})\|\partial_t u_k\|^2 \, dt &= \int_{I_m} (t - t_{m-1})(f + \Delta u_k, \partial_t u_k) \, dt \\ &\le \Big(\int_{I_m} (t - t_{m-1})\|f + \Delta u_k\|^2 \, dt\Big)^{\frac{1}{2}} \Big(\int_{I_m} (t - t_{m-1})\|\partial_t u_k\|^2 \, dt\Big)^{\frac{1}{2}}.\end{aligned}$$

The inverse estimate (5.12) yields by means of Hölder's inequality

$$\|\partial_t u_k\|_{I_m}^2 \leq C k_m^{-1} \int_{I_m} (t - t_{m-1}) \|f + \Delta u_k\|^2 \, dt \leq C\{\|f\|_{I_m}^2 + \|\Delta u_k\|_{I_m}^2\}.$$

Then, (5.11) implies the second intermediary result

$$\sum_{m=1}^{M} \|\partial_t u_k\|_{I_m}^2 \leq C\{\|f\|_I^2 + \|\nabla u_0\|^2\}. \tag{5.13}$$

(iii) It remains to estimate the jump terms. Therefor, we choose $\varphi = [u_k]_{m-1}$ (to be understood as function constant with respect to time) and obtain

$$\begin{aligned} \|[u_k]_{m-1}\|^2 &= (f + \Delta u_k - \partial_t u_k, [u_k]_{m-1})_{I_m} \\ &\leq \frac{k_m}{2} \|f + \Delta u_k - \partial_t u_k\|_{I_m}^2 + \frac{1}{2k_m} \|[u_k]_{m-1}\|_{I_m}^2. \end{aligned}$$

Since $[u_k]_{m-1}$ is constant in time, we have $\|[u_k]_{m-1}\|_{I_m}^2 = k_m \|[u_k]_{m-1}\|^2$. This implies

$$k_m^{-1} \|[u_k]_{m-1}\|^2 \leq \|f + \Delta u_k - \partial_t u_k\|_{I_m}^2.$$

The results (5.11) and (5.13) yield the remaining estimate

$$\sum_{m=1}^{M} k_m^{-1} \|[u_k]_{m-1}\|^2 \leq C\{\|f\|_I^2 + \|\nabla u_0\|^2\}. \qquad \square$$

The result of the previous theorem is also applied for the dual (adjoint) equation

$$\begin{aligned} -\partial_t z - \Delta z &= g \qquad \text{in } \Omega \times I, \\ z &= z_T \qquad \text{on } \Omega \times \{\, T \,\}, \end{aligned}$$

with given right-hand side $g \in L^2(I, H)$, terminal condition $z_T \in V$, and homogeneous boundary conditions of Dirichlet or Neumann type. Then, the corresponding semidiscrete dual equation is given by

$$B(\varphi, z_k) = (\varphi, g)_I + (\varphi_M^-, z_T) \quad \forall \varphi \in \widetilde{X}_k^r, \tag{5.14}$$

whereas the fully discretized equation reads as

$$B(\varphi, z_{kh}) = (\varphi, g)_I + (\varphi_M^-, z_T) \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,r}. \tag{5.15}$$

Remark 5.3. The continuous, semidiscrete, and fully discretized adjoint equations of the optimal control problem (P) fit in this formulation by setting $z_T = 0$ and $g = u - \hat{u}$, $g = u_k - \hat{u}$, or $g = u_{kh} - \hat{u}$, respectively.

Corollary 5.4. *For the solution $z_k \in \widetilde{X}_k^r$ of the semidiscrete dual equation* (5.14) *with right-hand side $g \in L^2(I, H)$ and terminal condition $z_T \in V$, the estimate from Theorem 5.3 reads as*

$$\sum_{m=1}^{M} \|\partial_t z_k\|_{I_m}^2 + \|\Delta z_k\|_I^2 + \sum_{m=1}^{M} k_m^{-1} \|[z_k]_m\|^2 \leq C\{\|g\|_I^2 + \|\nabla z_T\|^2\}.$$

Here, the jump term $[z_k]_M$ at $t = T$ is defined as $z_T - z_{k,M}^-$.

Proof. Let $z_k \in \widetilde{X}_k^r$ be the solution of (5.14). Then, formula (5.5) implies that z_k also fulfills for all $\varphi \in \widetilde{X}_k^r$ the following system of equations:

$$-(\varphi, \partial_t z_k)_{I_m} + (\nabla\varphi, \nabla z_k)_{I_m} - (\varphi_m^-, [z_k]_m) = (g, \varphi)_{I_m} \quad m = 1, 2, \dots, M.$$

Based on this representation, all steps of the proof of Theorem 5.3 can be repeated similarly to obtain the stated estimate. □

For proving a priori estimates for the control problem (P), we additionally need stability estimates for the $L^2(I, L^2(\Omega))$-norm of the solution u_k and of its gradient ∇u_k. These are given by the following theorem:

Theorem 5.5. *For the solution $u_k \in \widetilde{X}_k^r$ of the uncontrolled* dG(r)*-semidiscretized state equation* (5.6) *with right-hand side $f \in L^2(I, H)$ and initial condition $u_0 \in V$, the stability estimate*

$$\|u_k\|_I^2 + \|\nabla u_k\|_I^2 \le C\{\|f\|_I^2 + \|\nabla u_0\|^2 + \|u_0\|^2\}$$

holds true with a constant C that only depends on the polynomial degree r, the domain Ω, and the final time T.

Remark 5.4. In the case of homogeneous Dirichlet boundary conditions, the estimate from Theorem 5.5 can be proven by means of Poincaré's inequality with a constant independent of T.

Proof. The proof is done using a duality argument: Let $\tilde{z} \in X$ be the solution of

$$-(\varphi, \partial_t \tilde{z})_I + (\nabla\varphi, \nabla\tilde{z})_I = (\varphi, u_k)_I \quad \forall \varphi \in X$$

with the terminal condition $\tilde{z}_T = 0$. Thus, due to Remark 5.2, $\tilde{z}$ fulfills also

$$B(\varphi, \tilde{z}) = (\varphi, u_k)_I \quad \forall \varphi \in \widetilde{X}_k^r.$$

By means of this equality, we write

$$\|u_k\|_I^2 = B(u_k, \tilde{z}) = \sum_{m=1}^{M} (\partial_t u_k, \tilde{z})_{I_m} + (\nabla u_k, \nabla\tilde{z})_I + \sum_{m=2}^{M} ([u_k]_{m-1}, \tilde{z}(t_{m-1})) + (u_{k,0}^+, \tilde{z}(0)).$$

Usage of the setting $[u_k]_0 = u_{k,0}^+ - u_0$ leads to

$$\|u_k\|_I^2 = \sum_{m=1}^{M} (\partial_t u_k, \tilde{z})_{I_m} + (\nabla u_k, \nabla\tilde{z})_I + \sum_{m=1}^{M} ([u_k]_{m-1}, \tilde{z}(t_{m-1})) + (u_0, \tilde{z}(0)),$$

from which we obtain with integration by parts in space and Hölder's inequality

$$\begin{aligned}\|u_k\|_I^2 \le \left(\sum_{m=1}^{M} \|\partial_t u_k\|_{I_m}^2\right)^{\frac{1}{2}} \|\tilde{z}\|_I + \|\Delta u_k\|_I \|\tilde{z}\|_I \\ + \left(\sum_{m=1}^{M} k_m^{-1} \|[u_k]_{m-1}\|^2\right)^{\frac{1}{2}} \left(\sum_{m=1}^{M} k_m \|\tilde{z}(t_{m-1})\|^2\right)^{\frac{1}{2}} + \|u_0\| \|\tilde{z}(0)\|.\end{aligned}$$

The stability estimate for the continuous solution $\tilde{z} \in X$

$$\max_{t \in \bar{I}} \|\tilde{z}(t)\| \leq C\|u_k\|_I,$$

which makes use of the continuity of the mapping $u_k \mapsto \tilde{z} \in X$ (cf. Lions [53]) and the continuous embedding of X into $C(\bar{I}, H)$, implies

$$\|u_k\|_I \leq C\sqrt{T}\left(\sum_{m=1}^{M} \|\partial_t u_k\|_{I_m}^2\right)^{\frac{1}{2}} + C\sqrt{T}\|\Delta u_k\|_I + C\sqrt{T}\left(\sum_{m=2}^{M} k_m^{-1}\|[u_k]_{m-1}\|^2\right)^{\frac{1}{2}} + C\|u_0\|,$$

from what the desired estimate for $\|u_k\|_I^2$ follows by application of Theorem 5.3.

To prove the estimate for $\|\nabla u_k\|_I^2$, we proceed similarly to the proof of Theorem 5.3 and test (5.8) with $\varphi = u_k$. We obtain for $m = 1, 2, \ldots, M$

$$(\partial_t u_k, u_k)_{I_m} + (\nabla u_k, \nabla u_k)_{I_m} + ([u_k]_{m-1}, u_{k,m-1}^+) = (f, u_k)_{I_m}.$$

The identities (5.10) lead to

$$\frac{1}{2}\|u_{k,m}^-\|^2 + \frac{1}{2}\|[u_k]_{m-1}\|^2 - \frac{1}{2}\|u_{k,m-1}^-\|^2 + \|\nabla u_k\|_{I_m}^2 = (f, u_k)_{I_m}.$$

After summing up these equations for $m = 1, 2, \ldots, M$ and by application of Young's inequality, we have

$$\|\nabla u_k\|_I^2 \leq \frac{1}{2}\{\|f\|_I^2 + \|u_k\|_I^2 + \|u_0\|^2\}.$$

Insertion of the already proven estimate for $\|u_k\|_I^2$ completes the proof. □

Corollary 5.6. *For the solution $z_k \in \widetilde{X}_k^r$ of the semidiscrete dual equation* (5.14) *with right-hand side $g \in L^2(I, H)$ and terminal condition $z_T \in V$, the estimate from Theorem 5.5 reads as*

$$\|z_k\|_I^2 + \|\nabla z_k\|_I^2 \leq C\{\|g\|_I^2 + \|\nabla z_T\|^2 + \|z_T\|^2\}.$$

Proof. The proof is done similarly to the proof of Theorem 5.5. □

All the estimates proven in this section also hold true for the fully discrete cG(s)dG(r) solutions u_{kh}, $z_{kh} \in \widetilde{X}_{k,h}^{r,s}$ of (5.7) and (5.15) almost without any changes. Only two differences have to be regarded: We have to replace the continuous Laplacian Δ by its discrete analog $\Delta_h \colon V_h^s \to V_h^s$ defined by

$$(-\Delta_h u, \varphi) = (\nabla u, \nabla \varphi) \quad \forall \varphi \in V_h^s,$$

and the jump terms $[u_{kh}]_0$ and $[z_{kh}]_M$ are given here by means of the spatial L^2-projection $\mathit{\Pi}_h \colon V \to V_h^s$ as

$$[u_{kh}]_0 = u_{kh,0}^+ - \mathit{\Pi}_h u_0 \qquad \text{and} \qquad [z_{kh}]_M = \mathit{\Pi}_h z_T - z_{kh,M}^-.$$

For the convenience of the reader, we state here the estimates for the fully discrete solution:

Theorem 5.7. *For the solution $u_{kh} \in \widetilde{X}^{r,s}_{k,h}$ of the uncontrolled* cG(s)dG(r)*-discretized state equation* (5.7) *with right-hand side $f \in L^2(I,H)$ and initial condition $u_0 \in V$, the stability estimate*

$$\sum_{m=1}^{M} \|\partial_t u_{kh}\|^2_{I_m} + \|\Delta_h u_{kh}\|^2_I + \sum_{m=1}^{M} k_m^{-1} \|[u_{kh}]_{m-1}\|^2 \leq C\{\|f\|^2_I + \|\nabla \Pi_h u_0\|^2\}$$

holds. The constant C only depends on the polynomial degree r and the domain Ω. The jump term $[u_{kh}]_0$ at $t = 0$ is defined as $u^+_{kh,0} - \Pi_h u_0$. Furthermore, the estimate

$$\|u_{kh}\|^2_I + \|\nabla u_{kh}\|^2_I \leq C\{\|f\|^2_I + \|\nabla \Pi_h u_0\|^2 + \|\Pi_h u_0\|^2\}$$

holds true with a constant C that only depends on the polynomial degree r, the domain Ω and the final time T.

Corollary 5.8. *For the solution $z_{kh} \in \widetilde{X}^{r,s}_{k,h}$ of the discrete dual equation* (5.15) *with right-hand side $g \in L^2(I,H)$ and terminal condition $z_T \in V$, the estimates from Theorem 5.7 read as*

$$\sum_{m=1}^{M} \|\partial_t z_{kh}\|^2_{I_m} + \|\Delta_h z_{kh}\|^2_I + \sum_{m=1}^{M} k_m^{-1} \|[z_{kh}]_m\|^2 \leq C\{\|g\|^2_I + \|\nabla \Pi_h z_T\|^2\}$$

and

$$\|z_{kh}\|^2_I + \|\nabla z_{kh}\|^2_I \leq C\{\|g\|^2_I + \|\nabla \Pi_h z_T\|^2 + \|\Pi_h z_T\|^2\}.$$

Here, the jump term $[z_{kh}]_M$ at $t = T$ is defined as $\Pi_h z_T - z^-_{kh,M}$.

5.3 Error analysis for the state equation

In this section, we prove a priori estimates for the discretization error of the uncontrolled state equation (5.1). Such error estimates can also be found in Eriksson and Johnson [30, 31] and recently in Feistauer and Švadlenka [35]. However, the estimates presented therein are not applicable to the optimal control problem (P) under consideration, since they are formulated either by means of L^∞-norms in time or they measure the error in the norm of $L^2(I,H^1(\Omega))$. Nevertheless, we make use of the ideas presented therein to prove the desired estimates which bound the errors caused by the time and space discretizations in the norm of $Q = L^2(I,L^2(\Omega))$, that is by means of temporal and spatial L^2-norms.

Let $u \in X$ be the solution of the state equation (5.1), $u_k \in \widetilde{X}^r_k$ be the solution of the corresponding semidiscrete equation (5.6), and $u_{kh} \in \widetilde{X}^{r,s}_{k,h}$ be the solution of the fully discretized state equation (5.7) each with $q = 0$. To separate the influences of the space and time discretizations, we split the total discretization error $e := u - u_{kh}$ in its temporal part $e_k := u - u_k$ and its spatial part $e_h := u_k - u_{kh}$. The temporal discretization error is estimated in the following subsection, the spatial discretization error is treated afterwards in Section 5.3.2.

The two main results proven there in the Theorems 5.10 and 5.14 can be summarized in the following corollary:

Corollary 5.9. *For the error $e := u - u_{kh}$ between the continuous solution $u \in X$ of* (5.1) *and the* cG(s)dG(r)*-discretized solution $u_{kh} \in \widetilde{X}^{r,s}_{k,h}$ of* (5.7) *each with $q = 0$, the error estimate*

$$\|e\|_I \leq Ck^{r+1}\|\partial_t^{r+1}u\|_I + Ch^{s+1}\|\nabla^{s+1}u_k\|_I$$

holds with constants C which are independent of the size of the time steps k and the mesh size h.

Throughout this section, we assume that the solutions $u \in X$ and $u_k \in \widetilde{X}^r_k$ possess the regularity $\partial_t^{r+1}u \in L^2(I, H)$ and $\nabla^{s+1}u_k \in L^2(I, H)$. Note, that the Theorems 5.1 and 5.3 ensure this assumption for $r = 0$ and $s = 1$ on convex polygonally bounded domains. Results on higher regularity for $r > 0$ or $s > 1$ usually require stronger assumptions on the domain and additional compatibility conditions between the given initial state and the prescribed boundary conditions; see for instance Wloka [88].

5.3.1 Analysis of the temporal discretization error

In this subsection, we prove the following error estimate for the temporal discretization error e_k:

Theorem 5.10. *For the error $e_k = u - u_k$ between the continuous solution $u \in X$ of* (5.1) *and the* dG(r)*-semidiscretized solution $u_k \in \widetilde{X}^r_k$ of* (5.6) *each with $q = 0$, we have the error estimate*

$$\|e_k\|_I \leq Ck^{r+1}\|\partial_t^{r+1}u\|_I,$$

where the constant C is independent of the size of the time steps k.

For clarity of presentation, we divide the proof of this theorem into several steps, which are discussed in the following lemmas.

We define a semidiscrete projection $\pi_k \colon C(\bar{I}, V) \to \widetilde{X}^r_k$ piecewise for $m = 1, 2, \dots, M$ and $r \in \mathbb{N}$ by

$$\pi_k u\big|_{I_m} \in \mathcal{P}_r(I_m, V), \qquad (\pi_k u - u, \varphi)_{I_m} = 0 \quad \forall \varphi \in \mathcal{P}_{r-1}(I_m, V), \qquad \pi_k u(t_m) = u(t_m).$$

In the case $r = 0$, the projection $\pi_k u$ is determined for $m = 1, 2, \dots, M$ by the two conditions

$$\pi_k u\big|_{I_m} \in \mathcal{P}_0(I_m, V) \quad \text{and} \quad \pi_k u(t_m) = u(t_m).$$

The projection π_k is well-defined by these conditions, see for instance Thomée [76] or Schötzau [73]. We remark here, that due to Theorem 5.1 the solution $u \in X$ of (5.1) belongs also to $C(\bar{I}, V)$ and therefore this projection is applicable to u.

To shorten the notation in the following analysis, we introduce the abbreviations

$$\eta_k := u - \pi_k u \quad \text{and} \quad \xi_k := \pi_k u - u_k,$$

and split the error e_k as

$$e_k = \eta_k + \xi_k.$$

Lemma 5.11. *For the projection error η_k defined above, the identity*

$$B(\eta_k, \varphi) = (\nabla \eta_k, \nabla \varphi)_I$$

holds for all $\varphi \in \widetilde{X}_k^r$.

Proof. By means of (5.5), we have

$$B(\eta_k, \varphi) = -\sum_{m=1}^{M} (\eta_k, \partial_t \varphi)_{I_m} + (\nabla \eta_k, \nabla \varphi)_I - \sum_{m=1}^{M-1} (\eta_{k,m}^-, [\varphi]_m) + (\eta_{k,M}^-, \varphi_{k,M}^-) = (\nabla \eta_k, \nabla \varphi)_I,$$

since the terms $(\eta_k, \partial_t \varphi)_{I_m}$, $(\eta_{k,m}^-, [\varphi]_m)$, and $(\eta_{k,M}^-, \varphi_{k,M}^-)$ vanish due to the definition of π_k. □

Lemma 5.12. *The temporal discretization error $e_k = u - u_k$ is bounded by the projection error η_k in the sense*

$$\|e_k\|_I \le C \|\eta_k\|_I.$$

Proof. We define $\tilde{z}_k \in \widetilde{X}_k^r$ to be the solution of

$$B(\varphi, \tilde{z}_k) = (\varphi, e_k)_I \quad \forall \varphi \in \widetilde{X}_k^r.$$

Thus, we obtain by Galerkin orthogonality (cf. Remark 5.2)

$$\|e_k\|_I^2 = (\xi_k, e_k)_I + (\eta_k, e_k)_I = B(\xi_k, \tilde{z}_k) + (\eta_k, e_k)_I = -B(\eta_k, \tilde{z}_k) + (\eta_k, e_k)_I.$$

Using Lemma 5.11, integration by parts in space, and the stability estimate from Corollary 5.4, it follows

$$-B(\eta_k, \tilde{z}_k) = -(\nabla \eta_k, \nabla \tilde{z}_k)_I = (\eta_k, \Delta \tilde{z}_k)_I \le \|\eta_k\|_I \|\Delta \tilde{z}_k\|_I \le C \|\eta_k\|_I \|e_k\|_I.$$

Note, that again the arising boundary terms vanish for both homogeneous Neumann or homogeneous Dirichlet boundary conditions. This leads by means of Cauchy's inequality to the desired assertion. □

Lemma 5.13. *For the projection error $\eta_k = u - \pi_k u$ the following estimate holds:*

$$\|\eta_k\|_{I_m} \le C k_m^{r+1} \|\partial_t^{r+1} u\|_{I_m}.$$

Proof. Similarly to Thomée [76], the proof is done by standard arguments utilizing the Bramble-Hilbert lemma. □

After these preparations, we can give the proof of Theorem 5.10:

Proof of Theorem 5.10. From the Lemmas 5.12 and 5.13 we obtain directly

$$\|e_k\|_I^2 \le C \|\eta_k\|_I^2 = C \sum_{m=1}^{M} \|\eta_k\|_{I_m}^2 \le C \sum_{m=1}^{M} k_m^{2r+2} \|\partial_t^{r+1} u\|_{I_m}^2 \le C k^{2r+2} \|\partial_t^{r+1} u\|_I^2,$$

which implies the stated result. □

5.3.2 Analysis of the spatial discretization error

In this subsection, we are going to prove the following result for the spatial discretization error on space discretizations which do not vary in time:

Theorem 5.14. *For the error $e_h = u_k - u_{kh}$ between the* dG(r)*-semidiscretized solution $u_k \in \widetilde{X}_k^r$ of* (5.6) *and the fully* cG(s)dG(r)*-discretized solution $u_{kh} \in \widetilde{X}_{k,h}^{r,s}$ of* (5.7) *each with $q = 0$, we have the error estimate*

$$\|e_h\|_I \leq Ch^{s+1} \|\nabla^{s+1} u_k\|_I.$$

Here, the constant C is independent of the mesh size h and the size of the time steps k.

Similar to the subsection before, the proof is divided into several steps which are collected in the following lemmas.

We define the projection $\pi_h \colon \widetilde{X}_k^r \to \widetilde{X}_{k,h}^{r,s}$ pointwise in time by means of the spatial L^2-projection $\mathit{\Pi}_h \colon V \to V_h^s$ as

$$(\pi_h u_k)(t) := \mathit{\Pi}_h u_k(t).$$

For the solutions of the semidiscrete and fully discretized heat equation $u_k \in \widetilde{X}_k^r$ and $u_{kh} \in \widetilde{X}_{k,h}^{r,s}$ and for $\tilde{z}_k \in \widetilde{X}_k^r$ being the solution of the dual equation (5.14) with right-hand side $g = e_h$ and terminal condition $\tilde{z}_T = 0$, we use the abbreviations

$$\eta_h := u_k - \pi_h u_k, \quad \xi_h := \pi_h u_k - u_{kh}, \quad \text{and} \quad \eta_h^* := \tilde{z}_k - \pi_h \tilde{z}_k,$$

and split the error e_h as

$$e_h = \eta_h + \xi_h.$$

Lemma 5.15. *For the projection errors η_h and η_h^* defined above, the identities*

$$B(\eta_h, \varphi) = (\nabla \eta_h, \nabla \varphi)_I \qquad \text{and} \qquad B(\varphi, \eta_h^*) = (\nabla \varphi, \nabla \eta_h^*)_I$$

hold for all $\varphi \in \widetilde{X}_{k,h}^{r,s}$.

Proof. As in the proof of Lemma 5.11, we obtain

$$B(\eta_h, \varphi) = -\sum_{m=1}^{M} (\eta_h, \partial_t \varphi)_{I_m} + (\nabla \eta_h, \nabla \varphi)_I - \sum_{m=1}^{M-1} (\eta_{h,m}^-, [\varphi]_m) + (\eta_{h,M}^-, \varphi_M^-) = (\nabla \eta_h, \nabla \varphi)_I$$

by means of the definition of π_h. The assertion for $B(\varphi, \eta_h^*)$ follows immediately when employing representation (5.4) instead of (5.5). □

Lemma 5.16. *For the error ξ_h and the projection error η_h the estimate*

$$\|\nabla \xi_h\|_I \leq \|\nabla \eta_h\|_I$$

holds.

Proof. Like done in Feistauer and Švadlenka [35], we have for all $v \in \widetilde{X}_k^r$ by (5.4) and (5.5):

$$\begin{aligned} B(v,v) = & \sum_{m=1}^{M} (\partial_t v, v)_{I_m} + (\nabla v, \nabla v)_I + \sum_{m=1}^{M-1} ([v]_m, \quad v_m^+) + (v_0^+, v_0^+) \\ B(v,v) = & -\sum_{m=1}^{M} (v, \partial_t v)_{I_m} + (\nabla v, \nabla v)_I + \sum_{m=1}^{M-1} (-v_m^-, [v]_m) + (v_M^-, v_M^-). \end{aligned}$$

We arrive at

$$B(v,v) \geq (\nabla v, \nabla v)_I \quad \forall v \in \widetilde{X}_k^r$$

by adding these two identities. Utilizing the Galerkin orthogonality of the space discretization, we may write

$$\|\nabla \xi_h\|_I^2 = (\nabla \xi_h, \nabla \xi_h)_I \leq B(\xi_h, \xi_h) = -B(\eta_h, \xi_h) = -(\nabla \eta_h, \nabla \xi_h)_I \leq \|\nabla \eta_h\|_I \|\nabla \xi_h\|_I.$$

Division by $\|\nabla \xi_h\|_I$ leads to the asserted result. □

Lemma 5.17. *The projection errors η_h and η_h^* fulfill the following inequality:*

$$B(\eta_h, \eta_h^*) \leq \|\nabla \eta_h\|_I \|\nabla \eta_h^*\|_I + C\|\eta_h\|_I \|e_h\|_I.$$

Proof. Since $\pi_h \tilde{z}_k \in \widetilde{X}_{k,h}^{r,s}$, it holds by Lemma 5.15 and formula (5.5)

$$\begin{aligned} B(\eta_h, \eta_h^*) &= B(\eta_h, \tilde{z}_k) - B(\eta_h, \pi_h \tilde{z}_k) \\ &= B(\eta_h, \tilde{z}_k) - (\nabla \eta_h, \nabla \pi_h \tilde{z}_k)_I \\ &= -\sum_{m=1}^{M} (\eta_h, \partial_t \tilde{z}_k)_{I_m} + (\nabla \eta_h, \nabla \eta_h^*)_I - \sum_{m=1}^{M-1} (\eta_{h,m}^-, [\tilde{z}_k]_m) + (\eta_{h,M}^-, z_{k,M}^-). \end{aligned}$$

Because $\tilde{z}_T = 0$, we may subtract the term $(\eta_{h,M}^-, \tilde{z}_T)$ to obtain by means of the definition $[\tilde{z}_k] = \tilde{z}_T - \tilde{z}_{k,M}^-$ the identity

$$B(\eta_h, \eta_h^*) = -\sum_{m=1}^{M} (\eta_h, \partial_t \tilde{z}_k)_{I_m} + (\nabla \eta_h, \nabla \eta_h^*)_I - \sum_{m=1}^{M} (\eta_{h,m}^-, [\tilde{z}_k]_m). \tag{5.16}$$

Now, we treat the three terms on the right-hand side above separately: For the term containing spatial derivatives, we have immediately

$$(\nabla \eta_h, \nabla \eta_h^*)_I \leq \|\nabla \eta_h\|_I \|\nabla \eta_h^*\|_I. \tag{5.17}$$

By Cauchy's inequality and with the stability estimate from Corollary 5.4, we achieve for the term containing the time derivatives

$$-\sum_{m=1}^{M} (\eta_h, \partial_t \tilde{z}_k)_{I_m} \leq \|\eta_h\|_I \left(\sum_{m=1}^{M} \|\partial_t \tilde{z}_k\|_{I_m}^2 \right)^{\frac{1}{2}} \leq \|\eta_h\|_I \|e_h\|_I. \tag{5.18}$$

For the jump terms, we obtain again by Cauchy's inequality

$$-\sum_{m=1}^{M}(\eta_{h,m}^{-},[\tilde{z}_k]_m) \le \left(\sum_{m=1}^{M} k_m\|\eta_{h,m}^{-}\|^2\right)^{\frac{1}{2}}\left(\sum_{m=1}^{M} k_m^{-1}\|[\tilde{z}_k]_m\|^2\right)^{\frac{1}{2}}.$$

Utilizing the inverse estimate

$$k_m\|\eta_{h,m}^{-}\|^2 \le C\|\eta_h\|_{I_m}^2,$$

which holds true for polynomials in time (cf. Eriksson and Johnson [30]), and the stability estimate from Corollary 5.4, we obtain

$$-\sum_{m=1}^{M}(\eta_{h,m}^{-},[\tilde{z}_k]_m) \le C\|\eta_h\|_I\|e_h\|_I. \tag{5.19}$$

We complete the proof by inserting the three estimates (5.17), (5.18), and (5.19) into (5.16). □

We are now prepared to give the proof of Theorem 5.14:

Proof of Theorem 5.14. The solution $\tilde{z}_k \in \widetilde{X}_k^r$ is determined by

$$B(\varphi,\tilde{z}_k) = (\varphi, e_h)_I \quad \forall \varphi \in \widetilde{X}_k^r.$$

Due to Galerkin orthogonality, which is applicable for $\pi_h\tilde{z}_k \in \widetilde{X}_{k,h}^{r,s}$, the identity

$$\|e_h\|_I^2 = B(e_h,\tilde{z}_k) = B(e_h,\tilde{z}_k - \pi_h\tilde{z}_k) = B(\xi_h,\eta_h^*) + B(\eta_h,\eta_h^*)$$

is fulfilled. For the first term on the right-hand side, we obtain using the Lemmas 5.15 and 5.16:

$$B(\xi_h,\eta_h^*) = (\nabla\xi_h,\nabla\eta_h^*)_I \le \|\nabla\xi_h\|_I\|\nabla\eta_h^*\|_I \le \|\nabla\eta_h\|_I\|\nabla\eta_h^*\|_I.$$

This yields together with Lemma 5.17 applied to the second term on the right-hand side

$$\|e_h\|_I^2 \le 2\|\nabla\eta_h\|_I\|\nabla\eta_h^*\|_I + C\|\eta_h\|_I\|e_h\|_I. \tag{5.20}$$

Due to the definition of π_h, well-known a priori estimates for the spatial L^2-projection $\mathit{\Pi}_h$ can be employed to directly obtain estimates for the projection errors η_h and η_h^*. We have

$$\|\eta_h\|_I \le Ch^{s+1}\|\nabla^{s+1}u_k\|_I, \quad \|\nabla\eta_h\|_I \le Ch^s\|\nabla^{s+1}u_k\|_I, \quad \text{and} \quad \|\nabla\eta_h^*\|_I \le Ch\|\nabla^2\tilde{z}_k\|_I.$$

These estimates applied to (5.20) lead to

$$\|e_h\|_I^2 \le Ch^{s+1}\|\nabla^{s+1}u_k\|_I\{\|\nabla^2\tilde{z}_k\|_I + \|e_h\|_I\}.$$

Due to the fact that the domain $\mathit{\Omega}$ is assumed to be polygonal and convex, elliptic regularity theory yields

$$\|\nabla^2\tilde{z}_k\|_I \le C\|\Delta\tilde{z}_k\|_I,$$

and we obtain the stated result by means of the stability estimate from Corollary 5.4. □

5.4 Error analysis for the optimal control problem

In this section, we prove the main results of this chapter, namely the estimation for the error in the control, the state, and the adjoint state variables for the optimal control problem (P) as concrete formulation of the general optimization problem ($\mathbb{P}$). Moreover, we derive an estimate of the error in terms of the cost functional. In what follows, we make use of the control problems ($\widetilde{\mathrm{P}}_k$), ($\widetilde{\mathrm{P}}_{kh}$), and ($\widetilde{\mathrm{P}}_\sigma$) which are defined on the different levels of discretization as the concretizations of the abstract optimization problems ($\widetilde{\mathbb{P}}_k$), ($\widetilde{\mathbb{P}}_{kh}$), and ($\widetilde{\mathbb{P}}_\sigma$) by means of (P).

Throughout this section, we indicate the dependence of the state and the adjoint state on a specific control $q \in Q$ by notations like $u(q)$, $z(q)$ on the continuous level, $u_k(q)$, $z_k(q)$ on the semidiscrete, and $u_{kh}(q)$, $z_{kh}(q)$ on the discrete level.

5.4.1 Error in the control variable

The techniques used in the following proofs are already successfully employed to prove error estimates in the context of optimal control problems governed by elliptic equations, see for instance Vexler [83], Becker and Vexler [13], or Rösch and Vexler [71]. The stability estimates derived in Section 5.2 and the error estimates for the state equation from the previous section make it possible to apply these techniques to the here considered case of parabolic governing equations.

The main result of this chapter is formulated in the following theorem:

Theorem 5.18. *The error between the solution $q \in Q$ of the continuous optimal control problem* (P) *and the solution $q_\sigma \in Q_d$ of the associated discrete optimal control problem* ($\widetilde{\mathrm{P}}_\sigma$) *with* cG($s$)dG($r$) *state discretization can be estimated as*

$$\|q - q_\sigma\|_I \le \frac{C}{\alpha} k^{r+1} \{\|\partial_t^{r+1} u(q)\|_I + \|\partial_t^{r+1} z(q)\|_I\} \\ + \frac{C}{\alpha} h^{s+1} \{\|\nabla^{s+1} u_k(q)\|_I + \|\nabla^{s+1} z_k(q)\|_I\} + \left(2 + \frac{C}{\alpha}\right) \inf_{p_d \in Q_d} \|\hat{q} - p_d\|_I,$$

where $\hat{q} \in Q$ can be chosen either as the continuous solution q of (P) *or as the solution q_{kh} of the purely state-discretized problem* ($\widetilde{\mathrm{P}}_{kh}$)*. The arising constants are independent of the mesh size h, the size of the time steps k and the choice of the discrete control space $Q_d \subseteq Q$.*

We first discuss the infimum term appearing on the right-hand side of the error estimate above. Thereby, we make use of the two possible formulation of this term using $\hat{q} = q$ or $\hat{q} = q_{kh}$.

By inspection of the optimality condition for problem ($\widetilde{\mathrm{P}}_{kh}$) with discrete state and continuous control

$$(q_{kh}, \delta q)_I = \frac{1}{\alpha} (z_{kh}(q_{kh}), \delta q)_I \quad \forall \delta q \in Q,$$

the optimal control q_{kh} satisfies $q_{kh} = \frac{1}{\alpha} z_{kh} \in \widetilde{X}_{k,h}^{r,s} \subseteq Q$. Thus, if Q_d is chosen such that $Q_d \supseteq \widetilde{X}_{k,h}^{r,s}$, the term

$$\inf_{p_d \in Q_d} \|q_{kh} - p_d\|_I$$

vanishes. In this case, the solution q_σ of the fully discretized control problem $(\widetilde{\mathrm{P}}_\sigma)$ coincides with the solution q_{kh} of $(\widetilde{\mathrm{P}}_{kh})$; cf. Hinze [44]. Consequently, it is reasonable to discretize the control here at most as fine as the adjoint state. The same conclusion can be drawn in this case by inspection of the a posteriori error estimates developed in Chapter 6.

If the discrete control space Q_d does not contain the discrete state space $\widetilde{X}_{k,h}^{r,s}$ ($Q_d \not\supseteq \widetilde{X}_{k,h}^{r,s}$), it is desirable to choose $\hat{q} = q$ in the above theorem to obtain an a priori estimate for the infimum term. Concerning the possibilities discussed in Example 3.1 for the model problem under consideration, we obtain for the control discretization done like the state discretization by the cG(s_d)dG(r_d) method from stability and error estimates for the L^2-projection $\pi_d \colon Q \to Q_d$

$$\inf_{p_d \in Q_d} \|q - p_d\|_I \leq \|q - \pi_d q\|_I \leq C k_d^{r_d+1} \|\partial_t^{r_d+1} q\|_I + C h_d^{s_d+1} \|\nabla^{s_d+1} q\|_I.$$

By the same arguments, we obtain for the discretization of the control by means of the dG(0)dG(r_d) discretization the estimate

$$\inf_{p_d \in Q_d} \|q - p_d\|_I \leq C k_d^{r_d+1} \|\partial_t^{r_d+1} q\|_I + C h_d \|\nabla q\|_I.$$

Here, k_d and h_d indicate a possibly coarser time and space discretization for the control than for the state.

The proof of Theorem 5.18 makes use of assertions formulated in the following lemmas and is given at the end of this section.

Lemma 5.19. *Let $q \in Q$ be a given control. The error between the continuous state $u = u(q) \in X$ determined by* (5.1) *and the discrete state $u_{kh} = u_{kh}(q) \in \widetilde{X}_{k,h}^{r,s}$ determined by the corresponding discrete state equation* (5.7) *can be estimated as*

$$\|u(q) - u_{kh}(q)\|_I \leq C k^{r+1} \|\partial_t^{r+1} u(q)\|_I + C h^{s+1} \|\nabla^{s+1} u_k(q)\|_I.$$

For the error between the continuous adjoint state $z = z(q) \in X$ and the discrete state $z_{kh} = z_{kh}(q) \in \widetilde{X}_{k,h}^{r,s}$, we have

$$\begin{aligned} \|z(q) - z_{kh}(q)\|_I \leq C k^{r+1} \{ \|\partial_t^{r+1} u(q)\|_I + \|\partial_t^{r+1} z(q)\|_I \} \\ + C h^{s+1} \{ \|\nabla^{s+1} u_k(q)\|_I + \|\nabla^{s+1} z_k(q)\|_I \}. \end{aligned}$$

Proof. The estimate for the error in terms of the state variable is immediately obtained by splitting the error as

$$\|u(q) - u_{kh}(q)\|_I \leq \|u(q) - u_k(q)\|_I + \|u_k(q) - u_{kh}(q)\|_I$$

and applying the Theorems 5.10 and 5.14 to $\|u(q) - u_k(q)\|_I$ and $\|u_k(q) - u_{kh}(q)\|_I$ for the right-hand side $f + q \in L^2(I, H)$ instead of f.

For estimating the error in z, we split again

$$\|z(q) - z_{kh}(q)\|_I \leq \|z(q) - z_k(q)\|_I + \|z_k(q) - z_{kh}(q)\|_I$$

and introduce the solutions $\tilde{z}_k \in \widetilde{X}_k^r$ and $\tilde{z}_{kh} \in \widetilde{X}_{k,h}^{r,s}$ solving

$$B(\varphi, \tilde{z}_k) = (\varphi, u(q) - \hat{u})_I \quad \forall \varphi \in \widetilde{X}_k^r \quad \text{and} \quad B(\varphi, \tilde{z}_{kh}) = (\varphi, u_k(q) - \hat{u})_I \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s}.$$

Since in the considered model situation the adjoint state $z(q) \in X$ is determined by the adjoint equation

$$-(\varphi, \partial_t z(q))_I + (\nabla \varphi, \nabla z(q))_I = (\varphi, u(q) - \hat{u})_I \quad \forall \varphi \in X,$$

we may apply Theorem 5.10 to obtain

$$\|z(q) - \tilde{z}_k\|_I \leq C k^{r+1} \|\partial_t^{r+1} z(q)\|_I. \tag{5.21}$$

Correspondingly, due to the definition of the semidiscrete adjoint solution $z_k(q) \in \widetilde{X}_k^r$ by

$$B(\varphi, z_k(q)) = (\varphi, u_k(q) - \hat{u})_I \quad \forall \varphi \in \widetilde{X}_k^r,$$

Theorem 5.14 yields the estimate

$$\|z_k(q) - \tilde{z}_{kh}\|_I \leq C h^{s+1} \|\nabla^{s+1} z_k(q)\|_I. \tag{5.22}$$

Furthermore, the difference $\tilde{z}_k - z_k(q)$ solves

$$B(\varphi, \tilde{z}_k - z_k(q)) = (\varphi, u(q) - u_k(q))_I \quad \forall \varphi \in \widetilde{X}_k^r,$$

and the stability estimate from Corollary 5.4 implies together with Theorem 5.10

$$\|\tilde{z}_k - z_k(q)\|_I \leq C \|u(q) - u_k(q)\|_I \leq C k^{r+1} \|\partial_t^{r+1} u(q)\|_I. \tag{5.23}$$

Since $z_{kh}(q) \in \widetilde{X}_{k,h}^{r,s}$ is the solution of the discrete adjoint equation

$$B(\varphi, z_{kh}(q)) = (\varphi, u_{kh}(q) - \hat{u})_I \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s},$$

the difference $\tilde{z}_{kh} - z_{kh}(q)$ fulfills

$$B(\varphi, \tilde{z}_{kh} - z_{kh}(q)) = (\varphi, u_k(q) - u_{kh}(q))_I \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s}.$$

In the same way as before, the stability estimate from Corollary 5.8 yields together with Theorem 5.14

$$\|\tilde{z}_{kh} - z_{kh}(q)\|_I \leq C \|u_k(q) - u_{kh}(q)\|_I \leq C h^{s+1} \|\nabla^{s+1} u_k(q)\|_I. \tag{5.24}$$

Then, the estimates (5.21), (5.22), (5.23), and (5.24) lead to the proposed result. □

Lemma 5.20. *For given controls q, $r \in Q$, the difference between the derivatives of the continuous reduced functional j and the discrete reduced cost functional j_{kh} can be estimated as*

$$|j'(q)(r) - j'_{kh}(q)(r)| \leq \|z(q) - z_{kh}(q)\|_I \|r\|_I.$$

Proof. The representations for j' and j'_{kh} given here as concretizations of (2.10) by

$$j'(q)(r) = \alpha(q,r)_I - (z(q),r)_I \quad \text{and} \quad j'_{kh}(q)(r) = \alpha(q,r)_I - (z_{kh}(q),r)_I \tag{5.25}$$

imply directly the assertion:

$$|j'(q)(r) - j'_{kh}(q)(r)| = |(z(q) - z_{kh}(q),r)_I| \leq \|z(q) - z_{kh}(q)\|_I \|r\|_I. \qquad \square$$

Lemma 5.21. *The derivatives of the discrete reduced cost functional j_{kh} are Lipschitz continuous on Q. That is, for arbitrary p, q, $r \in Q$, the estimate*

$$|j'_{kh}(q)(r) - j'_{kh}(p)(r)| \leq (C+\alpha)\|q-p\|_I\|r\|_I$$

holds true.

Proof. By means of (5.25), we have again

$$\begin{aligned} |j'_{kh}(q)(r) - j'_{kh}(p)(r)| &\leq \alpha|(q-p,r)_I| + |(z_{kh}(q) - z_{kh}(p),r)_I| \\ &\leq \alpha\|q-p\|_I\|r\|_I + \|z_{kh}(q) - z_{kh}(p)\|_I\|r\|_I. \end{aligned}$$

Since $z_{kh}(q) - z_{kh}(p)$ solves

$$B(\varphi, z_{kh}(q) - z_{kh}(p)) = (\varphi, u_{kh}(q) - u_{kh}(p))_I \quad \forall \varphi \in \widetilde{X}^{r,s}_{k,h},$$

and $u_{kh}(q) - u_{kh}(p)$ satisfies

$$B(u_{kh}(q) - u_{kh}(p), \varphi) = (q-p,\varphi)_I \quad \forall \varphi \in \widetilde{X}^{r,s}_{k,h},$$

the stability estimate for z_{kh} from Corollary 5.8 and for u_{kh} from Theorem 5.7 yield

$$\|z_{kh}(q) - z_{kh}(p)\|_I \leq C\|u_{kh}(q) - u_{kh}(p)\|_I \leq C\|q-p\|_I,$$

which implies the desired result. $\square$

With the aid of these preliminary results, we now proof Theorem 5.18:

Proof of Theorem 5.18. To obtain the asserted result, we split the error to be estimated in two different ways:

$$\|q - q_\sigma\|_I \leq \|q - p_d\|_I + \|p_d - q_\sigma\|_I, \tag{5.26}$$
$$\|q - q_\sigma\|_I \leq \|q - q_{kh}\|_I + \|q_{kh} - p_d\|_I + \|p_d - q_\sigma\|_I. \tag{5.27}$$

Here, p_d is an arbitrary element of Q_d and q, q_{kh}, and q_σ are the optimal solutions of (P), ($\widetilde{\text{P}}_{kh}$), and ($\widetilde{\text{P}}_\sigma$) on the different levels of discretization.

Due to the linear-quadratic structure of the control problem under consideration, the second order sufficient optimality condition holds. That is, we have for all $p, r \in Q$

$$j''_{kh}(p)(r,r) \geq \alpha\|r\|^2_I,$$

and the derivative $j''_{kh}(p)$ does not depend on p. This implies for arbitrary $p \in Q$, $p_d \in Q_d$

$$\alpha \|p_d - q_\sigma\|_I^2 \leq j''_{kh}(p)(p_d - q_\sigma, p_d - q_\sigma) = j'_{kh}(p_d)(p_d - q_\sigma) - j'_{kh}(q_\sigma)(p_d - q_\sigma).$$

Since q, q_{kh}, and q_σ are the optimal solutions of the continuous, semidiscrete, and discrete optimal control problems and $p_d - q_\sigma$ is an element of the discrete control space Q_d, we have

$$j'_{kh}(q_\sigma)(p_d - q_\sigma) = j'_{kh}(q_{kh})(p_d - q_\sigma) = j'(q)(p_d - q_\sigma) = 0.$$

Using these identities, we obtain for separation (5.26) the estimate

$$\begin{aligned} \alpha \|p_d - q_\sigma\|_I^2 &\leq j'_{kh}(p_d)(p_d - q_\sigma) - j'(q)(p_d - q_\sigma) \\ &= j'_{kh}(p_d)(p_d - q_\sigma) - j'_{kh}(q)(p_d - q_\sigma) + j'_{kh}(q)(p_d - q_\sigma) - j'(q)(p_d - q_\sigma), \end{aligned}$$

which we use to prove the theorem in the case $\hat{q} = q$. By means of the Lemmas 5.20 and 5.21, we achieve

$$\alpha \|p_d - q_\sigma\|_I^2 \leq (C + \alpha)\|p_d - q\|_I \|p_d - q_\sigma\|_I + \|z(q) - z_{kh}(q)\|_I \|p_d - q_\sigma\|_I.$$

Using (5.26), we get the estimate

$$\|q - q_\sigma\|_I \leq \frac{1}{\alpha}\|z(q) - z_{kh}(q)\|_I + \left(2 + \frac{C}{\alpha}\right)\|q - p_d\|_I. \tag{5.28}$$

To use separation (5.27) for proving the theorem in the case $\hat{q} = q_{kh}$, we estimate alternatively by means of Lemma 5.21

$$\alpha \|p_d - q_\sigma\|_I^2 \leq j'_{kh}(p_d)(p_d - q_\sigma) - j'_{kh}(q_{kh})(p_d - q_\sigma) \leq (C + \alpha)\|p_d - q_{kh}\|_I \|p_d - q_\sigma\|_I.$$

In the same manner as before, we can estimate $\|q - q_{kh}\|_I$ using Lemma 5.20 as

$$\begin{aligned} \alpha \|q - q_{kh}\|_I^2 &\leq j''_{kh}(p)(q - q_{kh}, q - q_{kh}) \\ &= j'_{kh}(q)(q - q_{kh}) - j'_{kh}(q_{kh})(q - q_{kh}) \\ &= j'_{kh}(q)(q - q_{kh}) - j'(q)(q - q_{kh}) \\ &\leq \|z(q) - z_{kh}(q)\|_I \|q - q_{kh}\|_I, \end{aligned}$$

since we have for $q - q_{kh} \in Q$

$$j'_{kh}(q_{kh})(q - q_{kh}) = j'(q)(q - q_{kh}) = 0.$$

Then, the two latter estimates imply together with (5.27)

$$\|q - q_\sigma\|_I \leq \frac{1}{\alpha}\|z(q) - z_{kh}(q)\|_I + \left(2 + \frac{C}{\alpha}\right)\|q_{kh} - p_d\|_I. \tag{5.29}$$

Finally, the inequalities (5.28) and (5.29) prove the assertion by means of the estimate for $\|z(q) - z_{kh}(q)\|_I$ from Lemma 5.19. □

To concretize the result of Theorem 5.18, we consider the following choice of discretizations: The state space is discretized by the cG(1)dG(0) method, that is we consider the case $r = 0$ and $s = 1$. The time discretization of the control space is chosen as for the state space, that is piecewise constant discontinuous polynomials in time (dG(0)). For the space discretization of the controls, both possibilities discussed in Example 3.1 are examined:

- discretization by continuous piecewise (bi-/tri-)linear polynomials (cG(1))
- discretization by discontinuous piecewise constant polynomials (dG(0))

Thereby we assume that the control discretization uses the same triangulation of the spatial domain and the same distribution of the time steps as the discretization of the states, that is $h_d = h$ and $k_d = k$.

For the dG(0)dG(0) discretization of the controls, the infimum term of the error estimation from Theorem 5.18 has to be taken into account, whereas for the cG(1)dG(0) discretization it is zero since the discrete control space equals the discrete state space; see the discussion at the beginning of this subsection.

Thus, Theorem 5.18 implies for the discretization error to be of order

$$\|q - q_\sigma\|_I = \mathcal{O}(k + h^2)$$

for the cG(1)dG(0) discretization, and to be of order

$$\|q - q_\sigma\|_I = \mathcal{O}(k + h)$$

for the dG(0)dG(0) discretization case. Note, that the regularity of the optimal solutions required for these estimates is ensured by the Theorems 5.1 and 5.2 for the continuous solutions q, u, and z and by Theorem 5.3 and Corollary 5.4 for the time-discrete solutions u_k and z_k.

A numerical validation of these error estimates is given in Section 5.5.

5.4.2 Error in the state and adjoint state variable

In this subsection, we are going to prove error estimates for the optimal state and the corresponding adjoint state. That is, we consider the discretization errors

$$\|u - u_\sigma\|_I = \|u(q) - u_{kh}(q_\sigma)\|_I \quad \text{and} \quad \|z - z_\sigma\|_I = \|z(q) - z_{kh}(q_\sigma)\|_I$$

for the choice of discretizations described at the end of the previous subsection.

A first estimate of the error $\|u - u_\sigma\|_I$ between the state $u = u(q)$ associated with the continuous optimal control q and the discrete state $u_\sigma = u_{kh}(q_\sigma)$ associated with the discrete optimal control q_σ can be derived by means of the stability of the discrete state u_{kh}, a priori estimates for the error caused by the discretization of the state space, and a priori estimates for the error in the control: We have by the definitions of u and u_σ

$$\|u - u_\sigma\|_I \leq \|u(q) - u_{kh}(q)\|_I + \|u_{kh}(q) - u_{kh}(q_\sigma)\|_I. \tag{5.30}$$

The second term on the right-hand side of (5.30) can be estimated using the stability of the solution $u_{kh}(q) - u_{kh}(q_\sigma)$ for the right-hand side $q - q_\sigma$ and the initial condition $u_0 = 0$ from Theorem 5.7, that is

$$\|u_{kh}(q) - u_{kh}(q_\sigma)\|_I \leq C\|q - q_\sigma\|_I.$$

With (5.30), we obtain the following theorem:

Theorem 5.22. *Let* $(q, u) \in Q \times X$ *be the solution of the continuous control problem* (P) *and* $(q_\sigma, u_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{r,s}$ *the solution of the corresponding discrete control problem* $(\widetilde{\mathrm{P}}_\sigma)$*. Then, the following estimate holds:*

$$\|u - u_\sigma\|_I \leq \|u(q) - u_{kh}(q)\|_I + C\|q - q_\sigma\|_I.$$

A similar result holds for the error in the adjoint variable z:

Corollary 5.23. *We consider the configuration of Theorem 5.22. Let* $z = z(q) \in X$ *be the continuous adjoint state for* (P) *and* $z_\sigma = z_{kh}(q_\sigma)$ *be the discrete adjoint state for* $(\widetilde{\mathrm{P}}_\sigma)$*. Then, it holds:*

$$\|z - z_\sigma\|_I \leq \|z(q) - z_{kh}(q)\|_I + C\|q - q_\sigma\|_I.$$

Proof. For $\|z - z_\sigma\|_I$ we obtain directly by means of the stability results from Corollary 5.8 and Theorem 5.7:

$$\begin{aligned}\|z - z_\sigma\|_I &\leq \|z(q) - z_{kh}(q)\|_I + \|z_{kh}(q) - z_{kh}(q_\sigma)\|_I \\ &\leq \|z(q) - z_{kh}(q)\|_I + C\|u_{kh}(q) - u_{kh}(q_\sigma)\|_I \\ &\leq \|z(q) - z_{kh}(q)\|_I + C\|q - q_\sigma\|_I.\end{aligned}$$ □

When discretizing the control by cG(1)dG(0), these estimates lead to optimal orders of convergence using the assertions

$$\|q - q_\sigma\|_I = \mathcal{O}(k + h^2), \quad \|u(q) - u_{kh}(q)\|_I = \mathcal{O}(k + h^2) \quad \text{and} \quad \|z(q) - z_{kh}(q)\|_I = \mathcal{O}(k + h^2),$$

from Theorem 5.18 and Lemma 5.19. Thus, we have

$$\|u - u_\sigma\|_I = \mathcal{O}(k + h^2) \quad \text{and} \quad \|z - z_\sigma\|_I = \mathcal{O}(k + h^2).$$

However, in the case of dG(0)dG(0) discretization, these simple estimate do not lead to the optimal orders of convergence: In this case, we have indeed as before

$$\|u(q) - u_{kh}(q)\|_I = \mathcal{O}(k + h^2) \quad \text{and} \quad \|z(q) - z_{kh}(q)\|_I = \mathcal{O}(k + h^2)$$

since the discretization of the state space is unaffected by the discretization of the controls, but we only have

$$\|q - q_\sigma\|_I = \mathcal{O}(k + h)$$

due to the lower order discretization of the control space. This would lead to $\mathcal{O}(k + h)$-convergence for the state and adjoint state variable.

Utilizing a more detailed analysis, we can prove also in this case the optimal oder of convergence $\mathcal{O}(k+h^2)$ for the errors $\|u-u_\sigma\|_I$ and $\|z-z_\sigma\|_I$. For that purpose, we again consider the two mentioned types of control discretizations and introduce the space-time L^2-projection $\pi_d\colon Q \to Q_d$ to split the second term in (5.30) as

$$\|u_{kh}(q)-u_{kh}(q_\sigma)\|_I \le \|u_{kh}(q)-u_{kh}(\pi_d q)\|_I + \|u_{kh}(\pi_d q)-u_{kh}(q_\sigma)\|_I. \tag{5.31}$$

The first term on the right-hand side of (5.31) is estimated using a duality argument: Let $\tilde{z}_{kh} \in \widetilde{X}^{0,1}_{k,h}$ be the solution of

$$B(\varphi, \tilde{z}_{kh}) = (\varphi, u_{kh}(q)-u_{kh}(\pi_d q))_I \quad \forall \varphi \in \widetilde{X}^{0,1}_{k,h}.$$

By means of the state equation for $u_{kh}(q)-u_{kh}(\pi_d q)$, we have

$$\|u_{kh}(q)-u_{kh}(\pi_d q)\|_I^2 = B(u_{kh}(q)-u_{kh}(\pi_d q), \tilde{z}_{kh}) = (q-\pi_d q, \tilde{z}_{kh})_I.$$

Since π_d is the L^2-projection, we may insert $\pi_d \tilde{z}_{kh} \in Q_d$ to obtain

$$\|u_{kh}(q)-u_{kh}(\pi_d q)\|_I^2 = (q-\pi_d q, \tilde{z}_{kh}-\pi_d \tilde{z}_{kh})_I \le \|q-\pi_d q\|_I \|\tilde{z}_{kh}-\pi_d \tilde{z}_{kh}\|_I. \tag{5.32}$$

Employing the fact that the same time discretization is used for the control and the adjoint state variable, the space-time L^2-projection π_d applied to $\tilde{z}_{kh}$ can be expressed as spatial L^2-projection $\mathit{\Pi}_d \tilde{z}_{kh}$. Here, we have to distinguish the two considered cases of control discretizations:

- In the case of cG(1)dG(0) discretization, we have $\pi_d \tilde{z}_{kh} = \mathit{\Pi}_d \tilde{z}_{kh} = \tilde{z}_{kh}$ and thus
$$\|\tilde{z}_{kh}-\pi_d \tilde{z}_{kh}\|_I = 0.$$
- For dG(0)dG(0) discretization we obtain by estimating the projection error
$$\|\tilde{z}_{kh}-\pi_d \tilde{z}_{kh}\|_I = \|\tilde{z}_{kh}-\mathit{\Pi}_d \tilde{z}_{kh}\|_I \le Ch\|\nabla \tilde{z}_{kh}\|_I.$$

Accordingly, by the stability estimate of Corollary 5.8 for $\|\nabla \tilde{z}_{kh}\|_I$, we achieve in both cases

$$\|\tilde{z}_{kh}-\pi_d \tilde{z}_{kh}\|_I \le Ch\|u_{kh}(q)-u_{kh}(\pi_d q)\|_I.$$

Plugging this into (5.32) yields for the first term on the right-hand side of (5.31)

$$\|u_{kh}(q)-u_{kh}(\pi_d q)\|_I \le Ch\|q-\pi_d q\|_I. \tag{5.33}$$

For the second term on the right-hand side of (5.31), we obtain due to Theorem 5.7

$$\|u_{kh}(\pi_d q)-u_{kh}(q_\sigma)\|_I \le C\|\pi_d q-q_\sigma\|_I.$$

Now, it remains to derive an estimate for $\|\pi_d q-q_\sigma\|_I$. In the same way as done for $\|p_d-q_\sigma\|_I$ in the proof of Theorem 5.18, we have

$$\alpha\|\pi_d q-q_\sigma\|_I^2 \le j'_{kh}(\pi_d q)(\pi_d q-q_\sigma) - j'(q)(\pi_d q-q_\sigma).$$

By using the representation (5.25) of j' and j'_{kh} in terms of the adjoint state we get

$$\alpha\|\pi_d q - q_\sigma\|_I^2 \leq \alpha(\pi_d q - q, \pi_d q - q_\sigma)_I + (z_{kh}(\pi_d q) - z(q), \pi_d q - q_\sigma)_I.$$

Since $\pi_d q - q_\sigma \in Q_d$, the term $(\pi_d q - q, \pi_d q - q_\sigma)_I$ vanishes, and due to Corollary 5.8 we end up with

$$\begin{aligned} \alpha\|\pi_d q - q_\sigma\|_I &\leq \|z_{kh}(\pi_d q) - z(q)\|_I \\ &\leq \|z_{kh}(\pi_d q) - z_{kh}(q)\|_I + \|z_{kh}(q) - z(q)\|_I \\ &\leq C\|u_{kh}(\pi_d q) - u_{kh}(q)\|_I + \|z_{kh}(q) - z(q)\|_I, \end{aligned} \tag{5.34}$$

which implies by (5.33) the estimate

$$\begin{aligned} \|u_{kh}(\pi_d q) - u_{kh}(q_\sigma)\|_I &\leq \frac{C}{\alpha}\|u_{kh}(\pi_d q) - u_{kh}(q)\|_I + \frac{C}{\alpha}\|z_{kh}(q) - z(q)\|_I \\ &\leq \frac{C}{\alpha}h\|q - \pi_d q\|_I + \frac{C}{\alpha}\|z_{kh}(q) - z(q)\|_I. \end{aligned} \tag{5.35}$$

Plugging (5.33) and (5.35) in (5.31), estimates $\|u_{kh}(q) - u_{kh}(q_\sigma)\|_I$ by

$$\|u_{kh}(q) - u_{kh}(q_\sigma)\|_I \leq Ch\Big(1 + \frac{1}{\alpha}\Big)\|q - \pi_d q\|_I + \frac{C}{\alpha}\|z(q) - z_{kh}(q)\|_I, \tag{5.36}$$

which leads together with (5.30) to the following Theorem:

Theorem 5.24. *The error between the state $u = u(q) \in X$ associated with the solution $q \in Q$ of the continuous optimal control problem* (P) *and the discrete state $u_\sigma = u_{kh}(q_\sigma) \in \widetilde{X}^{0,1}_{k,h}$ associated with the solution $q_\sigma \in Q_d$ of the discrete optimal control problem* $(\widetilde{\mathrm{P}}_\sigma)$ *with* cG(1)dG(0) *state discretization and discretization of the control by* cG(1)dG(0) *or* dG(0)dG(0) *can be estimated as*

$$\|u - u_\sigma\|_I \leq Ch\Big(1 + \frac{1}{\alpha}\Big)\|q - \pi_d q\|_I + \|u(q) - u_{kh}(q)\|_I + \frac{C}{\alpha}\|z(q) - z_{kh}(q)\|_I.$$

The constants are independent of the mesh size h, the size of the time step k and the choice of the discrete control space $Q_d \subseteq Q$.

By means of Lemma 5.19, we have as before

$$\|u(q) - u_{kh}(q)\|_I = \mathcal{O}(k + h^2) \qquad \text{and} \qquad \|z(q) - z_{kh}(q)\|_I = \mathcal{O}(k + h^2).$$

In contrast to the estimate derived at the beginning of this subsection, the estimate from Theorem 5.24 leads to an improved order of convergence since in both considered cases of control discretizations the limiting order of convergence of the error

$$\|q - \pi_d q\|_I \leq Ck\|\partial_t q\|_I + Ch^{s+1}\|\nabla^{s+1} q\|_I \quad \text{for } s \in \{\,0, 1\,\} \tag{5.37}$$

is now enhanced by the gained additional h. Thus, we have in all cases the optimal order of convergence for the state variable, that is

$$\|u - u_\sigma\|_I = \mathcal{O}(k + h^2).$$

By similar techniques, the following convergence result can be obtained for the error in the adjoint solution:

Corollary 5.25. *We consider the configuration of Theorem 5.24. For the error between the adjoint state $z \in X$ associated with $q \in Q$ and the adjoint state $z_\sigma \in \widetilde{X}^{0,1}_{k,h}$ associated with $q_\sigma \in Q_d$, the estimate*

$$\|z - z_\sigma\|_I \leq Ch\Big(1 + \frac{1}{\alpha}\Big)\|q - \pi_d q\|_I + C\Big(1 + \frac{1}{\alpha}\Big)\|z(q) - z_{kh}(q)\|_I$$

holds true.

Proof. We deduce using the stability of the solution z_{kh} of the fully discrete adjoint equation from Corollary 5.8:

$$\begin{aligned}\|z - z_\sigma\|_I &= \|z(q) - z_{kh}(q_\sigma)\|_I\\ &\leq \|z(q) - z_{kh}(q)\|_I + \|z_{kh}(q) - z_{kh}(q_\sigma)\|_I\\ &\leq \|z(q) - z_{kh}(q)\|_I + C\|u_{kh}(q) - u_{kh}(q_\sigma)\|_I.\end{aligned}$$

By means of (5.36), this inequality proves the assertion. □

Thus, Corollary 5.25 implies for the error $\|z - z_\sigma\|_I$ in terms of the adjoint state variable the same order of convergence as for $\|u - u_\sigma\|_I$, that is

$$\|z - z_\sigma\|_I = \mathcal{O}(k + h^2).$$

Numerical experiments confirming these results are given in Section 5.5.

5.4.3 Error in terms of the cost functional

In many applications, the quality of approximation is measured in terms of the cost functional J. There, the error

$$|J(q, u) - J(q_\sigma, u_\sigma)|$$

is of interest. We also aim at this error in the development of goal-oriented a posteriori error estimates presented in Chapter 6.

Due to the structure of J given by (5.2), we have

$$J(q, u) - J(q_\sigma, u_\sigma) = \frac{1}{2}\{\|u - \hat{u}\|_I^2 - \|u_\sigma - \hat{u}\|_I^2\} + \frac{\alpha}{2}\{\|q\|_I^2 - \|q_\sigma\|_I^2\}. \tag{5.38}$$

Thus, we have to consider the order of convergence separately for the u-term and the q-term. At first glance, one might guess that the order of convergence of the error $|J(q, u) - J(q_\sigma, u_\sigma)|$ is limited by the convergence orders of the errors $\|u - u_\sigma\|_I$ and $\|q - q_\sigma\|_I$. However, similarly to the analysis for the discretization error in the state variable, we obtain a better order of convergence for the difference $\|q\|_I^2 - \|q_\sigma\|_I^2$ than for the discretization error $\|q - q_\sigma\|_I$.

Using the identity

$$\|v\|_I^2 - \|w\|_I^2 = 2(v, v-w)_I - \|v-w\|_I^2$$

for functions $v, w \in L^2(I, H)$, we have for the terms on the right-hand side of (5.38):

$$\begin{aligned}\|u-\hat{u}\|_I^2 - \|u_\sigma - \hat{u}\|_I^2 &= 2(u-\hat{u}, u-u_\sigma)_I - \|u-u_\sigma\|_I^2,\\ \|q\|_I^2 - \|q_\sigma\|_I^2 &= 2(q, q-q_\sigma)_I - \|q-q_\sigma\|_I^2.\end{aligned}$$

For the u-term, this implies directly

$$\frac{1}{2}\Big|\|u-\hat{u}\|_I^2 - \|u_\sigma - \hat{u}\|_I^2\Big| \le \{\|u\|_I + \|\hat{u}\|_I\}\|u-u_\sigma\|_I + \frac{1}{2}\|u-u_\sigma\|_I^2.$$

Hence, this term exhibits the optimal order of convergence $\mathcal{O}(k+h^2)$.

Application of the same techniques to the q-part of (5.38), that is the difference $\|q\|_I^2 - \|q_\sigma\|_I^2$, would for the dG(0)dG(0) discretization of the controls not lead to the optimal order of convergence. However, by proceeding as in the previous subsection when proving the $\mathcal{O}(k+h^2)$-convergence of the error $\|u-u_\sigma\|_I$, it is possible to show here optimal order of convergence, too.

By means of the already introduced space-time L^2-projection $\pi_d \colon Q \to Q_d$, we have

$$(q, q-q_\sigma)_I = (q, q-\pi_d q)_I + (q, \pi_d q - q_\sigma)_I = \|q-\pi_d q\|_I^2 + (q, \pi_d q - q_\sigma)_I,$$

and thus

$$|(q, q-q_\sigma)_I| \le \|q-\pi_d q\|_I^2 + \|q\|_I \|\pi_d q - q_\sigma\|_I.$$

Then, the estimates (5.34) and (5.33) imply

$$|(q, q-q_\sigma)_I| \le \|q-\pi_d q\|_I^2 + \frac{C}{\alpha} h \|q\|_I \|q-\pi_d q\| + \frac{1}{\alpha}\|q\|_I \|z(q) - z_{kh}(q)\|_I,$$

and we obtain for the second term on the right-hand side of (5.38) the assessment

$$\frac{\alpha}{2}\Big|\|q\|_I^2 - \|q_\sigma\|_I^2\Big| \le \alpha\|q-\pi_d q\|_I^2 + \frac{\alpha}{2}\|q-q_\sigma\|_I^2 + Ch\|q\|_I\|q-\pi_d q\| + \|q\|_I\|z(q)-z_{kh}(q)\|_I.$$

Consequently, we proved the following theorem:

Theorem 5.26. *Let the cost functional J be defined by* (5.2)*, $(q,u) \in Q \times X$ be the optimal solution of* (P)*, and $(q_\sigma, u_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{0,1}$ be the optimal solution of* ($\widetilde{\mathrm{P}}_\sigma$)*. If the discrete control space is constructed employing* cG(1)dG(0) *or* dG(0)dG(0) *discretizations, the following estimate holds for the error in terms of the cost functional:*

$$\begin{aligned}|J(q,u) - J(q_\sigma,u_\sigma)| &\le \{\|u\|_I + \|\hat{u}\|_I\}\|u-u_\sigma\|_I + \frac{1}{2}\|u-u_\sigma\|_I^2\\ &\quad + \alpha\|q-\pi_d q\|_I^2 + \frac{\alpha}{2}\|q-q_\sigma\|_I^2 + Ch\|q\|_I\|q-\pi_d q\| + \|q\|_I\|z(q)-z_{kh}(q)\|_I = \mathcal{O}(k+h^2).\end{aligned}$$

In particular, for all considered types of discretizations, the two terms building the cost functional, $1/2\|u_\sigma - \hat{u}\|_I^2$ and $\alpha/2\|q_\sigma\|_I^2$, exhibit the same order of convergence.

Proof. It only remains to proof the order of convergence in terms of k and h: The order of convergence $\mathcal{O}(k+h^2)$ for the u-term is obtained by the estimate from Theorem 5.24 for $\|u-u_\sigma\|_I$, whereas the convergence of order $\mathcal{O}(k+h^2)$ for the q-term is implied by (5.37) and the assertions of Theorem 5.18 and Lemma 5.19. □

This result is in particular important for justifying the a posteriori error analysis derived in the following chapter. If the two parts of the cost functional J depending on the state and the control would exhibit different orders of convergence, it would be quite questionable to choose J as a meaningful measure for the approximation quality.

5.5 Numerical results

In this section, we numerically validate the a priori error estimates for the error in the control, state, and adjoint state as well as the error in terms of the cost functional. To this end, we consider the following concretization of the model problem (P) with known analytical solution on $\Omega \times I = (0,1)^2 \times (0,0.1)$ equipped with homogeneous Dirichlet boundary conditions. The right-hand side f, the desired state $\hat{u}$, and the initial condition u_0 are given in terms of the eigenfunctions

$$w_a(t,x_1,x_2) = \exp(a\pi^2 t)\sin(\pi x_1)\sin(\pi x_2), \quad a \in \mathbb{R}$$

of the operator $\pm\partial_t - \Delta$ as

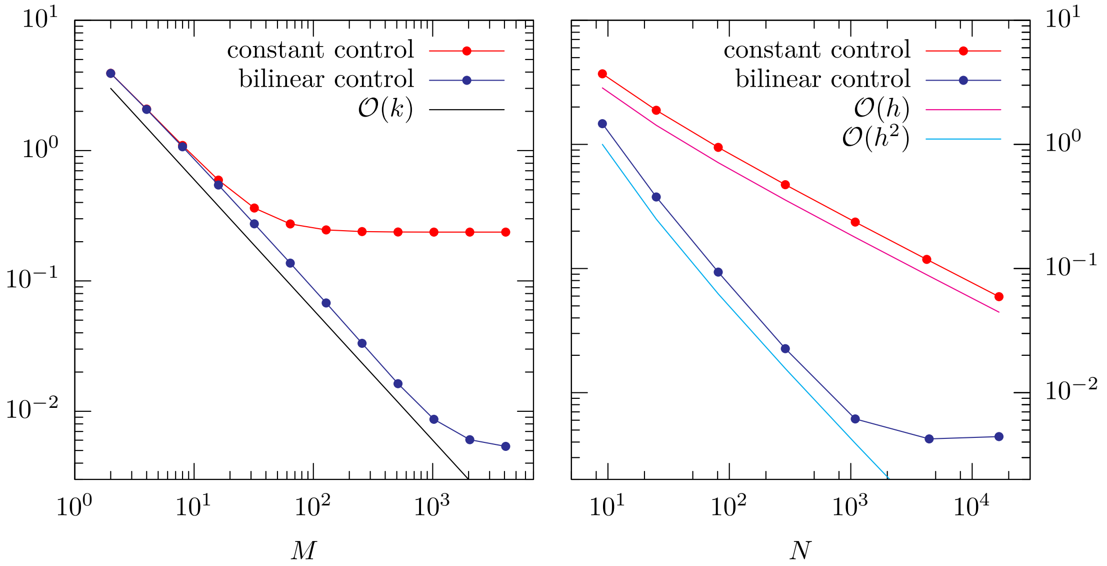
$$\begin{aligned} f(t,x_1,x_2) &= -\pi^4 w_a(T,x_1,x_2),\\ \hat{u}(t,x_1,x_2) &= \frac{a^2-5}{2+a}\pi^2 w_a(t,x_1,x_2) + 2\pi^2 w_a(T,x_1,x_2),\\ u_0(x_1,x_2) &= \frac{-1}{2+a}\pi^2 w_a(0,x_1,x_2). \end{aligned}$$

For this choice of data and with the regularization parameter α chosen as $\alpha = \pi^{-4}$, the optimal solution triple (q,u,z) of the control problem (P) is given by

$$\begin{aligned} q(t,x_1,x_2) &= -\pi^4\{w_a(t,x_1,x_2) - w_a(T,x_1,x_2)\},\\ u(t,x_1,x_2) &= \frac{-1}{2+a}\pi^2 w_a(t,x_1,x_2),\\ z(t,x_1,x_2) &= w_a(t,x_1,x_2) - w_a(T,x_1,x_2). \end{aligned}$$

We validate the estimates developed in the previous section by separating the discretization errors, that is we consider at first the behavior of the error for a sequence of decreasing sizes of the time steps on a fixed spatial triangulation with $N = 1{,}089$ nodes. Secondly, we examine the behavior of the error under refinement of the spatial triangulation for $M = 2{,}048$ time steps.

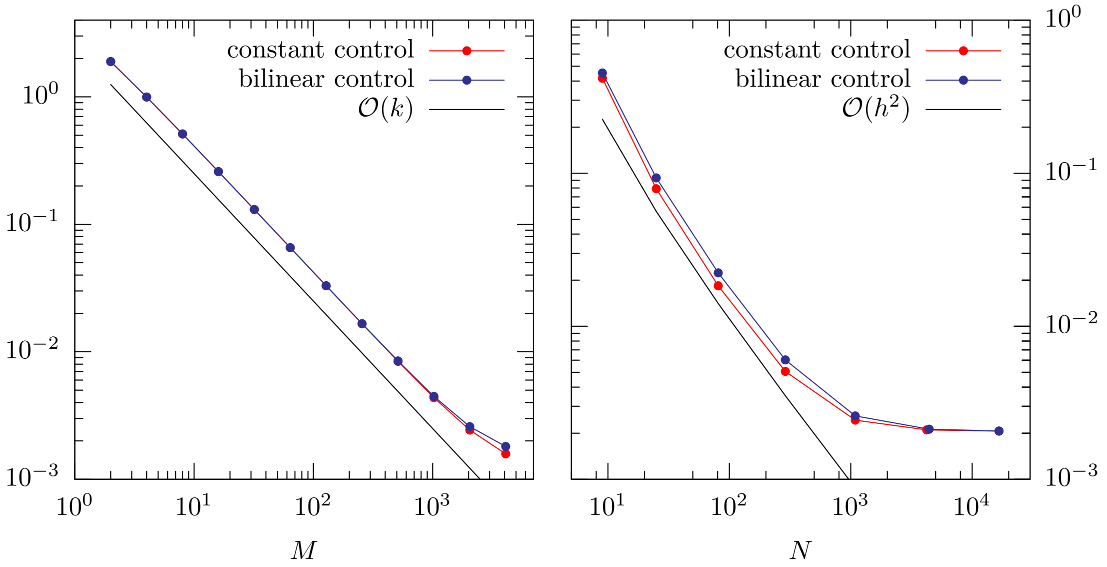
For the following computations, we choose the free parameter a in the definition of w_a to be $-\sqrt{5}$. For this choice, the right-hand side f and the desired state $\hat{u}$ do not depend on time what avoids side effects introduced by numerical quadrature. The considered state and control discretizations are chosen as discussed in the previous subsection.



(a) Refinement of the time steps for $N = 1{,}089$ spatial nodes

(b) Refinement of the spatial triangulation for $M = 2{,}048$ time steps

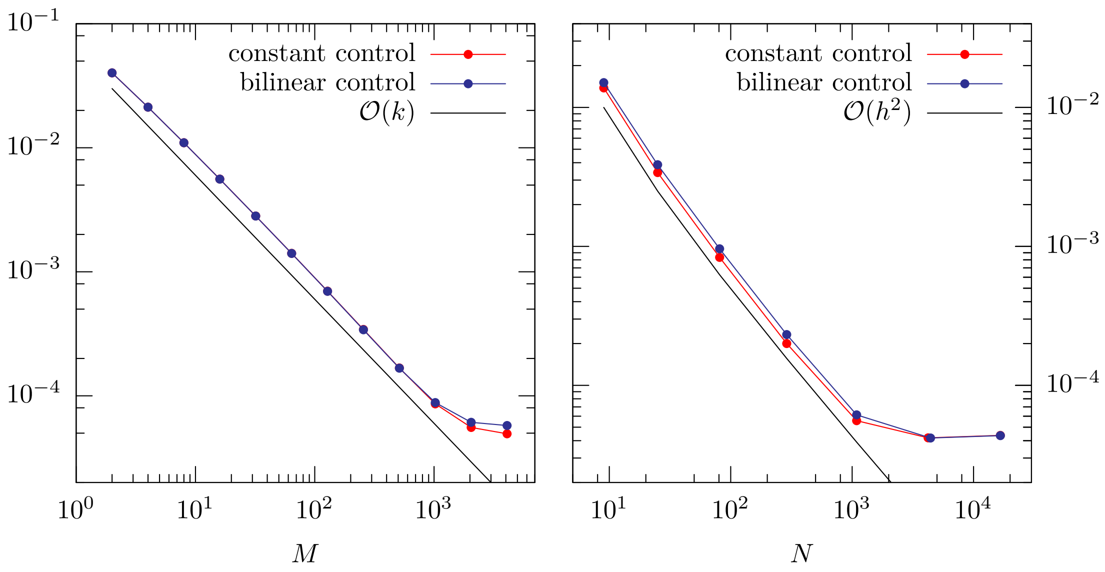
Figure 5.1. Discretization error $\|q - q_\sigma\|_I$



(a) Refinement of the time steps for $N = 1{,}089$ spatial nodes

(b) Refinement of the spatial triangulation for $M = 2{,}048$ time steps

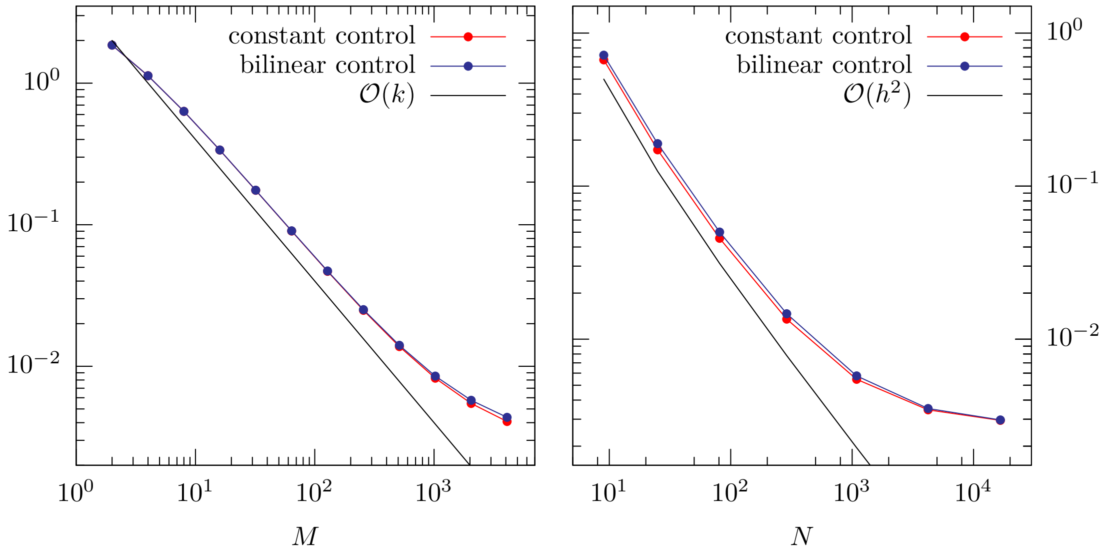
Figure 5.2. Discretization error $\|u - u_\sigma\|_I$



(a) Refinement of the time steps for $N = 1{,}089$ spatial nodes

(b) Refinement of the spatial triangulation for $M = 2{,}048$ time steps

Figure 5.3. Discretization error $\|z - z_\sigma\|_I$



(a) Refinement of the time steps for $N = 1{,}089$ spatial nodes

(b) Refinement of the spatial triangulation for $M = 2{,}048$ time steps

Figure 5.4. Discretization error $|J(q,u) - J(q_\sigma, u_\sigma)|$

Figure 5.1(a) depicts the development of the error $\|q - q_\sigma\|_I$ under refinement of the temporal step size k. Up to the spatial discretization error, it exhibits the proven convergence order $\mathcal{O}(k)$ for both kinds of spatial discretizations of the control space. For piecewise constant control, the discretization error is already reached at 128 time steps, whereas in the case of bilinear control, the number of time steps could be increased up to $M = 4{,}096$ before reaching the spatial accuracy. In Figure 5.1(b), the development of the error in the control variable under spatial refinement is shown. The expected order $\mathcal{O}(h)$ for piecewise constant control (dG(0)cG(0) discretization) and $\mathcal{O}(h^2)$ for bilinear control (cG(1)dG(0) discretization) is observed. Hence, these results confirm the estimate for the error $\|q - q_\sigma\|_I$ from Theorem 5.18.

The Figures 5.2 and 5.3 show the errors $\|u - u_\sigma\|_I$ and $\|z - z_\sigma\|_I$ in terms the state variable u and the adjoint variable z for separate refinement of the time and space discretizations. Thereby, we observe for all errors convergence of order $\mathcal{O}(k + h^2)$ as proven in the previous section, regardless the type of spatial discretization used for the controls. This substantiates the assertions of Theorem 5.24 and Corollary 5.25.

Finally, the estimate from Theorem 5.26 concerning the error $|J(q, u) - J(q_\sigma, u_\sigma)|$ is confirmed by Figure 5.4. The error exhibits the proposed convergence of order $\mathcal{O}(k + h^2)$.

6 A Posteriori Error Estimation and Adaptivity

The main goal of this chapter is to derive a posteriori error estimates which assess the error between the solution (q, u) of the continuous and the solution (q_σ, u_σ) of the discrete optimization problem with respect to a given quantity of interest. This quantity of interest (denoted by E) may coincide with the cost functional J or expresses a different goal for the computation. In order to set up an efficient adaptive algorithm, we separate the influences of the different discretizations (time and space discretizations of the state discretization of the control) on the total discretization error measured in terms of the quantity of interest. This quantitative error estimation allows to balance the different types of errors during an equilibration procedure and to successively improve the accuracy by the construction of locally refined discretizations.

The use of adaptive techniques based on a posteriori error estimation is well accepted in the context of finite element discretization of partial differential equations; see for instance Eriksson, Estep, Hansbo, and Johnson [28], Verfürth [81], or Becker and Rannacher [9, 10]. In the last years, the application of these techniques has also been investigated for optimization problems governed by partial differential equations. Energy-type error estimators for the error in the control, state, and adjoint state variable are developed in Liu and Yan [55, 56] in the context of distributed elliptic optimal control problems subject to pointwise control constraints. Recently, these techniques are also applied in the context of optimal control problems governed by linear parabolic equations; see Liu, Ma, Tang, and Yan [54]. In Picasso [66], an anisotropic error estimate is derived for the error due to the space discretization of an optimal control problem governed by the linear heat equation.

However, in many applications, the error in global norms does not provide useful error bounds for the error in the quantity of physical interest. In Becker and Kapp [6], Becker, Kapp, and Rannacher [7], and Becker and Rannacher [10], a general concept for a posteriori estimation of the discretization error with respect to the cost functional in the context of optimal control problems is presented. In Becker and Vexler [11, 12], this approach is extended to the estimation of the discretization error with respect to an arbitrary functional depending on both the control and the state variable, that is with respect to a given quantity of interest. This allows—amongst others—an efficient treatment of parameter identification and model calibration problems.

In this chapter, these approaches are extended to optimization problems governed by parabolic partial differential equations: At First, we derive an a posteriori error estimate with respect to the cost functional J:

$$J(q, u) - J(q_\sigma, u_\sigma) \approx \eta_k^J + \eta_h^J + \eta_d^J.$$

Thereby, the estimators η_k^J, η_h^J, and η_d^J assess the errors caused by the discretization of the state variable in time and space (cf. the Sections 3.1 and 3.2) and by the discretization of the control variable (cf. Section 3.3). This splitting allows for balancing the different error contributions within an adaptive refinement algorithm; see Section 6.5.

Since in many applications the quantity of physical interest does not coincide with the cost functional, we also investigate error estimations which assess the error in terms of a given quantity of interest E:

$$E(q,u) - E(q_\sigma, u_\sigma) \approx \eta_k^E + \eta_h^E + \eta_d^E.$$

Again, η_k^E, η_h^E, and η_d^E estimate the error contributions due to the discretization of the state variable in space and time and due to the control discretization.

This chapter is organized as follows: After recalling an abstract error identity in Section 6.1, we derive an a posteriori error estimate in terms of the cost functional in Section 6.2. Thereby, we present the detailed derivation of the error estimator for the fully discrete optimization problem in the case of discontinuous Galerkin time discretization. For the continuous Galerkin time discretization, we only present the results since the derivation can be done similarly to the discontinuous case. This applies also for Section 6.3, where we extend the presented techniques to obtain estimates in terms of a given quantity of interest. The Sections 6.4 and 6.5 are devoted to the practical aspects of error estimation and adaptive refinement of the underlying discretizations. In particular, we cover thereby the topics of approximating the weights arising in the error estimator and localizing the error indicators as well as equilibration strategies to balance the different types of discretization errors. For comparison purposes, we introduce in Section 6.6 a heuristic error estimator based on smoothness properties of the optimal state and adjoint state. In Section 6.7, we present two numerical examples elucidating the theoretical results developed in this chapter. Furthermore, we compare the derived techniques with the performance of the heuristic error estimator introduced in Section 6.6.

Major parts of the results presented here are already published in Meidner and Vexler [59]. Additionally, we present in this thesis results obtained by the usage of dynamically changing meshes, which are not included in the article. Using dynamically changing meshes means to allow different spatial meshes for different time steps; see the detailed discussion in Section 3.2.2. This offers further possibilities for saving computational costs especially when considering highly dynamical systems as done later in Chapter 7.

6.1 Abstract error estimate

As a preparation we recall a modification of an abstract result from Becker and Rannacher [10], which we utilize below to establish the desired a posteriori error estimates:

Lemma 6.1. *Let Y be a function space and L be a three times Gâteaux differentiable functional on Y. We seek a stationary point y_1 of L on a subspace $Y_1 \subseteq Y$, that is we seek y_1 fulfilling*

$$L'(y_1)(\delta y_1) = 0 \quad \forall \delta y_1 \in Y_1. \tag{6.1}$$

This equation is approximated by a Galerkin method using a subspace $Y_2 \subseteq Y$. The approximative problem seeks $y_2 \in Y_2$ satisfying

$$L'(y_2)(\delta y_2) = 0 \quad \forall \delta y_2 \in Y_2. \tag{6.2}$$

If the continuous solution y_1 fulfills additionally

$$L'(y_1)(y_2) = 0, \tag{6.3}$$

then we have for arbitrary $\hat{y}_2 \in Y_2$ the error representation

$$L(y_1) - L(y_2) = \frac{1}{2} L'(y_2)(y_1 - \hat{y}_2) + \mathcal{R}, \tag{6.4}$$

where the remainder term $\mathcal{R}$ is given by means of $e := y_1 - y_2$ as

$$\mathcal{R} = \frac{1}{2} \int_0^1 L'''(y_2 + se)(e, e, e) \cdot s \cdot (s - 1)\, ds.$$

Proof. By the main theorem of calculus, we have

$$L(y_1) - L(y_2) = \int_0^1 L'(y_2 + se)(e)\, ds.$$

Evaluation of this integral by the trapezoidal rule

$$\int_0^1 f(s)\, ds = \frac{1}{2} f(0) + \frac{1}{2} f(1) + \frac{1}{2} \int_0^1 f''(s) \cdot s \cdot (s - 1)\, ds$$

yields

$$L(y_1) - L(y_2) = \frac{1}{2} L'(y_2)(e) + \frac{1}{2} L'(y_1)(e) + \mathcal{R}.$$

Due to the assertions (6.1) and (6.3), the term $L'(y_1)(e)$ vanishes and due to (6.2), the term $L'(y_2)(e)$ can be replaced by $L'(y_2)(y_1 - \hat{y}_2)$ for any $\hat{y}_2 \in Y_2$. This completes the proof. □

Remark 6.1. Usually, Lemma 6.1 is formulated with the stronger requirement $Y_1 = Y$ instead of condition (6.3). However, the presented formulation is necessary, since we can not always assure the property $Y_2 \subseteq Y_1$ for the concrete discretizations considered in the following.

6.2 Error estimator for the cost functional

In this section, we use the abstract result of Lemma 6.1 for deriving error estimators in terms of the cost functional J assessing the error

$$J(q, u) - J(q_\sigma, u_\sigma).$$

Here, $(q, u) \in Q \times X$ denotes the continuous optimal solution of problem $(\mathbb{P})$ and $(q_\sigma, u_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{r,s}$ is the optimal solution of problem $(\widetilde{\mathbb{P}}_\sigma)$ where the state is discretized by the cG(s)dG(r) method and the control is searched in the discrete control space $Q_d \subseteq Q$.

To separate the influences of the different discretizations on the discretization error we are interested in, we split

$$J(q, u) - J(q_\sigma, u_\sigma) = J(q, u) - J(q_k, u_k) + J(q_k, u_k) - J(q_{kh}, u_{kh}) + J(q_{kh}, u_{kh}) - J(q_\sigma, u_\sigma),$$

where $(q_k, u_k) \in Q \times \widetilde{X}_k^r$ is the solution of the time-discretized problem $(\widetilde{\mathbb{P}}_k)$, and $(q_{kh}, u_{kh}) \in Q \times \widetilde{X}_{k,h}^{r,s}$ is the solution of the time- and space-discretized problem $(\widetilde{\mathbb{P}}_{kh})$ where the control space is still continuous.

The following theorem forms the basis for a posteriori estimation of the discretization error with respect to the cost functional in the context of parabolic optimization problems:

Theorem 6.2. *Let (q, u, z), (q_k, u_k, z_k), (q_{kh}, u_{kh}, z_{kh}), and $(q_\sigma, u_\sigma, z_\sigma)$ be stationary points of $\mathcal{L}$ resp. $\widetilde{\mathcal{L}}$ on the different levels of discretization, that is*

$$\begin{aligned}
\mathcal{L}'(q,u,z)(\delta q,\delta u,\delta z) = \widetilde{\mathcal{L}}'(q,u,z)(\delta q,\delta u,\delta z) &= 0 &\quad \forall(\delta q,\delta u,\delta z) &\in Q \times X \times X,\\
\widetilde{\mathcal{L}}'(q_k,u_k,z_k)(\delta q_k,\delta u_k,\delta z_k) &= 0 &\quad \forall(\delta q_k,\delta u_k,\delta z_k) &\in Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r,\\
\widetilde{\mathcal{L}}'(q_{kh},u_{kh},z_{kh})(\delta q_{kh},\delta u_{kh},\delta z_{kh}) &= 0 &\quad \forall(\delta q_{kh},\delta u_{kh},\delta z_{kh}) &\in Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s},\\
\widetilde{\mathcal{L}}'(q_\sigma,u_\sigma,z_\sigma)(\delta q_\sigma,\delta u_\sigma,\delta z_\sigma) &= 0 &\quad \forall(\delta q_\sigma,\delta u_\sigma,\delta z_\sigma) &\in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}.
\end{aligned}$$

Then, there holds for the errors with respect to the cost functional J due to dG(r)*-time,* cG(s)*-space, and control discretization:*

$$\begin{aligned}
J(q,u) - J(q_k,u_k) &= \frac{1}{2}\widetilde{\mathcal{L}}'(q_k,u_k,z_k)(q-\hat{q}_k, u-\hat{u}_k, z-\hat{z}_k) + \mathcal{R}_k^J,\\
J(q_k,u_k) - J(q_{kh},u_{kh}) &= \frac{1}{2}\widetilde{\mathcal{L}}'(q_{kh},u_{kh},z_{kh})(q_k-\hat{q}_{kh}, u_k-\hat{u}_{kh}, z_k-\hat{z}_{kh}) + \mathcal{R}_h^J,\\
J(q_{kh},u_{kh}) - J(q_\sigma,u_\sigma) &= \frac{1}{2}\widetilde{\mathcal{L}}'(q_\sigma,u_\sigma,z_\sigma)(q_{kh}-\hat{q}_\sigma, u_{kh}-\hat{u}_\sigma, z_{kh}-\hat{z}_\sigma) + \mathcal{R}_d^J.
\end{aligned}$$

Here, $(\hat{q}_k, \hat{u}_k, \hat{z}_k) \in Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r$, $(\hat{q}_{kh}, \hat{u}_{kh}, \hat{z}_{kh}) \in Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}$, and $(\hat{q}_\sigma, \hat{u}_\sigma, \hat{z}_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}$ can be chosen arbitrarily and the remainder terms $\mathcal{R}_k^J$, $\mathcal{R}_h^J$, and $\mathcal{R}_d^J$ have the same structure as given in Lemma 6.1 for $L = \widetilde{\mathcal{L}}$.

Proof. Since all the used solution pairs are optimal solutions of the optimization problem on different discretization levels, we obtain for arbitrary $z \in X$, $z_k \in \widetilde{X}_k^r$, and $z_{kh}, z_\sigma \in \widetilde{X}_{k,h}^{r,s}$

$$J(q,u) - J(q_k,u_k) = \widetilde{\mathcal{L}}(q,u,z) - \widetilde{\mathcal{L}}(q_k,u_k,z_k), \tag{6.5a}$$

$$J(q_k,u_k) - J(q_{kh},u_{kh}) = \widetilde{\mathcal{L}}(q_k,u_k,z_k) - \widetilde{\mathcal{L}}(q_{kh},u_{kh},z_{kh}), \tag{6.5b}$$

$$J(q_{kh},u_{kh}) - J(q_\sigma,u_\sigma) = \widetilde{\mathcal{L}}(q_{kh},u_{kh},z_{kh}) - \widetilde{\mathcal{L}}(q_\sigma,u_\sigma,z_\sigma), \tag{6.5c}$$

whereas the identity

$$J(q,u) = \mathcal{L}(q,u,z) = \widetilde{\mathcal{L}}(q,u,z)$$

follows from the fact that $u \in X$ is continuous and thus the additional jump terms in $\widetilde{\mathcal{L}}$ compared to $\mathcal{L}$ vanish.

To apply the abstract error identity (6.4) to the three right-hand sides in (6.5), we choose the spaces Y_1 and Y_2 in Lemma 6.1 as

$$\begin{aligned}
Y_1 &= Q \times X \times X, & Y_2 &= Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r & &\text{for (6.5a)},\\
Y_1 &= Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r, & Y_2 &= Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s} & &\text{for (6.5b)},\\
Y_1 &= Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}, & Y_2 &= Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s} & &\text{for (6.5c)}.
\end{aligned}$$

Hence, for the second and third pairing we have $Y_2 \subseteq Y_1$ since we have $\widetilde{X}_{k,h}^{r,s} \subseteq \widetilde{X}_k^r$ and $Q_d \subseteq Q$. Thus, we can choose $Y = Y_1$ in these cases which implies directly condition (6.3).

For the choice of the spaces for (6.5a), we have to take into account the fact that $\widetilde{X}_k^r \not\subseteq X$. Thus, we choose $Y = Y_1 + Y_2$ and have to ensure condition (6.3), which reads here as

$$\widetilde{\mathcal{L}}'(q,u,z)(q_k,u_k,z_k) = 0$$

with $(q_k,u_k,z_k) \in Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r$ the optimal solution of $(\widetilde{\mathbb{P}}_k)$. This can be expressed equivalently in terms of assertions for the three derivatives of $\widetilde{\mathcal{L}}$:

$$\widetilde{\mathcal{L}}'_q(q,u,z)(q_k) = 0, \qquad \widetilde{\mathcal{L}}'_u(q,u,z)(u_k) = 0, \qquad \widetilde{\mathcal{L}}'_z(q,u,z)(z_k) = 0.$$

We only demonstrate the details of proving the condition

$$\widetilde{\mathcal{L}}'_u(q,u,z)(u_k) = 0. \tag{6.6}$$

The other two conditions can be treated similarly. Due to the continuity of z with respect to time, (6.6) can be rewritten after integration by parts in time as

$$-(u_k, \partial_t z)_I + a'_u(q,u)(u_k,z) + (u_{k,M}^-, z(T)) = \int_I J'_1(u)(u_k)\,dt + J'_2(u(T))(u_{k,M}^-).$$

The adjoint equation (2.11) implies for the continuous adjoint solution z the equality

$$(\varphi, z(T)) = J'_2(u(T))(\varphi) \quad \forall \varphi \in H.$$

Consequently, the terms containing $u_{k,M}^-$ cancel out and it remains to ensure

$$-(u_k, \partial_t z)_I + a'_u(q,u)(u_k,z) = \int_I J'_1(u)(u_k)\,dt.$$

Again from the continuous adjoint equation (2.11), we have that z fulfills

$$-(\varphi, \partial_t z)_I + a'_u(q,u)(\varphi,z) = \int_I J'_1(u)(\varphi)\,dt \quad \forall \varphi \in X.$$

Since X is dense in $L^2(I,V)$ in regard to the $L^2(I,V)$-norm and since there are no time derivatives on the test function φ in this formulation, it also holds true for all test functions $\varphi \in L^2(I,V)$. Then, the inclusion $u_k \in \widetilde{X}_k^r \subseteq L^2(I,V)$ implies that condition (6.6) is fulfilled.

Finally, the assertion of the theorem follows immediately by application of Lemma 6.1 to the three separated errors (6.5). □

For the cG(r) time discretization, Theorem 6.2 reads as:

Corollary 6.3. *Let (q,u,z), (q_k,u_k,z_k), (q_{kh},u_{kh},z_{kh}), and $(q_\sigma,u_\sigma,z_\sigma)$ be stationary points of $\mathcal{L}$ on the different levels of discretization, that is*

$$\begin{aligned}
\mathcal{L}'(q,u,z)(\delta q,\delta u,\delta z) &= 0 && \forall(\delta q,\delta u,\delta z) \in Q \times X \times X,\\
\mathcal{L}'(q_k,u_k,z_k)(\delta q_k,\delta u_k,\delta z_k) &= 0 && \forall(\delta q_k,\delta u_k,\delta z_k) \in Q \times X_k^r \times \widetilde{X}_k^r,\\
\mathcal{L}'(q_{kh},u_{kh},z_{kh})(\delta q_{kh},\delta u_{kh},\delta z_{kh}) &= 0 && \forall(\delta q_{kh},\delta u_{kh},\delta z_{kh}) \in Q \times X_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s},\\
\mathcal{L}'(q_\sigma,u_\sigma,z_\sigma)(\delta q_\sigma,\delta u_\sigma,\delta z_\sigma) &= 0 && \forall(\delta q_\sigma,\delta u_\sigma,\delta z_\sigma) \in Q_d \times X_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}.
\end{aligned}$$

Then, there holds for the errors with respect to the cost functional J due to cG(r)*-time,* cG(s)*-space, and control discretization:*

$$\begin{aligned}
J(q,u)-J(q_k,u_k) &= \frac{1}{2}\mathcal{L}'(q_k,u_k,z_k)(q-\hat{q}_k,u-\hat{u}_k,z-\hat{z}_k)+\mathcal{R}^J_k,\\
J(q_k,u_k)-J(q_{kh},u_{kh}) &= \frac{1}{2}\mathcal{L}'(q_{kh},u_{kh},z_{kh})(q_k-\hat{q}_{kh},u_k-\hat{u}_{kh},z_k-\hat{z}_{kh})+\mathcal{R}^J_h,\\
J(q_{kh},u_{kh})-J(q_\sigma,u_\sigma) &= \frac{1}{2}\mathcal{L}'(q_\sigma,u_\sigma,z_\sigma)(q_{kh}-\hat{q}_\sigma,u_{kh}-\hat{u}_\sigma,z_{kh}-\hat{z}_\sigma)+\mathcal{R}^J_d.
\end{aligned}$$

Here, $(\hat{q}_k,\hat{u}_k,\hat{z}_k)\in Q\times X^r_k\times\widetilde{X}^r_k$, $(\hat{q}_{kh},\hat{u}_{kh},\hat{z}_{kh})\in Q\times X^{r,s}_{k,h}\times\widetilde{X}^{r,s}_{k,h}$, and $(\hat{q}_\sigma,\hat{u}_\sigma,\hat{z}_\sigma)\in Q_d\times X^{r,s}_{k,h}\times\widetilde{X}^{r,s}_{k,h}$ can be chosen arbitrarily and the remainder terms $\mathcal{R}^J_k$, $\mathcal{R}^J_h$, and $\mathcal{R}^J_d$ have the same structure as given in Lemma 6.1 for $L=\mathcal{L}$.

By means of the residuals of the three equations building the optimality system in term of the semidiscrete Lagrangian (cf. (2.12) for the continuous Lagrangian)

$$\begin{aligned}
\tilde{\rho}^u(q,u)(\varphi) &:= \widetilde{\mathcal{L}}'_z(q,u,z)(\varphi),\\
\tilde{\rho}^z(q,u,z)(\varphi) &:= \widetilde{\mathcal{L}}'_u(q,u,z)(\varphi),\\
\tilde{\rho}^q(q,u,z)(\varphi) &:= \widetilde{\mathcal{L}}'_q(q,u,z)(\varphi),
\end{aligned}$$

the statement of Theorem 6.2 can be rewritten as

$$\begin{aligned}
J(q,u)-J(q_k,u_k) &\approx \frac{1}{2}\Big\{\tilde{\rho}^u(q_k,u_k)(z-\hat{z}_k)+\tilde{\rho}^z(q_k,u_k,z_k)(u-\hat{u}_k)\Big\},\\
J(q_k,u_k)-J(q_{kh},u_{kh}) &\approx \frac{1}{2}\Big\{\tilde{\rho}^u(q_{kh},u_{kh})(z_k-\hat{z}_{kh})+\tilde{\rho}^z(q_{kh},u_{kh},z_{kh})(u_k-\hat{u}_{kh})\Big\},\\
J(q_{kh},u_{kh})-J(q_\sigma,u_\sigma) &\approx \frac{1}{2}\tilde{\rho}^q(q_\sigma,u_\sigma,z_\sigma)(q_{kh}-\hat{q}_\sigma).
\end{aligned}$$

Here, we employed the fact, that the terms

$$\begin{aligned}
&\tilde{\rho}^q(q_k,u_k,z_k)(q-\hat{q}_k), && \tilde{\rho}^q(q_{kh},u_{kh},z_{kh})(q_k-\hat{q}_{kh}),\\
&\tilde{\rho}^u(q_\sigma,u_\sigma)(z_{kh}-\hat{z}_\sigma), && \tilde{\rho}^z(q_\sigma,u_\sigma,z_\sigma)(u_{kh}-\hat{u}_\sigma)
\end{aligned}$$

are zero for the feasible choices

$$\begin{aligned}
&\hat{q}_k=q\in Q, && \hat{q}_{kh}=q_k\in Q,\\
&\hat{z}_\sigma=z_{kh}\in\widetilde{X}^{r,s}_{k,h}, && \hat{u}_\sigma=u_{kh}\in\widetilde{X}^{r,s}_{k,h}.
\end{aligned}$$

This is possible since for the errors $J(q,u)-J(q_k,u_k)$ and $J(q_k,u_k)-J(q_{kh},u_{kh})$ only the state space is discretized, and for $J(q_{kh},u_{kh})-J(q_\sigma,u_\sigma)$ we keep the discrete state space while discretizing the control space Q.

6.3 Error estimator for an arbitrary functional

We now tend towards an estimation of the different types of discretization errors in terms of a given functional $E\colon Q\times X\to\mathbb{R}$ describing a quantity of interest. This is done by utilizing solutions to some auxiliary problems. In order to ensure the solvability of these problems we make the following assumptions:

- The semidiscrete and the fully discrete optimal solutions (q_k, u_k), (q_{kh}, u_{kh}), and (q_σ, u_σ) are in the neighborhood $W \subseteq Q \times X$ of the optimal solution (q, u) introduced at the end of Section 2.3.
- These (semi-)discrete solutions as well as the continuous solution (q, u) fulfill a second order sufficient optimality condition as stated in Theorem 2.5.

Remark 6.2. In most publications concerning the topic of estimating the discretization error in terms of a quantity of interest, this quantity was denoted by I. To avoid confusion concerning the time interval which is here also called I, we denote the functional by E like it was initially denoted in Vexler [82].

For formulating the error estimate, we define exterior Lagrangians $\mathcal{M}\colon [Q \times X \times X]^2 \to \mathbb{R}$ and $\widetilde{\mathcal{M}}\colon [Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r]^2 \to \mathbb{R}$ as

$$\mathcal{M}(\xi, \chi) := E(q, u) + \mathcal{L}'(\xi)(\chi) \quad \text{and} \quad \widetilde{\mathcal{M}}(\xi_k, \chi_k) := E(q_k, u_k) + \widetilde{\mathcal{L}}'(\xi_k)(\chi_k)$$

with $\xi := (q, u, z)$, $\chi := (p, v, y)$ and $\xi_k := (q_k, u_k, z_k)$, $\chi_k := (p_k, v_k, y_k)$. In this connection, the variables χ and χ_k can be interpreted as dual variables for optimization problem $(\mathbb{P})$.

Now we are in a similar setting as in the section before: We split the total discretization error with respect to E as

$$E(q, u) - E(q_\sigma, u_\sigma) = E(q, u) - E(q_k, u_k) + E(q_k, u_k) - E(q_{kh}, u_{kh}) + E(q_{kh}, u_{kh}) - E(q_\sigma, u_\sigma)$$

and obtain the following theorem:

Theorem 6.4. *Let (ξ, χ), (ξ_k, χ_k), (ξ_{kh}, χ_{kh}), and $(\xi_\sigma, \chi_\sigma)$ be stationary points of $\mathcal{M}$ resp. $\widetilde{\mathcal{M}}$ on the different levels of discretization, that is*

$$\begin{aligned}
\mathcal{M}'(\xi, \chi)(\delta\xi, \delta\chi) = \widetilde{\mathcal{M}}'(\xi, \chi)(\delta\xi, \delta\chi) &= 0 \qquad &&\forall (\delta\xi, \delta\chi) \in [Q \times X \times X]^2,\\
\widetilde{\mathcal{M}}'(\xi_k, \chi_k)(\delta\xi_k, \delta\chi_k) &= 0 &&\forall (\delta\xi_k, \delta\chi_k) \in [Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r]^2,\\
\widetilde{\mathcal{M}}'(\xi_{kh}, \chi_{kh})(\delta\xi_{kh}, \delta\chi_{kh}) &= 0 &&\forall (\delta\xi_{kh}, \delta\chi_{kh}) \in [Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2,\\
\widetilde{\mathcal{M}}'(\xi_\sigma, \chi_\sigma)(\delta\xi_\sigma, \delta\chi_\sigma) &= 0 &&\forall (\delta\xi_\sigma, \delta\chi_\sigma) \in [Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2.
\end{aligned}$$

Then, there holds for the errors with respect to the quantity of interest E due to dG(r)*-time,* cG(s)*-space, and control discretization:*

$$\begin{aligned}
E(q, u) - E(q_k, u_k) &= \frac{1}{2}\widetilde{\mathcal{M}}'(\xi_k, \chi_k)(\xi - \hat{\xi}_k, \chi - \hat{\chi}_k) + \mathcal{R}_k^E,\\
E(q_k, u_k) - E(q_{kh}, u_{kh}) &= \frac{1}{2}\widetilde{\mathcal{M}}'(\xi_{kh}, \chi_{kh})(\xi_k - \hat{\xi}_{kh}, \chi_k - \hat{\chi}_{kh}) + \mathcal{R}_h^E,\\
E(q_{kh}, u_{kh}) - E(q_\sigma, u_\sigma) &= \frac{1}{2}\widetilde{\mathcal{M}}'(\xi_\sigma, \chi_\sigma)(\xi_{kh} - \hat{\xi}_\sigma, \chi_{kh} - \hat{\chi}_\sigma) + \mathcal{R}_d^E.
\end{aligned}$$

Here, $(\hat{\xi}_k, \hat{\chi}_k) \in [Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r]^2$, $(\hat{\xi}_{kh}, \hat{\chi}_{kh}) \in [Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2$, and $(\hat{\xi}_\sigma, \hat{\chi}_\sigma) \in [Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2$ can be chosen arbitrarily and the remainder terms $\mathcal{R}_k^E$, $\mathcal{R}_h^E$, and $\mathcal{R}_d^E$ have the same structure as given in Lemma 6.1 for $L = \widetilde{\mathcal{M}}$.

Proof. Since $\widetilde{\mathcal{M}}' = 0$ implies $\widetilde{\mathcal{L}}' = 0$, we have on the different levels of discretization the representations

$$E(q,u) - E(q_k,u_k) = \widetilde{\mathcal{M}}(\xi,\chi) - \widetilde{\mathcal{M}}(\xi_k,\chi_k), \tag{6.7a}$$

$$E(q_k,u_k) - E(q_{kh},u_{kh}) = \widetilde{\mathcal{M}}(\xi_k,\chi_k) - \widetilde{\mathcal{M}}(\xi_{kh},\chi_{kh}), \tag{6.7b}$$

$$E(q_{kh},u_{kh}) - E(q_\sigma,u_\sigma) = \widetilde{\mathcal{M}}(\xi_{kh},\chi_{kh}) - \widetilde{\mathcal{M}}(\xi_\sigma,\chi_\sigma), \tag{6.7c}$$

where the identity

$$E(q,u) = \mathcal{M}(\xi,\chi) = \widetilde{\mathcal{M}}(\xi,\chi)$$

follows again from the fact that $u \in X$ and $z \in X$ are continuous and thus the additional jump terms in $\widetilde{\mathcal{M}}$ compared to $\mathcal{M}$ vanish.

We choose the spaces Y_1 and Y_2 for application of Lemma 6.1 as

$$\begin{aligned}
&Y_1 = [Q \times X \times X]^2, && Y_2 = [Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r]^2 && \text{for (6.7a)},\\
&Y_1 = [Q \times \widetilde{X}_k^r \times \widetilde{X}_k^r]^2, && Y_2 = [Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2 && \text{for (6.7b)},\\
&Y_1 = [Q \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2, && Y_2 = [Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2 && \text{for (6.7c)},
\end{aligned}$$

and end up with the stated error representations after ensuring the prerequisites of Lemma 6.1 as done in the proof of Theorem 6.2. □

In the case of cG(r) time discretization, Theorem 6.4 reads as:

Corollary 6.5. *Let (ξ,χ), (ξ_k,χ_k), (ξ_{kh},χ_{kh}), and (ξ_σ,χ_σ) be stationary points of $\mathcal{M}$ on the different levels of discretization, that is*

$$\begin{aligned}
\mathcal{M}'(\xi,\chi)(\delta\xi,\delta\chi) &= 0 & \forall(\delta\xi,\delta\chi) &\in [Q \times X \times X]^2,\\
\mathcal{M}'(\xi_k,\chi_k)(\delta\xi_k,\delta\chi_k) &= 0 & \forall(\delta\xi_k,\delta\chi_k) &\in [Q \times X_k^r \times \widetilde{X}_k^r]^2,\\
\mathcal{M}'(\xi_{kh},\chi_{kh})(\delta\xi_{kh},\delta\chi_{kh}) &= 0 & \forall(\delta\xi_{kh},\delta\chi_{kh}) &\in [Q \times X_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2,\\
\mathcal{M}'(\xi_\sigma,\chi_\sigma)(\delta\xi_\sigma,\delta\chi_\sigma) &= 0 & \forall(\delta\xi_\sigma,\delta\chi_\sigma) &\in [Q_d \times X_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2.
\end{aligned}$$

Then, there holds for the errors with respect to the quantity of interest E due to the cG(r)*-time,* cG(s)*-space, and control discretization:*

$$\begin{aligned}
E(q,u) - E(q_k,u_k) &= \frac{1}{2}\mathcal{M}'(\xi_k,\chi_k)(\xi - \hat{\xi}_k, \chi - \hat{\chi}_k) + \mathcal{R}_k^E,\\
E(q_k,u_k) - E(q_{kh},u_{kh}) &= \frac{1}{2}\mathcal{M}'(\xi_{kh},\chi_{kh})(\xi_k - \hat{\xi}_{kh}, \chi_k - \hat{\chi}_{kh}) + \mathcal{R}_h^E,\\
E(q_{kh},u_{kh}) - E(q_\sigma,u_\sigma) &= \frac{1}{2}\mathcal{M}'(\xi_\sigma,\chi_\sigma)(\xi_{kh} - \hat{\xi}_\sigma, \chi_{kh} - \hat{\chi}_\sigma) + \mathcal{R}_d^E.
\end{aligned}$$

Here, $(\hat{\xi}_k,\hat{\chi}_k) \in [Q \times X_k^r \times \widetilde{X}_k^r]^2$, $(\hat{\xi}_{kh},\hat{\chi}_{kh}) \in [Q \times X_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2$, and $(\hat{\xi}_\sigma,\hat{\chi}_\sigma) \in [Q_d \times X_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2$ can be chosen arbitrarily and the remainder terms $\mathcal{R}_k^E$, $\mathcal{R}_h^E$, and $\mathcal{R}_d^E$ have the same structure as given in Lemma 6.1 for $L = \mathcal{M}$.

To apply Theorem 6.4 for instance to $E(q_{kh}, u_{kh}) - E(q_\sigma, u_\sigma)$, we have to require that

$$\widetilde{\mathcal{M}}'(\xi_\sigma, \chi_\sigma)(\delta\xi_\sigma, \delta\chi_\sigma) = 0 \quad \forall(\delta\xi_\sigma, \delta\chi_\sigma) \in [Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}]^2,$$

or equivalently

$$\begin{aligned} \widetilde{\mathcal{M}}'_\xi(\xi_\sigma, \chi_\sigma)(\delta\xi_\sigma) &= 0 \qquad \forall\delta\xi_\sigma \in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s} \qquad \text{and} \\ \widetilde{\mathcal{M}}'_\chi(\xi_\sigma, \chi_\sigma)(\delta\chi_\sigma) &= 0 \qquad \forall\delta\chi_\sigma \in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}. \end{aligned}$$

Since $\xi_\sigma = (q_\sigma, u_\sigma, z_\sigma)$ is already determined by the condition

$$\widetilde{\mathcal{M}}'_\chi(\xi_\sigma, \chi_\sigma)(\delta\chi_\sigma) = \widetilde{\mathcal{L}}'(\xi_\sigma)(\delta\chi_\sigma) = 0 \quad \forall\delta\chi_\sigma \in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s},$$

the triple ξ_σ is the solution of the fully discrete optimization problem. Thus, the solution triple $\chi_\sigma = (p_\sigma, v_\sigma, y_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}$ is given as solution of

$$\widetilde{\mathcal{M}}'_\xi(\xi_\sigma, \chi_\sigma)(\delta\xi_\sigma) = 0 \quad \forall\delta\xi_\sigma \in Q_d \times \widetilde{X}_{k,h}^{r,s} \times \widetilde{X}_{k,h}^{r,s}. \tag{6.8}$$

Thereby, the derivative $\widetilde{\mathcal{M}}'_\xi(\xi_\sigma, \chi_\sigma)(\delta\xi_\sigma)$ is the sum of the partial derivatives

$$\begin{aligned} \widetilde{\mathcal{M}}'_q(\xi_\sigma, \chi_\sigma)(\delta q_\sigma) &= E'_q(q_\sigma, u_\sigma)(\delta q_\sigma) + \widetilde{\mathcal{L}}''_{qq}(\xi_\sigma)(p_\sigma, \delta q_\sigma) + \widetilde{\mathcal{L}}''_{uq}(\xi_\sigma)(v_\sigma, \delta q_\sigma) + \widetilde{\mathcal{L}}''_{zq}(\xi_\sigma)(y_\sigma, \delta q_\sigma), \\ \widetilde{\mathcal{M}}'_u(\xi_\sigma, \chi_\sigma)(\delta u_\sigma) &= E'_u(q_\sigma, u_\sigma)(\delta u_\sigma) + \widetilde{\mathcal{L}}''_{qu}(\xi_\sigma)(p_\sigma, \delta u_\sigma) + \widetilde{\mathcal{L}}''_{uu}(\xi_\sigma)(v_\sigma, \delta u_\sigma) + \widetilde{\mathcal{L}}''_{zu}(\xi_\sigma)(y_\sigma, \delta u_\sigma), \\ \widetilde{\mathcal{M}}'_z(\xi_\sigma, \chi_\sigma)(\delta z_\sigma) &= \widetilde{\mathcal{L}}''_{qz}(\xi_\sigma)(p_\sigma, \delta z_\sigma) + \widetilde{\mathcal{L}}''_{uz}(\xi_\sigma)(v_\sigma, \delta z_\sigma). \end{aligned}$$

For ensuring the existence of a triple $\chi_\sigma = (p_\sigma, v_\sigma, y_\sigma)$ solving the system (6.8), we observe by minor transformations, that (6.8) is the optimality system of the following linear-quadratic optimization problem:

$$\begin{aligned} \text{Minimize } G(\xi_\sigma; p_\sigma, v_\sigma) &:= E'_q(q_\sigma, u_\sigma)(p_\sigma) + E'_u(q_\sigma, u_\sigma)(v_\sigma) \\ &\quad + \frac{1}{2}\widetilde{\mathcal{L}}''_{qq}(\xi_\sigma)(p_\sigma, p_\sigma) + \widetilde{\mathcal{L}}''_{qu}(\xi_\sigma)(p_\sigma, v_\sigma) + \frac{1}{2}\widetilde{\mathcal{L}}''_{uu}(\xi_\sigma)(v_\sigma, v_\sigma) \end{aligned} \tag{6.9a}$$

such that $(p_\sigma, v_\sigma) \in Q_d \times \widetilde{X}_{k,h}^{r,s}$ fulfills

$$\widetilde{\mathcal{L}}''_{qz}(\xi_\sigma)(p_\sigma, \varphi) + \widetilde{\mathcal{L}}''_{uz}(\xi_\sigma)(v_\sigma, \varphi) = 0 \quad \forall\varphi \in \widetilde{X}_{k,h}^{r,s}. \tag{6.9b}$$

Hence, the existence of a triple χ_σ solving (6.8) is equivalent to the solvability of the auxiliary optimization problem (6.9).

Since (6.9b) is identical to the fully discretized tangent equation (cf. the Chapters 2 and 3), v_σ can be expressed in terms of the discrete solution operator $S_{kh} \colon Q \to \widetilde{X}_{k,h}^{r,s}$ as $v_\sigma = S'_{kh}(q_\sigma)(p_\sigma)$. For the reduced cost functional $g_{kh}(\xi_\sigma; \cdot) \colon Q \to \mathbb{R}$ associated to (6.9) given by

$$g_{kh}(\xi_\sigma; p_\sigma) := G(\xi_\sigma; p_\sigma, S'_{kh}(q_\sigma)(p_\sigma)),$$

we obtain with the abbreviation $\delta v_\sigma := S'_{kh}(q_\sigma)(\delta p_\sigma)$ the following relation:

$$\begin{aligned} g''_{kh}(\xi_\sigma; p_\sigma)(\delta p_\sigma, \delta p_\sigma) &= \widetilde{\mathcal{L}}''_{qq}(\xi_\sigma)(\delta p_\sigma, \delta p_\sigma) + 2\widetilde{\mathcal{L}}''_{qu}(\xi_\sigma)(\delta p_\sigma, \delta v_\sigma) + \widetilde{\mathcal{L}}''_{uu}(\xi_\sigma)(\delta v_\sigma, \delta v_\sigma) \\ &= j''_{kh}(q_\sigma)(\delta p_\sigma, \delta p_\sigma). \end{aligned}$$

Assuming the second order sufficient condition for problem $(\widetilde{\mathbb{P}}_\sigma)$ at the solution $q_\sigma \in Q_d$, that is

$$j''_{kh}(q_\sigma)(\delta q_\sigma, \delta q_\sigma) \geq \gamma \|\delta q_\sigma\|_Q^2 \quad \forall \delta q_\sigma \in Q_d,$$

implies the convexity and coercivity of g_{kh}. Consequently, we obtain by Theorem 2.1 the solvability of problem (6.9). Hence, under the second order sufficient optimality condition for the discrete optimization problem $(\widetilde{\mathbb{P}}_\sigma)$, the existence of a solution χ_σ to the auxiliary problem (6.8) is ensured. For ensuring the solvability of (6.8) on the other levels of discretization, we proceed similarly.

To state an efficient way for solving system (6.8) numerically, we split $y_\sigma = y_\sigma^{(0)} + y_\sigma^{(1)}$, where $y_\sigma^{(0)} \in \widetilde{X}_{k,h}^{r,s}$ is given as the solution of

$$E'_u(q_\sigma, u_\sigma)(\varphi) + \widetilde{\mathcal{L}}''_{zu}(\xi_\sigma)(y_\sigma^{(0)}, \varphi) = 0 \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s}.$$

Consequently, $y_\sigma^{(1)} \in \widetilde{X}_{k,h}^{r,s}$ is determined by

$$\widetilde{\mathcal{L}}''_{qu}(\xi_\sigma)(p_\sigma, \varphi) + \widetilde{\mathcal{L}}''_{uu}(\xi_\sigma)(v_\sigma, \varphi) + \widetilde{\mathcal{L}}''_{zu}(\xi_\sigma)(y_\sigma^{(1)}, \varphi) = 0 \quad \forall \varphi \in \widetilde{X}_{k,h}^{r,s}.$$

This is the already introduced discrete additional adjoint equation, but now formulated in terms of the variables $(p_\sigma, v_\sigma, y_\sigma^{(1)})$ instead of $(\delta q_\sigma, \delta u_\sigma, \delta z_\sigma)$; cf. the Chapters 2 and 3. Hence, the second derivative $j''_{kh}(q_\sigma)(p_\sigma, \cdot)$ of the reduced cost functional j_{kh} can be expressed as

$$j''_{kh}(q_\sigma)(p_\sigma, \delta q_\sigma) = \widetilde{\mathcal{L}}''_{qq}(\xi_\sigma)(p_\sigma, \delta q_\sigma) + \widetilde{\mathcal{L}}''_{uq}(\xi_\sigma)(v_\sigma, \delta q_\sigma) + \widetilde{\mathcal{L}}''_{zq}(\xi_\sigma)(y_\sigma^{(1)}, \delta q_\sigma).$$

Using this representation, system (6.8) can be rewritten as

$$j''_{kh}(q_\sigma)(p_\sigma, \delta q_\sigma) = -E'_q(q_\sigma, u_\sigma)(\delta q_\sigma) - \widetilde{\mathcal{L}}''_{zq}(\xi_\sigma)(y_\sigma^{(0)}, \delta q_\sigma) \quad \forall \delta q_\sigma \in Q_d,$$

with v_σ, $y_\sigma^{(0)}$, and $y_\sigma^{(1)}$ defined as above. Consequently, solving system (6.8) is—apart from a different right-hand side—equivalent to the execution of one step of Newton's method for the reduced cost functional j.

By means of the residuals of the presented equations for v, y, and p, that is

$$\begin{aligned}
\tilde{\rho}^v(\xi, p, v)(\varphi) &:= \widetilde{\mathcal{L}}''_{uz}(\xi)(v, \varphi) + \widetilde{\mathcal{L}}''_{qz}(\xi)(p, \varphi),\\
\tilde{\rho}^y(\xi, p, v, y)(\varphi) &:= \widetilde{\mathcal{L}}''_{zu}(\xi)(y, \varphi) + \widetilde{\mathcal{L}}''_{qu}(\xi)(p, \varphi) + \widetilde{\mathcal{L}}''_{uu}(\xi)(v, \varphi) + E'_u(q, u)(\varphi),\\
\tilde{\rho}^p(\xi, p, v, y)(\varphi) &:= \widetilde{\mathcal{L}}''_{qq}(\xi)(p, \varphi) + \widetilde{\mathcal{L}}''_{uq}(\xi)(v, \varphi) + \widetilde{\mathcal{L}}''_{zq}(\xi)(y, \varphi) + E'_q(q, u)(\varphi),
\end{aligned}$$

and the already defined residuals $\tilde{\rho}^u$, $\tilde{\rho}^z$, and $\tilde{\rho}^q$, the result of Theorem 6.4 can be expressed as

$$\begin{aligned}
E(q, u) - E(q_k, u_k) &\approx \frac{1}{2}\Big\{\tilde{\rho}^u(q_k, u_k)(y - \hat{y}_k) + \tilde{\rho}^z(q_k, u_k, z_k)(v - \hat{v}_k)\\
&\qquad + \tilde{\rho}^v(\xi_k, p_k, v_k)(z - \hat{z}_k) + \tilde{\rho}^y(\xi_k, p_k, v_k, y_k)(u - \hat{u}_k)\Big\},\\
E(q_k, u_k) - E(q_{kh}, u_{kh}) &\approx \frac{1}{2}\Big\{\tilde{\rho}^u(q_{kh}, u_{kh})(y_k - \hat{y}_{kh}) + \tilde{\rho}^z(q_{kh}, u_{kh}, z_{kh})(v_k - \hat{v}_{kh})\\
&\qquad + \tilde{\rho}^v(\xi_{kh}, p_{kh}, v_{kh})(z_k - \hat{z}_{kh}) + \tilde{\rho}^y(\xi_{kh}, p_{kh}, v_{kh}, y_{kh})(u_k - \hat{u}_{kh})\Big\},\\
E(q_{kh}, u_{kh}) - E(q_\sigma, u_\sigma) &\approx \frac{1}{2}\Big\{\tilde{\rho}^q(q_\sigma, u_\sigma, z_\sigma)(p_{kh} - \hat{p}_\sigma) + \tilde{\rho}^p(\xi_\sigma, p_\sigma, v_\sigma, y_\sigma)(q_{kh} - \hat{q}_\sigma)\Big\}.
\end{aligned}$$

As for the estimator for the error in the cost functional, we employed here the fact, that the terms

$$\begin{aligned}
&\tilde{\rho}^q(q_k, u_k, z_k)(p - \hat{p}_k), && \tilde{\rho}^p(\xi_k, p_k, v_k, y_k)(q - \hat{q}_k),\\
&\tilde{\rho}^q(q_{kh}, u_{kh}, z_{kh})(p_k - \hat{p}_{kh}), && \tilde{\rho}^p(\xi_{kh}, p_{kh}, v_{kh}, y_{kh})(q_k - \hat{q}_{kh}),\\
&\tilde{\rho}^u(q_\sigma, u_\sigma)(y_{kh} - \hat{y}_\sigma), && \tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(v_{kh} - \hat{v}_\sigma),\\
&\tilde{\rho}^v(\xi_\sigma, p_\sigma, v_\sigma)(z_{kh} - \hat{z}_\sigma), && \tilde{\rho}^y(\xi_\sigma, p_\sigma, v_\sigma, y_\sigma)(u_{kh} - \hat{u}_\sigma)
\end{aligned}$$

vanish if $\hat{p}_k$, $\hat{q}_k$, $\hat{p}_{kh}$, $\hat{q}_{kh}$, $\hat{y}_\sigma$, $\hat{v}_\sigma$, $\hat{z}_\sigma$, $\hat{u}_\sigma$ are chosen appropriately.

As shown, for the error estimation with respect to the cost functional no additional equations have to be solved. However, the error estimation with respect to a given quantity of interest requires the computation of the three auxiliary variables p_σ, v_σ, y_σ. This additional numerical effort is comparable to the execution of one step of Newton's method employed for solving the optimization problem.

6.4 Evaluation of the error estimators

In this section, we concretize the a posteriori error estimator developed in the previous sections for the case of cG(1)dG(0) and cG(1)cG(1) space-time discretizations on quadrilateral meshes in two space dimensions. That is, we consider the combination of dG(0) or cG(1) time discretization with piecewise bilinear finite elements for the space discretization. As before, we present the detailed derivation for the dG(0) time discretization, the cG(1) discretization can be treated in exactly the same manner.

6.4.1 Approximation of the weights

The error estimates presented in the previous sections still contain the unknown solutions u, z, and q as well as their semidiscrete analogs. When appearing in the weights, that is in the differences $q - \hat{q}_k$, $u - \hat{u}_k$, $z - \hat{z}_k$, and so on, they are approximated using interpolations in higher-order finite element spaces. This approach relies on the "super-closeness" of the constructed higher-order interpolants to the corresponding exact solutions. It has been observed to work very satisfactory in the context of a posteriori error estimation; see for example Becker and Rannacher [10].

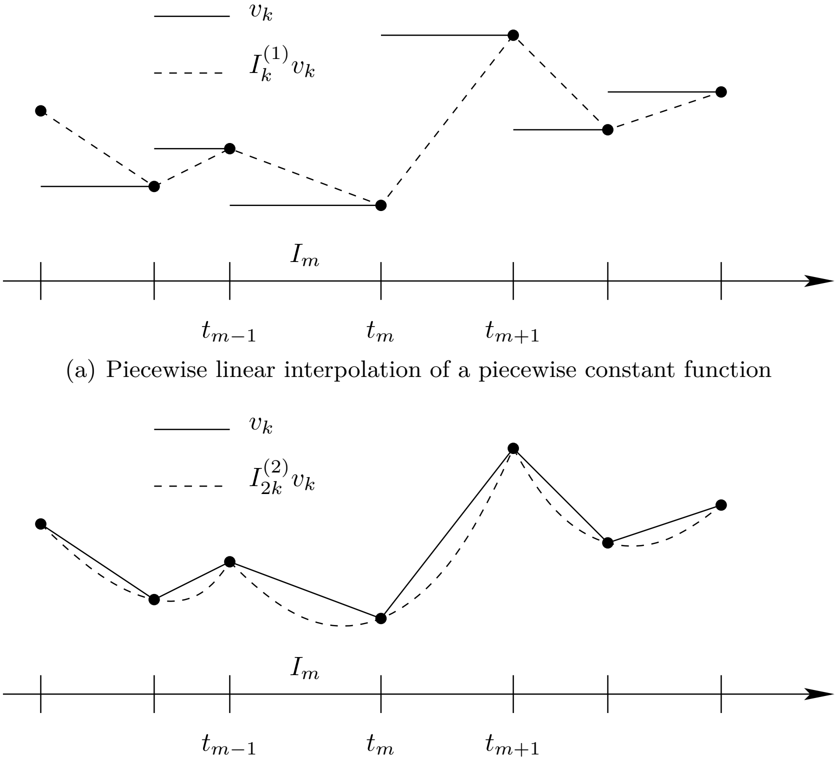
To define this higher-order approximation, we introduce linear operators P_h, P_k, and P_d, which map the computed solutions to the approximations of the interpolation errors:

$$\begin{aligned}
z - \hat{z}_k &\approx P_k z_k, & u - \hat{u}_k &\approx P_k u_k, & y - \hat{y}_k &\approx P_k y_k, & v - \hat{v}_k &\approx P_k v_k,\\
z_k - \hat{z}_{kh} &\approx P_h z_{kh}, & u_k - \hat{u}_{kh} &\approx P_h u_{kh}, & y_k - \hat{y}_{kh} &\approx P_h y_{kh}, & v_k - \hat{v}_{kh} &\approx P_h v_{kh},\\
q_{kh} - \hat{q}_\sigma &\approx P_d q_\sigma, & & & p_{kh} - \hat{p}_\sigma &\approx P_d p_\sigma.
\end{aligned}$$

For the considered case of cG(1)dG(0) and cG(1)cG(1) discretizations of the state space, the operators P_k and P_h are chosen as

$$\begin{aligned}
P_k &= I_k^{(1)} - \mathrm{id} \qquad \text{with} \qquad I_k^{(1)} \colon \widetilde{X}_k^0 \to X_k^1 \qquad \text{for cG(1)dG(0)},\\
P_k &= I_{2k}^{(2)} - \mathrm{id} \qquad \text{with} \qquad I_{2k}^{(2)} \colon X_k^1 \to X_{2k}^2 \qquad \text{for cG(1)cG(1)},\\
P_h &= I_{2h}^{(2)} - \mathrm{id} \qquad \text{with} \qquad I_{2h}^{(2)} \colon V_h^1 \to V_{2h}^2 \qquad \text{for cG(1)dG(0) and cG(1)cG(1)}.
\end{aligned}$$

Here, the elements of X_{2k}^2 are constructed as piecewise quadratic polynomials defined on unions of two adjacent subintervals. The actions of the piecewise linear and piecewise quadratic interpolation operators $I_k^{(1)}$ and $I_{2k}^{(2)}$ in time are depicted in Figure 6.1.



(a) Piecewise linear interpolation of a piecewise constant function

(b) Piecewise quadratic interpolation of a piecewise linear function

Figure 6.1. Action of the interpolation operators $I_k^{(1)}$ and $I_{2k}^{(2)}$

The piecewise biquadratic spatial interpolation $I_{2h}^{(2)}$ into the space V_{2h}^2 consisting of biquadratic finite elements on patches of cells can easily be computed since the underlying mesh is required to provide a patch structure; see Section 3.2. That is, one can always combine four adjacent cells to a macro cell on which the biquadratic interpolation can be defined. An example of such a patched mesh is shown in Figure 3.3 in Section 3.2. The interpolation $I_{2h}^{(2)}$ defined on V_h^1 is extended to functions v_{kh} in $\widetilde{X}_{k,h}^{0,1}$ and $X_{k,h}^{1,1}$ pointwise in time

$$\left(I_{2h}^{(2)} v_{kh}\right)(t) := I_{2h}^{(2)} v_{kh}(t).$$

The choice of P_d depends on the discretization of the control space Q for which we have presented several reasonable possibilities in Section 3.3. If the finite-dimensional subspaces Q_d are constructed like the discrete state spaces, P_d can be chosen as a modification of the operators P_k and P_h defined above. If for example the control q depends only on time and the discretization is done with piecewise constant polynomials, we can choose $P_d = I_d^{(1)} - \mathrm{id}$. If the control space Q is already finite-dimensional, which is usually the case in the context of parameter estimation, it is possible to choose $P_d = 0$ and thus, the estimator for the error $J(q_{kh}, u_{kh}) - J(q_\sigma, u_\sigma)$ is zero—as well as this discretization error itself.

Remark 6.3. The error estimator for the error due to discretization of the control space vanishes also if the optimality conditions for q_σ and p_σ

$$\widetilde{\mathcal{L}}'_q(\xi_\sigma)(\cdot) = 0 \quad \text{and} \quad \widetilde{\mathcal{L}}''_{qq}(\xi_\sigma)(p_\sigma, \cdot) + \widetilde{\mathcal{L}}''_{uq}(\xi_\sigma)(v_\sigma, \cdot) + \widetilde{\mathcal{L}}''_{zq}(\xi_\sigma)(y_\sigma, \cdot) + E'_q(q_\sigma, u_\sigma)(\cdot) = 0$$

are fulfilled not only in a variational sense for all test functions $\delta q \in Q_d$ but also pointwise. Then, the corresponding residuals

$$\tilde{\rho}^q(\xi_\sigma)(\cdot) \quad \text{and} \quad \tilde{\rho}^p(\xi_\sigma, p_\sigma, v_\sigma, y_\sigma)(\cdot)$$

are zero independently of the choice of the test functions. This situation is usually found in problems where the control enters linearly the right-hand side, the boundary conditions, or the initial condition and the discrete control space is chosen as the discrete state space. For instance, there is no error due to the discretization of the control space Q for the Examples 2.1 and 2.3 when choosing the discrete spaces appropriately. We refer to the a priori error analysis derived in Section 5.4 where similar observations are made for the optimal control problem considered there.

In order to make the error representations from the previous sections computable, we now replace all unknown solutions appearing in the residuals and weights by their fully discrete analogs. That is, we replace for instance

$$\tilde{\rho}^u(q_k, u_k)(P_k z_k) \quad \text{by} \quad \tilde{\rho}^u(q_\sigma, u_\sigma)(P_k z_\sigma) \quad \text{and} \quad \tilde{\rho}^u(q_{kh}, u_{kh})(P_h z_{kh}) \quad \text{by} \quad \tilde{\rho}^u(q_\sigma, u_\sigma)(P_h z_\sigma).$$

While the replacement of the unknown solutions in the weights seems uncritical and is also well accepted, one can argue about the replacement of the solution in the residuals, that is, about the replacement of the linearization point. Here too, it would be possible to replace the continuous and semidiscrete solutions by higher order interpolations of the discrete solutions. However, the numerical results (see Section 6.7 and Chapter 7) yield that we can pass on this additional effort and that the proposed replacement of the unknown solutions by their discrete analogs is sufficient.

Remark 6.4. This observation is substantiated by the fact, that the errors caused by the discussed replacement are usually of "higher order": In the concrete configuration of the optimization problem from Example 2.1 with cG(1)dG(0) discretization of the state and the control, we have accordingly to Remark 6.3 that $\xi_{kh} = (q_{kh}, u_{kh}, z_{kh})$ coincides with $\xi_\sigma = (q_\sigma, u_\sigma, z_\sigma)$. Hence, in the given situation, only the error caused by the replacement of $\xi_k = (q_k, u_k, z_k)$ by $\xi_{kh} = \xi_\sigma$ needs to be considered. In the case of error estimation with respect to the cost functional, it is given in terms of derivatives of the Lagrangian as

$$\widetilde{\mathcal{L}}'(\xi_k)(\xi - \hat{\xi}_k) - \widetilde{\mathcal{L}}'(\xi_\sigma)(\xi - \hat{\xi}_k) = \widetilde{\mathcal{L}}''(\xi_k + s(\xi_\sigma - \xi_k))(\xi - \hat{\xi}_k, \xi_k - \xi_\sigma)$$

with some parameter $s \in [0,1]$. When choosing $\hat{\xi}_k$ to be some interpolant of ξ, it can be shown that this error is of order $\mathcal{O}(kh^2)$, whereas the error $J(q,u) - J(q_\sigma, u_\sigma)$ itself is not better than $\mathcal{O}(k+h^2)$; cf. Theorem 5.26 in Section 5.4.3.

By the proposed procedure, we obtain the computable a posteriori error estimate

$$J(q,u) - J(q_\sigma, u_\sigma) \approx \eta_k^J + \eta_h^J + \eta_d^J$$

for the cost functional J, where estimators η_k^J, η_h^J, and η_d^J are given by

$$\begin{aligned}
\eta_k^J &:= \frac{1}{2}\Big\{\tilde{\rho}^u(q_\sigma, u_\sigma)(P_k z_\sigma) + \tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_k u_\sigma)\Big\},\\
\eta_h^J &:= \frac{1}{2}\Big\{\tilde{\rho}^u(q_\sigma, u_\sigma)(P_h z_\sigma) + \tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_h u_\sigma)\Big\},\\
\eta_d^J &:= \frac{1}{2}\tilde{\rho}^q(q_\sigma, u_\sigma, z_\sigma)(P_d q_\sigma).
\end{aligned}$$

By proceeding similarly, we obtain for the quantity of interest E the error estimate

$$E(q,u) - E(q_\sigma, u_\sigma) \approx \eta_k^E + \eta_h^E + \eta_d^E$$

with the estimators η_k^E, η_h^E, and η_d^E given by

$$\begin{aligned}
\eta_k^E &:= \frac{1}{2}\Big\{\tilde{\rho}^u(q_\sigma, u_\sigma)(P_k y_\sigma) + \tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_k v_\sigma)\\
&\qquad + \tilde{\rho}^v(\xi_\sigma, v_\sigma, p_\sigma)(P_k z_\sigma) + \tilde{\rho}^y(\xi_\sigma, v_\sigma, y_\sigma, p_\sigma)(P_k u_\sigma)\Big\},\\
\eta_h^E &:= \frac{1}{2}\Big\{\tilde{\rho}^u(q_\sigma, u_\sigma)(P_h y_\sigma) + \tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_h v_\sigma)\\
&\qquad + \tilde{\rho}^v(\xi_\sigma, v_\sigma, p_\sigma)(P_h z_\sigma) + \tilde{\rho}^y(\xi_\sigma, v_\sigma, y_\sigma, p_\sigma)(P_h u_\sigma)\Big\},\\
\eta_d^E &:= \frac{1}{2}\Big\{\tilde{\rho}^q(q_\sigma, u_\sigma, z_\sigma)(P_d p_\sigma) + \tilde{\rho}^p(\xi_\sigma, v_\sigma, y_\sigma, p_\sigma)(P_d q_\sigma)\Big\}.
\end{aligned}$$

To give an impression of the terms that have to be evaluated when using these error estimators, we present for the implicit Euler variant of the cG(1)dG(0) discretization the explicit form of the state residuals $\tilde{\rho}^u(q_\sigma, u_\sigma)(P_k z_\sigma)$ and $\tilde{\rho}^u(q_\sigma, u_\sigma)(P_h z_\sigma)$ and the adjoint residuals $\tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_k u_\sigma)$ and $\tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_h u_\sigma)$ used by η_k^J and η_h^J for estimating the discretization error due to time and space discretization of the state variable. We evaluate the arising integrals in time for the residuals weighted with z_σ or u_σ by the box rule and for the residuals weighted with $I_k^{(1)} z_\sigma$ or $I_k^{(1)} u_\sigma$ by the trapezoidal rule. Hence, we have to assume the right-hand side f to be continuous in time, that is $f \in C(\bar{I}, H)$. Then, we obtain with the abbreviations $Q_m := q_{\sigma,m}^-$, $U_m := u_{\sigma,m}^-$, and $Z_m := z_{\sigma,m}^-$ known from Section 3.4 the following parts of the error estimators:

$$\begin{aligned}
\tilde{\rho}^u(q_\sigma, u_\sigma)(P_k z_\sigma) = \sum_{m=1}^{M} \Big\{ &(U_m - U_{m-1}, Z_m - Z_{m-1}) + \frac{k_m}{2}\bar{a}(Q_m, U_m)(Z_m - Z_{m-1})\\
&+ \frac{k_m}{2}(f(t_{m-1}), Z_{m-1}) - \frac{k_m}{2}(f(t_m), Z_m)\Big\},
\end{aligned}$$

$$\begin{aligned}\tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_k u_\sigma) = \sum_{m=1}^{M} \Big\{ &\frac{k_m}{2} \bar{a}_u'(Q_m, U_m)(U_m, Z_m) - \frac{k_m}{2} \bar{a}_u'(Q_{m-1}, U_{m-1})(U_{m-1}, Z_m) \\ &+ \frac{k_m}{2} J_1'(U_{m-1})(U_{m-1}) - \frac{k_m}{2} J_1'(U_m)(U_m) \Big\},\end{aligned}$$

$$\begin{aligned}\tilde{\rho}^u(q_\sigma, u_\sigma)(P_h z_\sigma) = \sum_{m=1}^{M} \Big\{ &k_m(f(t_m), I_{2h}^{(2)} Z_m - Z_m) - k_m \bar{a}(Q_m, U_m)(I_{2h}^{(2)} Z_m - Z_m) \\ &- (U_m - U_{m-1}, I_{2h}^{(2)} Z_m - Z_m) \Big\} - (U_0 - u_0(q_\sigma), I_{2h}^{(2)} Z_0 - Z_0),\end{aligned}$$

$$\begin{aligned}\tilde{\rho}^z(q_\sigma, u_\sigma, z_\sigma)(P_h u_\sigma) = \sum_{m=1}^{M} \Big\{ &k_m J_1'(U_m)(I_{2h}^{(2)} U_m - U_m) - k_m \bar{a}_u'(Q_m, U_m)(I_{2h}^{(2)} U_m - U_m, Z_m) \\ &+ (I_{2h}^{(2)} U_{m-1} - U_{m-1}, Z_m - Z_{m-1}) \Big\} + J_2'(U_M)(I_{2h}^{(2)} U_M - U_M) \\ &- (I_{2h}^{(2)} U_M - U_M, Z_M).\end{aligned}$$

For the cG(1)cG(1) discretization, the terms that have to be evaluated are very similar and the evaluation can be treated as presented here for the cG(1)dG(0) discretization. Of course, to evaluate the time integrals for the residuals weighted with $I_{2k}^{(2)} u_\sigma$ and $I_{2k}^{(2)} z_\sigma$ exactly, higher order quadrature formulas have to be employed.

6.4.2 Localization of the error estimators

The presented a posteriori error estimators are directed towards two aims: assessment of the discretization error and improvement of the accuracy by adaptive refinement of the underlying discretizations. For the second aim, the information provided by the error estimators has to be localized to cellwise or nodewise contributions (*local error indicators*). We concretize this procedure here for the error estimators η_k^J and η_h^J assessing the error with respect to the cost functional. For concrete choices of discretizations for the control space one can proceed with η_d^J in the same manner. Of course, the error indicators η_k^E, η_h^E, and η_d^E can be treated similarly, too.

For localizing the error estimators, we split up the error estimates η_k^J and η_h^J into their contributions on each subinterval I_m by

$$\eta_k^J = \sum_{m=1}^{M} \eta_k^{J,m} \qquad \text{and} \qquad \eta_h^J = \sum_{m=0}^{M} \eta_h^{J,m},$$

where the contributions $\eta_k^{J,m}$ and $\eta_h^{J,m}$ are given in terms of the time stepping residuals $\tilde{\rho}_m^u$ and $\tilde{\rho}_m^z$ as

$$\begin{aligned}\eta_k^{J,m} &= \frac{1}{2}\Big\{ \tilde{\rho}_m^u(q_\sigma, u_\sigma)(P_k z_\sigma) + \tilde{\rho}_m^z(q_\sigma, u_\sigma, z_\sigma)(P_k u_\sigma) \Big\}, \\ \eta_h^{J,m} &= \frac{1}{2}\Big\{ \tilde{\rho}_m^u(q_\sigma, u_\sigma)(P_h z_\sigma) + \tilde{\rho}_m^z(q_\sigma, u_\sigma, z_\sigma)(P_h u_\sigma) \Big\}.\end{aligned}$$

Thereby, the time stepping residuals $\tilde{\rho}^u_m$ and $\tilde{\rho}^z_m$ are those parts of the global residuals $\tilde{\rho}^u$ and $\tilde{\rho}^z$ belonging to the time interval I_m or to the initial time $t = 0$ for $m = 0$.

Whereas the temporal indicators $\eta_k^{J,m}$ can be used directly for determining the time intervals to be refined, the indicators $\eta_h^{J,m}$ for the spatial discretization error have to be further localized to indicators on each spatial mesh. Since a direct localization of $\eta_h^{J,m}$ by separating the contributions of the different mesh cells leads to large over-estimation of the error due to the oscillatory behavior of the residual terms (see Carstensen and Verführt [20]), the localization is often done by using integration by parts in space (see for instance Becker and Rannacher [9, 10]). To compute the indicators obtained by this procedure, the strong formulation of the differential operator and jump terms of the discrete solution over faces of the mesh cells have to be evaluated.

We avoid this additional computations by using the following technique introduced in Braack and Ern [16], which also leads to local error indicators with the correct local order of convergence. For doing so, we consider the Lagrange nodal bases

$$\{\, \varphi_i^m \mid i = 1, 2, \dots, N_m \,\}$$

of $V_h^{1,m}$ $(m = 0, 1, \dots, M)$ with $N_m \coloneqq \dim V_h^{1,m}$. By means of these bases, we introduce the sets of quadratic nodal functions

$$\left\{\, \psi_i^m \coloneqq I_{2h}^{(2)} \varphi_i^m \;\middle|\; i = 1, 2, \dots, N_m \,\right\} \subseteq V_{2h}^{2,m}$$

associated with each node of the triangulation $\mathcal{T}_h^m$. Let Ψ_m^u and Ψ_m^z be the difference of the contributions of the state and adjoint residuals with respect to the bilinear basis $\{\, \varphi_i^m \,\}$ and the biquadratic basis $\{\, \psi_i^m \,\}$, that is

$$\Psi_{m,i}^u \coloneqq \tilde{\rho}_m^u(q_\sigma, u_\sigma)(\psi_i^m - \varphi_i^m) \qquad \text{and} \qquad \Psi_{m,i}^z \coloneqq \tilde{\rho}_m^z(q_\sigma, u_\sigma, z_\sigma)(\psi_i^m - \varphi_i^m).$$

Since for the considered case of dG(0) time discretization, u_σ and z_σ are constant in time on the interval I_m, we have

$$\begin{aligned} u_\sigma\big|_{I_m} &= \sum_{i=1}^{N_m} \varphi_i^m U_i^m, & z_\sigma\big|_{I_m} &= \sum_{i=1}^{N_m} \varphi_i^m Z_i^m, \\ I_{2h}^{(2)} u_\sigma\big|_{I_m} &= \sum_{i=1}^{N_m} \psi_i^m U_i^m, & I_{2h}^{(2)} z_\sigma\big|_{I_m} &= \sum_{i=1}^{N_m} \psi_i^m Z_i^m, \end{aligned}$$

with $U^m, Z^m \in \mathbb{R}^{N_m}$ the nodal vectors of $u_\sigma\big|_{I_m}$ and $z_\sigma\big|_{I_m}$, respectively. Thus, we can formulate the spatial error indicators as

$$\eta_h^{J,m} = \frac{1}{2}\{\langle \Psi_m^u, Z^m \rangle + \langle \Psi_m^z, U^m \rangle\},$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product on $\mathbb{R}^{N_m}$.

Further, we introduce a *filtering operator* π given by

$$\pi \coloneqq \mathrm{id} - I_{2h}^{(1)} \qquad \text{with} \qquad I_{2h}^{(1)} \colon \widetilde{X}_{k,h}^{0,1} \to \widetilde{X}_{k,2h}^{0,1}.$$

The spaces $V_{2h}^{1,m}$, which are implicitly used in the definition of $\widetilde{X}_{k,2h}^{0,1}$, are spaces of bilinear finite elements on patches. This construction again makes use of the patch-structure of the triangulation, which implies that the finite element spaces $V_{2h}^{1,m}$ and $V_h^{1,m}$ are nested, that is $V_{2h}^{1,m} \subseteq V_h^{1,m}$.

We denote the nodal vectors of the filtered solution of the state equation $\pi u_\sigma\big|_{I_m}$ and of the adjoint solution $\pi z_\sigma\big|_{I_m}$ by $U^{\pi,m}$ and $Z^{\pi,m}$ given as

$$\pi u_\sigma\big|_{I_m} = \sum_{i=1}^{N_m} \varphi_i^m U_i^{\pi,m} \qquad \text{and} \qquad \pi z_\sigma\big|_{I_m} = \sum_{i=1}^{N_m} \varphi_i^m Z_i^{\pi,m}.$$

Since $I_{2h}^{(2)}$ is the identity on $V_{2h}^{1,m}$, we have the equality

$$I_{2h}^{(2)} \pi\varphi_i^m - \pi\varphi_i^m = I_{2h}^{(2)} \varphi_i^m - \varphi_i^m = \psi_i^m - \varphi_i^m.$$

Then, the linearity of the residuals with respect to the weights implies (cf. Braack and Ern [16])

$$\eta_h^{J,m} = \frac{1}{2}\{\langle \Psi_m^u, Z^m\rangle + \langle \Psi_m^z, U^m\rangle\} = \frac{1}{2}\{\langle \Psi_m^u, Z^{\pi,m}\rangle + \langle \Psi_m^z, U^{\pi,m}\rangle\}.$$

We obtain the computable quantities

$$\eta_{h,i}^{J,m} := \frac{1}{2}\{\Psi_{m,i}^u Z_i^{\pi,m} + \Psi_{m,i}^z U_i^{\pi,m}\}, \quad i = 1, 2, \ldots, N_m$$

providing the upper bound

$$\left|\eta_h^{J,m}\right| \leq \sum_{i=1}^{N_m} \left|\eta_{h,i}^{J,m}\right|$$

for the error estimator $\eta_h^{J,m}$.

While the error estimator η_h^J itself is almost independent on the size of the time steps (cf. Table 6.1 in Section 6.7.1), its parts $\eta_h^{J,m}$ depend linearly on the size of the (possibly locally refined) time steps k_m. For obtaining independent spatial error indicators which can be treated simultaneously in a mesh adaptation process, it is necessary to get rid of this dependence. This can be done by rescaling the indicators $\eta_{h,i}^{J,m}$ by means of the reference time step $\hat{k} := {}^{T}\!/_{M}$ as

$$\tilde{\eta}_{h,i}^{J,m} := \frac{\hat{k}}{k_m} \eta_{h,i}^{J,m}, \quad i = 1, 2, \ldots, N_m,\ m = 0, 1, \ldots, M.$$

By reassembling the computed nodewise error indicators $\tilde{\eta}_{h,i}^{J,m}$, we obtain cellwise indicators $\tilde{\eta}_{h,K}^{J,m}$ for cells $K \in \mathcal{T}_h^m$ suitable for the usage in an adaptive refinement procedure.

Hence, we end up with two sets of error indicators, one for the temporal and one for the spatial discretization error, given as

$$\Sigma_k := \left\{\ \eta_k^{J,m} \ \middle|\ m = 1, 2, \ldots, M\ \right\} \quad \text{and} \quad \Sigma_h := \left\{\ \tilde{\eta}_{h,K}^{J,m} \ \middle|\ K \in \mathcal{T}_h^m,\ m = 0, 1, \ldots, M\ \right\}.$$

In the next section, we state an adaptive refinement algorithm for the automatic choice of suitable discretizations, which bases on these sets of indicators.

6.5 Adaptive refinement algorithm

Goal of the adaption of the different types of discretizations has to be the equilibrated reduction of the corresponding discretization errors. For this purpose, it is crucial to have reliable and especially quantitative information about the sizes of the different error contributions. This is provided by the error estimates derived in the Sections 6.2 and 6.3.

If a given tolerance TOL has to be reached, the equilibration can be done by refining each discretization as long as the value of this part of the error estimator is greater than TOL/ν if ν different types of discretizations are considered. Typically we have $\nu \in \{2,3,4\}$ depending on the type of discretization of the control space: no discretization, only time or space discretization or discretization in time and space. We present here a strategy which equilibrates the different discretization errors even if no tolerance is given.

Aim of the equilibration algorithm presented in the sequel is to obtain discretizations such that

$$|\eta^{(1)}| \approx |\eta^{(2)}| \approx \cdots \approx |\eta^{(\nu)}|,$$

and to keep this property during further refinement. Here, the estimators $\eta^{(i)}$ may denote some of the estimators η_k^J, η_h^J, and η_d^J for the cost functional J or η_k^E, η_h^E, and η_d^E for the quantity of interest E.

For doing this equilibration, we choose an *equilibration factor* $\kappa \geq 1$ (usually $\kappa \approx 5$) and propose the following strategy: We compute a permutation $(i_1, i_2, \dots, i_\nu)$ of the indices $(1, 2, \dots, \nu)$ such that

$$|\eta^{(i_1)}| \geq |\eta^{(i_2)}| \geq \cdots \geq |\eta^{(i_\nu)}|,$$

and define the relations

$$\gamma_j := \left| \frac{\eta^{(i_j)}}{\eta^{(i_{j+1})}} \right| \geq 1, \quad j = 1, 2, \dots, \nu - 1.$$

Then, we decide by means of Algorithm 6.1 in every repetition of the adaptive refinement algorithm given by Algorithm 6.2, which discretizations shall be refined. For simplicity, we present Algorithm 6.2 for the case of three discretizations symbolized by $\sigma = (k, h, d)$ as already introduced before.

Algorithm 6.1. Equilibration algorithm

```
Require: The relations $\gamma_j$, $j = 1, 2, \dots, \nu - 1$ are computed.
for $j = \nu - 1$ downto $1$ do
    if $\gamma_j > \kappa$ then
        Refine discretizations $i_1, i_2, \dots, i_j$.
        return
Refine all discretizations.
```

In Algorithm 6.1, we test all γ_j, $j = \nu - 1, \nu - 2, \dots, 1$, for $\gamma_j > \kappa$. That is, we start by testing the quotient $\gamma_{\nu-1}$ of the two error estimators $|\eta^{(i_{\nu-1})}|$ and $|\eta^{(i_\nu)}|$ with the smallest values. We break whenever we reach a quotient γ_j with $\gamma_j > \kappa$ and refine all discretizations $i_1, i_2, \dots, i_j$

with corresponding error contributions larger than $|\eta^{(i_{j+1})}|$. If all quotients γ_j fulfill $\gamma_j \leq \kappa$, then the errors are equilibrated and we refine all discretizations for a further reduction of the error.

Algorithm 6.2. Adaptive refinement algorithm

```
Choose an initial triple of discretizations $\mathcal{T}_{\sigma_0}$, $\sigma_0 = (k_0, h_0, d_0)$ for the space-time discretization of the states and an appropriate discretization of the controls.
Set $l = 0$.
loop
    Compute the optimal solution pair $(q_{\sigma_l}, u_{\sigma_l})$.
    Evaluate the a posteriori error estimators $\eta_{k_l}$, $\eta_{h_l}$ and $\eta_{d_l}$.
    if the given maximal degree of refinement is reached then
        return
    Determine the discretization(s) to be refined by means of Algorithm 6.1.
    Separately refine the selected discretizations using the information from the corresponding set of error indicators obtained from the estimators $\eta_{k_l}$, $\eta_{h_l}$, or $\eta_{d_l}$. Obtain the new discretization $\mathcal{T}_{\sigma_{l+1}}$.
    Increment $l$.
```

The termination rule in Algorithm 6.2 is formulated in terms of degrees of refinement. That is, we stop Algorithm 6.2 if a given maximal degree of refinement (a maximal number of cells or subintervals) is reached. If, of course, a given tolerance has to be met, the stopping criterion can be stated by means of a comparison of the sum of the estimators $\eta_l := \eta_{k_l} + \eta_{h_l} + \eta_{d_l}$ and the given tolerance.

For every discretization to be adapted, we select the cells for refinement by means of sets of local error indicators like Σ_k and Σ_h introduced at the end of the previous section. Hence, we have to choose subsets $\Sigma_k^R \subseteq \Sigma_k$ and $\Sigma_h^R \subseteq \Sigma_h$ of cells to be refined. Then, we refine the time intervals corresponding to the indicators in Σ_k^R and the cells of the spatial triangulations corresponding to the indicators in Σ_h^R. That is, we apply the selection procedure for the spatial cells simultaneously on all triangulations $\mathcal{T}_h^m$, $m = 0, 1, \ldots, M$.

Several standard approaches are available for choosing such subsets. For the computations done in this thesis, we use a selection scheme which differs from most other methods and is presented for instance in Richter [68, 69]. For convenience of the reader, we briefly sketch its key idea for a prototypical set of error indicators $\Sigma = \{\, \eta_i \mid i = 1, 2, \ldots, N \,\}$, which is assumed to be the localization of some error indicator η: At first, we compute a permutation $(i_1, i_2, \ldots, i_N)$ of the indices $(1, 2, \ldots, N)$ such that

$$|\eta_{i_1}| \geq |\eta_{i_2}| \geq \cdots \geq |\eta_{i_N}|.$$

The subset $\Sigma^R \subseteq \Sigma$ of indicators to be determined is always chosen as coherent queue $\Sigma^R = \{\, \eta_{i_1}, \eta_{i_2}, \ldots, \eta_{i_r} \,\}$. Thereby, the number r is determined by

$$r = \arg \min_{1 \leq r \leq N} \mathcal{E}(r) \mathcal{N}(r)^{\delta}, \tag{6.10}$$

where $\mathcal{E}(r)$ is a prediction of the discretization error on the refined discretization and $\mathcal{N}(r)$ is the number of degrees of freedom in the refined discretization. The parameter δ depends on the degree of the polynomials used for discretization and on the dimension of the domain to be discretized. The explicit formulation of (6.10) can be found in Richter [69] and a justification of this approach under some regularity assumptions is given in Braack [15].

The practical behavior of the algorithms presented in this section in concrete configurations is demonstrated in Section 6.7 and in Chapter 7.

6.6 A heuristic error estimator

For substantiating the capabilities of the systematic and quantitative error estimation techniques derived in the Sections 6.2 and 6.3, we introduce additionally a heuristic estimator for the spatial discretization error based on the smoothness of the optimal state u and the optimal adjoint state z. After motivating this estimator, we use it in Section 6.7.3 as reference in numerical comparison to the quantitative error estimator.

To derive the heuristic smoothness-based error estimator, we examine the classical Poisson equation

$$\begin{aligned} -\Delta u &= f \qquad \text{in } \Omega \\ u &= 0 \qquad \text{on } \partial\Omega \end{aligned}$$

on a domain $\Omega \subset \mathbb{R}^2$. For this equation, the estimator for assessing the discretization error with respect to the energy norm $\|\nabla u - \nabla u_h\|$ is given for instance in Verfürht [80] by

$$\|\nabla u - \nabla u_h\| \leq C_I \left(\sum_{K \in \mathcal{T}_h} h_K^2 \{\rho_K(u_h)^2 + \rho_{\partial K}(u_h)^2\} \right)^{\frac{1}{2}} \tag{6.11}$$

with the *cell residuals* $\rho_K(u_h)$ and the *jump residuals* $\rho_{\partial K}(u_h)$ defined as

$$\rho_K(u_h) := \|f + \Delta u_h\|_{L^2(K)} \quad \text{and} \quad \rho_{\partial K}(u_h) := \frac{1}{2} h_K^{-\frac{1}{2}} \|[\partial_n u_h]\|_{L^2(\partial K)}.$$

Additionally, estimate (6.11) involves an interpolation constant C_I whose value is in general unknown.

It is known (cf. Carstensen and Verfürht [20]), that in the case of bilinear finite elements the influence of the cell residuals $\rho_K(u_h)$ can usually be disregarded compared to the jump residuals $\rho_{\partial K}(u_h)$. There holds

$$\rho_K(u_h) = \rho_{\partial K}(u_h) + \mathcal{O}(h_K).$$

Furthermore, the jump $[\partial_n u_h]$ of the normal derivatives of u_h over faces ∂K can be estimated by a suitable recovery of the second derivatives of u_h. This recovery process can be done for instance by means of the biquadratic interpolant $I_{2h}^{(2)} u_h$ (see Section 6.4) leading to

$$\rho_{\partial K}(u_h) \approx \|\nabla^2 I_{2h}^{(2)} u_h\|_{L^2(K)}.$$

In total, we obtain the following approximation to the energy estimator (6.11):

$$\|\nabla u - \nabla u_h\| \lesssim C_I \left(\sum_{K \in \mathcal{T}_h} h_K^2 \|\nabla^2 I_{2h}^{(2)} u_h\|_{L^2(K)}^2 \right)^{\frac{1}{2}}.$$

Hence, we define the heuristic energy-based error estimator $\tilde{\eta}_h(u_h)$ by

$$\tilde{\eta}_h(u_h) := C_I \left(\sum_{K \in \mathcal{T}_h} \tilde{\eta}_{h,K}(u_h)^2 \right)^{\frac{1}{2}} \quad \text{with the indicators} \quad \tilde{\eta}_{h,K}(u_h) := h_K \|\nabla^2 I_{2h}^{(2)} u_h\|_{L^2(K)}.$$

Applying this result to the prototypical stationary optimal control problem

$$\text{Minimize } J(q,u) \text{ such that } \begin{cases} -\Delta u = q & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases}$$

leads under incorporation of the adjoint equation to the estimator

$$\tilde{\eta}_h := \tilde{\eta}_h(u_h) + \tilde{\eta}_h(z_h).$$

Even if this estimator is motivated for stationary problems, we use it for spatial refinement in the here considered situation of time-dependent optimization problems to compare the results obtained from this process to the results obtained from the application of the quantitative error estimators derived in the Sections 6.2 and 6.3. Therefore, we rewrite the estimator in terms of the solutions u_σ and z_σ of the fully discretized optimization problem as

$$\tilde{\eta}_h = \tilde{\eta}_h(u_\sigma) + \tilde{\eta}_h(z_\sigma).$$

and apply it separately to each time interval I_m and the corresponding triangulation $\mathcal{T}_h^m$. Similarly, a heuristic estimator for the error due to time discretization can be derived. However, we restrict ourselves here to the more interesting case of comparing different local refinements of the spatial triangulation.

6.7 Numerical results

Our aim for this section is to substantiate the methods for error estimation and mesh adaptation developed in the previous sections of this chapter. Therefore, we examine two prototypical configurations similar to the examples discussed in Section 2.2. In the examples considered here, we assess the error with respect to the cost functional. More involved applications including the estimation of the error in terms of a quantity of interest which is different from the cost functional are treated in Chapter 7. In the final Section 6.7.3, we substantiate our approach to error estimation by comparing its performance to those of the heuristic approach described in the previous section.

6.7.1 Time-dependent Neumann boundary control

We consider the linear parabolic state equation on the two-dimensional unit square $\Omega = (0,1)^2$ (see Figure 6.2) with final time $T = 1$ given by

$$\begin{aligned} \partial_t u - \varepsilon \Delta u + u &= f && \text{in } \Omega \times I, \\ \partial_n u &= 0 && \text{on } \Gamma_0 \times I, \\ \partial_n u &= q_i && \text{on } \Gamma_i \times I,\ i = 1,2, \\ u &= 0 && \text{on } \Omega \times \{\,0\,\}. \end{aligned} \tag{6.12}$$

The control $q = (q_1, q_2)$ acts as a purely time-dependent boundary control of Neumann type on the two parts of the boundary denoted by Γ_1 and Γ_2. Thus, the control space Q is chosen as $Q = [L^2(I,R)]^2$ with $R = \mathbb{R}$ and the spaces V and H used in the definition of the state space X are set to $V = H^1(\Omega)$ and $H = L^2(\Omega)$.

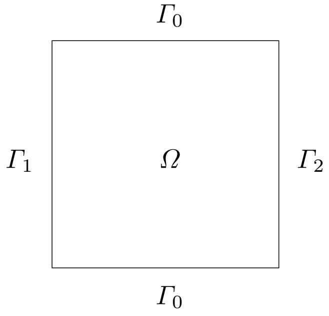


Figure 6.2. Computational domain Ω

As cost functional J to be minimized subject to the state equation (6.12) we choose

$$J(q,u) = \frac{1}{2}\int_I \|u(t) - 1\|^2_{L^2(\Omega)}\,dt + \frac{\alpha}{2}\int_I \{q_1^2(t) + q_2^2(t)\}\,dt$$

of tracking type endowed with a L^2-regularization term.

For the computations, the right-hand side f of (6.12) is chosen as

$$f(t,x) = 10t\exp\left(1 - \frac{1}{1 - 100\|x - \tilde{x}\|^2}\right) \quad \text{with} \quad \tilde{x} = \left(\frac{2}{3}, \frac{1}{2}\right)^T,$$

and the parameters α and ε are set to $\alpha = 0.1$ and $\varepsilon = 0.1$, respectively.

The discretization of the state space is done here via the cG(1)cG(1) space-time Galerkin method which is a variant of the Crank-Nicolson scheme (cf. Section 3.4.2). Consequently, the state is discretized in time by piecewise linear and the adjoint state by piecewise constant polynomials. The controls are discretized using piecewise constant polynomials on a partition of the time interval I which has to be at most as fine as the time discretization of the states (cf. the discussions in the Sections 5.3 and 6.3).

At first, we present in Table 6.1 the numerical justification for splitting the total discretization error in three parts regarding the discretizations of time, space, and control. The table

Table 6.1. Independence of one part of the error estimator on the refinement of the other parts

M	N	$\dim Q_d$	η_k^J	η_h^J	η_d^J
256	289	16		$-4.9104\cdot 10^{-04}$	$-8.6152\cdot 10^{-04}$
512	289	16		$-4.9110\cdot 10^{-04}$	$-8.6232\cdot 10^{-04}$
1024	289	16	—	$-4.9111\cdot 10^{-04}$	$-8.6251\cdot 10^{-04}$
2048	289	16		$-4.9111\cdot 10^{-04}$	$-8.6256\cdot 10^{-04}$
4096	289	16		$-4.9112\cdot 10^{-04}$	$-8.6258\cdot 10^{-04}$
1024	25	16	$-3.8360\cdot 10^{-07}$		$-8.7015\cdot 10^{-04}$
1024	81	16	$-4.3463\cdot 10^{-07}$		$-8.5900\cdot 10^{-04}$
1024	289	16	$-4.5039\cdot 10^{-07}$	—	$-8.6251\cdot 10^{-04}$
1024	1089	16	$-4.5529\cdot 10^{-07}$		$-8.6398\cdot 10^{-04}$
1024	4225	16	$-4.6096\cdot 10^{-07}$		$-8.6432\cdot 10^{-04}$
4096	289	16	$-2.8171\cdot 10^{-08}$	$-4.9112\cdot 10^{-04}$	
4096	289	32	$-3.0332\cdot 10^{-08}$	$-4.8826\cdot 10^{-04}$	
4096	289	64	$-3.1317\cdot 10^{-08}$	$-4.8688\cdot 10^{-04}$	—
4096	289	128	$-3.1704\cdot 10^{-08}$	$-4.8651\cdot 10^{-04}$	
4096	289	256	$-3.1828\cdot 10^{-08}$	$-4.8642\cdot 10^{-04}$	

demonstrates the independence of each part of the error estimator on the refinement of the other parts. This feature is especially important to reach an equilibration of the discretization errors by applying the adaptive refinement algorithm given in Section 6.5.

Table 6.2. Local refinement with equilibration

N	M	$\dim Q_d$	η_h^J	η_k^J	η_d^J	e^J	I_{eff}
25	64	16	$2.0\cdot 10^{-03}$	$-9.7\cdot 10^{-05}$	$-8.5\cdot 10^{-04}$	$-2.567\cdot 10^{-04}$	-0.23
81	64	20	$-1.0\cdot 10^{-03}$	$-1.1\cdot 10^{-04}$	$-3.2\cdot 10^{-04}$	$-7.818\cdot 10^{-04}$	0.50
289	64	20	$-4.8\cdot 10^{-04}$	$-1.3\cdot 10^{-04}$	$-3.2\cdot 10^{-04}$	$-8.009\cdot 10^{-04}$	0.84
813	74	32	$-2.2\cdot 10^{-05}$	$-4.7\cdot 10^{-05}$	$-1.3\cdot 10^{-04}$	$-2.116\cdot 10^{-04}$	1.02
813	74	48	$-2.2\cdot 10^{-05}$	$-4.8\cdot 10^{-05}$	$-7.7\cdot 10^{-05}$	$-1.493\cdot 10^{-04}$	1.01
2317	87	76	$1.1\cdot 10^{-05}$	$-2.7\cdot 10^{-05}$	$-2.9\cdot 10^{-05}$	$-4.559\cdot 10^{-05}$	1.00
8213	104	128	$2.7\cdot 10^{-06}$	$-1.8\cdot 10^{-05}$	$-1.3\cdot 10^{-05}$	$-2.842\cdot 10^{-05}$	0.96
8213	208	128	$2.7\cdot 10^{-06}$	$-4.3\cdot 10^{-06}$	$-1.5\cdot 10^{-05}$	$-1.661\cdot 10^{-05}$	0.99
8213	208	192	$2.7\cdot 10^{-06}$	$-4.2\cdot 10^{-06}$	$-7.0\cdot 10^{-06}$	$-8.335\cdot 10^{-06}$	0.97

Table 6.2 shows the development of the discretization error $e^J := J(q,u) - J(q_\sigma, u_\sigma)$ and the a posteriori error estimators η_k^J, η_h^J, and η_d^J during an adaptive run with local refinement of all three types of discretizations. Here and in what follows, M denotes the number of time steps, N denotes the number of nodes in the spatial mesh, and $\dim Q_d$ is the number of degrees of freedom for the discretization of the control. The effectivity index to measure the quality of the error estimator given in the last column of this table is defined by $I_{\text{eff}} := {e^J}/{\eta^J}$ using the estimation for the total discretization error given by $\eta^J := \eta_k^J + \eta_h^J + \eta_d^J$. In Table 6.2, we observe for finer discretizations that $I_{\text{eff}} \approx 1$. This demonstrates the very good quantitative assessment of the discretization error by the developed estimator.

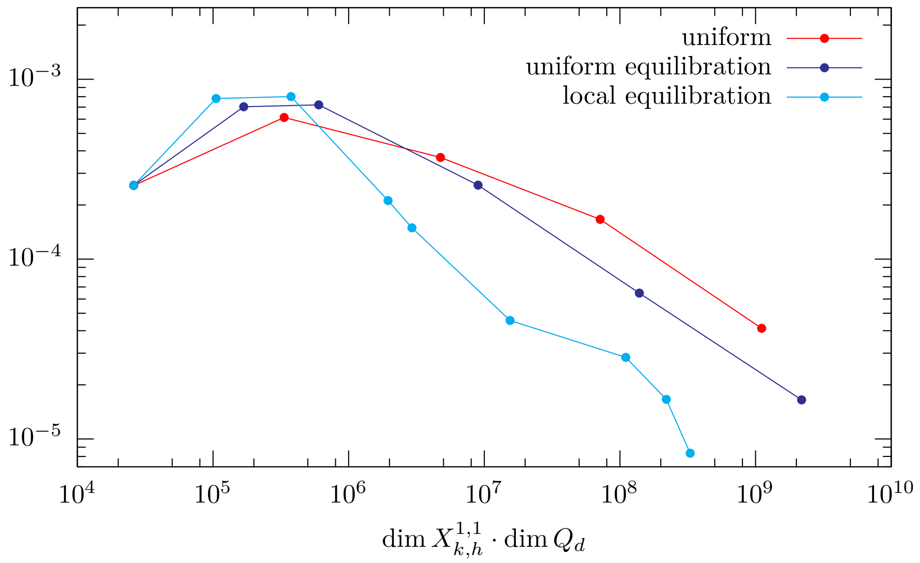


Figure 6.3. Comparison of the error $|e^J|$ for different refinement strategies

A comparison of the error e^J for the different refinement strategies is depicted in Figure 6.3. Therein, the following labeling is used:

- “uniform”: Here, we apply uniform refinement of all discretizations after each run of the optimization loop.
- “uniform equilibration”: Here, we still allow only uniform refinements but use the error estimators within the equilibration strategy (Algorithm 6.1) to decide which discretizations have to be refined.
- “local equilibration”: Here, we combine local refinement of all discretizations with the proposed equilibration strategy. Thereby, only one spatial triangulation is employed for the whole time interval I.

The figure shows for example, that to reach a discretization error of $4 \cdot 10^{-5}$ the uniform refinement needs about 70 times the number of degrees of freedom the fully adaptive refinement needs. This reduction of degrees of freedoms reflects also in a significant saving of computational costs.

6.7.2 Space- and time-dependent control by right-hand side

As second exemplary configuration, we consider the following nonlinear optimal control problem: The governing equation is given on $\Omega = (0, 3) \times (0, 1)$ with final time $T = 1$ by

$$\begin{aligned} \partial_t u - \Delta u + u^3 &= q && \text{in } \Omega \times I, \\ u &= 0 && \text{on } \Gamma \times I, \\ u &= u_0 && \text{on } \Omega \times \{\, 0 \,\}. \end{aligned} \tag{6.13}$$

The initial condition u_0 is given by means of $g\colon \mathbb{R} \to \mathbb{R}$,

$$g(s) = \begin{cases} \exp\Bigl(1 - \frac{1}{1-s^2}\Bigr) & |s| < 1, \\ 0 & \text{otherwise,} \end{cases}$$

as

$$u_0(x) = g\Bigl(\frac{10}{3}|x - \tilde{x}|\Bigr) \quad \text{with} \quad \tilde{x} = \Bigl(\frac{1}{2}, \frac{1}{2}\Bigr)^T,$$

and is depicted in Figure 6.4. The control q depends on space and time and acts as source term in the domain. Thus, the control space Q is chosen here as $Q = L^2(I, R)$ with $R = L^2(\Omega)$. Because of the homogeneous Dirichlet boundary conditions, the spaces V and H used for defining the state space X are set to $V = H^1_0(\Omega)$ and $H = L^2(\Omega)$.

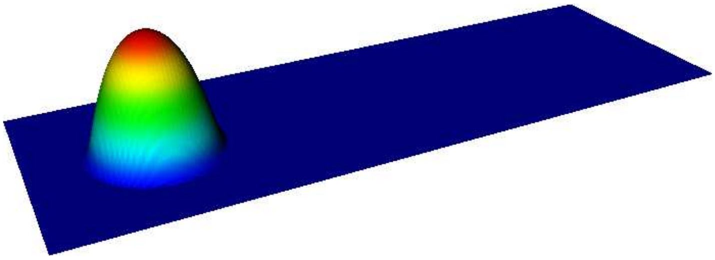


Figure 6.4. Initial condition u_0

As functional J to be minimized subject to the state equation (6.13) we consider

$$J(q,u) = \frac{1}{2}\|u(T) - \hat{u}\|^2_{L^2(\Omega)} + \frac{\alpha}{2}\int_I \|q(t)\|^2_{L^2(\Omega)}\,dt.$$

Thereby, the desired state $\hat{u}$ is given bilateral symmetric to the initial condition as

$$\hat{u}(x) = g\Bigl(\frac{10}{3}|x - \tilde{x}|\Bigr) \quad \text{with} \quad \tilde{x} = \Bigl(\frac{5}{2}, \frac{1}{2}\Bigr)^T.$$

It is shown in Figure 6.5. The regularization parameter α is chosen to be $\alpha = 10^{-2}$.

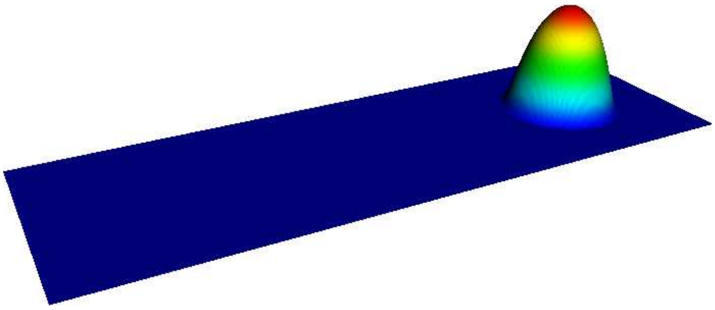


Figure 6.5. Desired state $\hat{u}$

In the computations, the control space was discretized as the state space by the cG(1)dG(0) method. Because of this choice, the optimality condition (gradient equation) can be fulfilled pointwise, and thus there is no additional discretization error arising from discretizing the control (cf. Remark 6.3).

Table 6.3 shows the behavior of the equilibration strategy used to balance the errors due to space and time discretization on locally refined discretizations using dynamically changing meshes. Thereby, N_{tot} denotes the total number of degrees of freedom which is also given by $\dim \widetilde{X}^{0,1}_{k,h}$, the dimension of the discrete state space. Conversely, $N_{\max}$ denotes the maximal number of degrees of freedom used in one spatial mesh. As already introduced, M is the number of time steps. We observe in Table 6.3 the desired equilibration of the spatial and temporal contributions to the overall error estimator.

Table 6.3. Local refinement on dynamic meshes with equilibration

N_{tot}	$N_{\max}$	M	η_h^J	η_k^J
441	21	20	$2.0 \cdot 10^{-02}$	$-3.6 \cdot 10^{-03}$
531	51	20	$9.2 \cdot 10^{-05}$	$-1.6 \cdot 10^{-03}$
633	51	22	$1.1 \cdot 10^{-04}$	$-1.4 \cdot 10^{-03}$
837	51	26	$1.1 \cdot 10^{-04}$	$-1.2 \cdot 10^{-03}$
1041	51	30	$1.1 \cdot 10^{-04}$	$-9.5 \cdot 10^{-04}$
1245	51	34	$1.2 \cdot 10^{-04}$	$-6.9 \cdot 10^{-04}$
1449	51	38	$1.3 \cdot 10^{-04}$	$-5.0 \cdot 10^{-04}$
5981	173	60	$-2.8 \cdot 10^{-04}$	$-2.7 \cdot 10^{-04}$
27445	431	104	$-1.2 \cdot 10^{-04}$	$-1.4 \cdot 10^{-04}$
133703	1281	194	$-3.7 \cdot 10^{-05}$	$-7.2 \cdot 10^{-05}$
380585	3673	210	$-1.0 \cdot 10^{-05}$	$-6.1 \cdot 10^{-05}$
760855	3673	406	$-1.0 \cdot 10^{-05}$	$-3.0 \cdot 10^{-05}$
1520389	3673	796	$-1.0 \cdot 10^{-05}$	$-1.5 \cdot 10^{-05}$

Comparisons of different refinement strategies for separate refinement of the time and space discretizations are given in the Figures 6.6 and 6.7. Thereby, we consider for the space discretization additionally to uniform and adaptive local refinement on a fixed triangulation the promising approach of dynamically changing meshes introduced in Section 3.2.2. The depicted errors $e_k^J := J(q_{hd}, u_{hd}) - J(q_\sigma, u_\sigma)$ and $e_h^J := J(q_{kd}, u_{kd}) - J(q_\sigma, u_\sigma)$ are defined by means of approximations of the values for $J(q_{hd}, u_{hd})$ and $J(q_{kd}, u_{kd})$ computed by extrapolation. The figures also depict the very good approximation of the discretization error by the corresponding estimators η_k^J and η_h^J.

The development of the total discretization error under uniform refinement and under local refinement using dynamic meshes and equilibration is compared in Figure 6.8. We observe a remarkable reduction of the degrees of freedom of both the discretization of the states and the controls by a factor of 43.

Figure 6.9 depicts the distribution of the time steps obtained by local refinement based on the error indicators. Thereby, we observe a strong refinement towards the end of the time interval. This behavior is hardly surprising since the objective functional for both the optimization and the error estimation acts only on the final time T. The coarsest and finest time step size used differ by 12 levels of refinement.

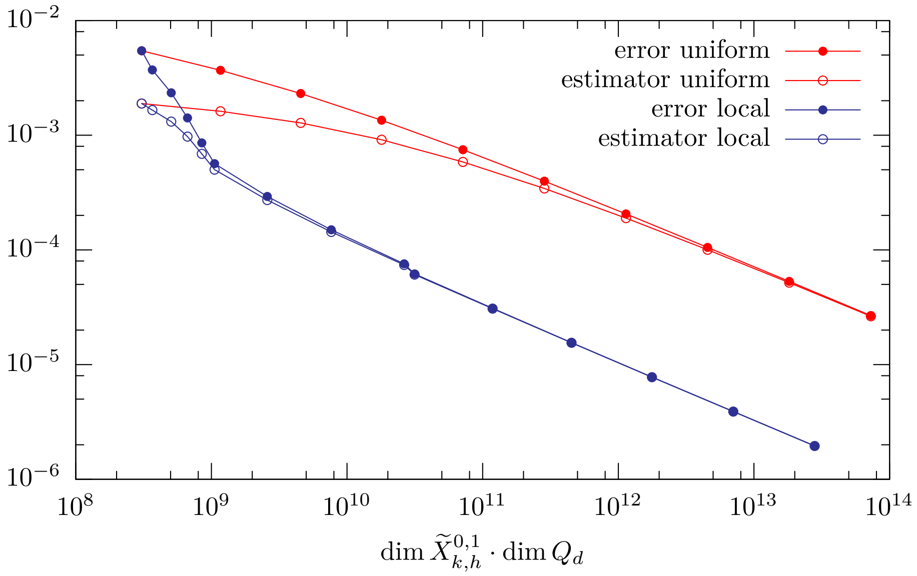


Figure 6.6. Comparison of the error $|e_k^J|$ and the estimator $|\eta_k^J|$ for uniform and local refinement of the time steps

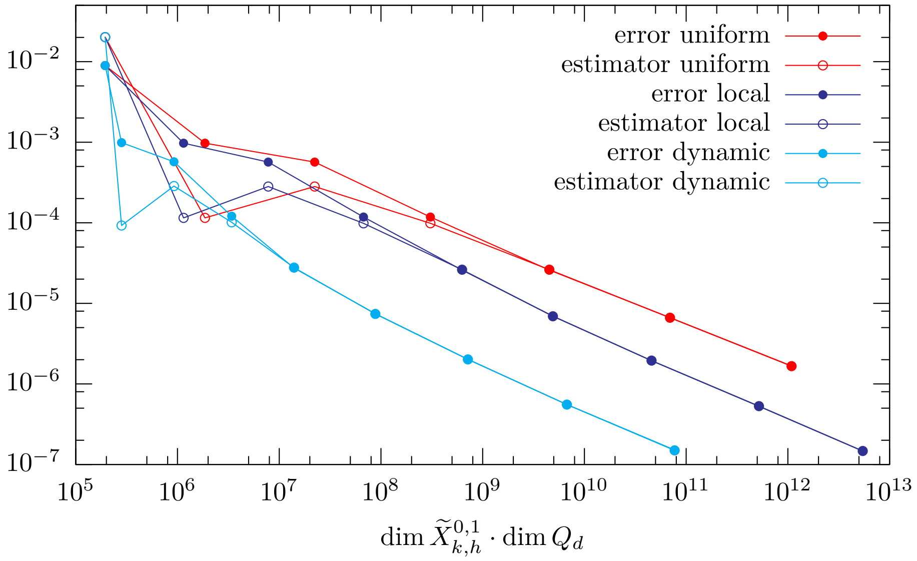


Figure 6.7. Comparison of the error $|e_h^J|$ and the estimator $|\eta_h^J|$ for uniform and local refinement of a fixed spatial triangulation as well as local refinement on dynamically changing meshes

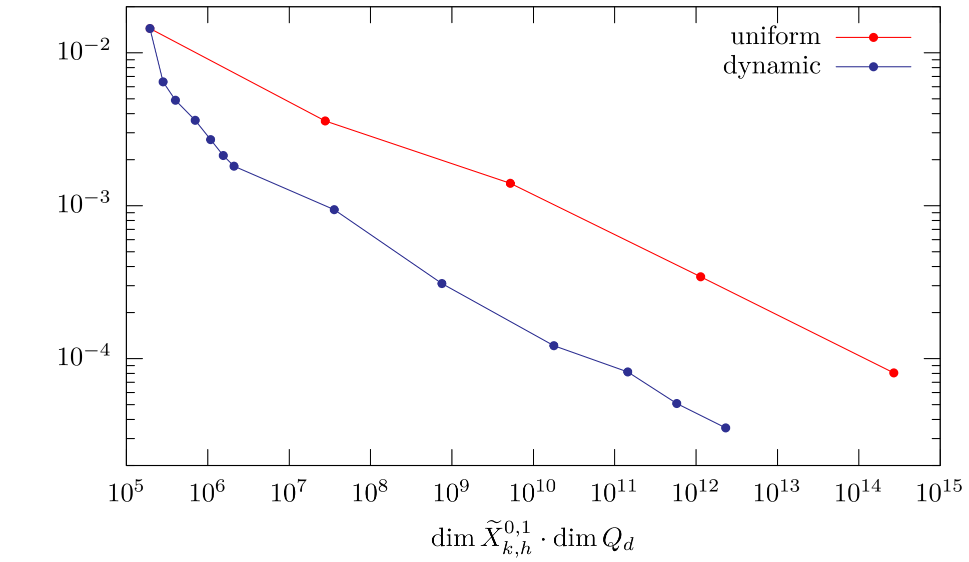


Figure 6.8. Comparison of the error $|e^J|$ for uniform and local space-time refinement on dynamically changing meshes using equilibration

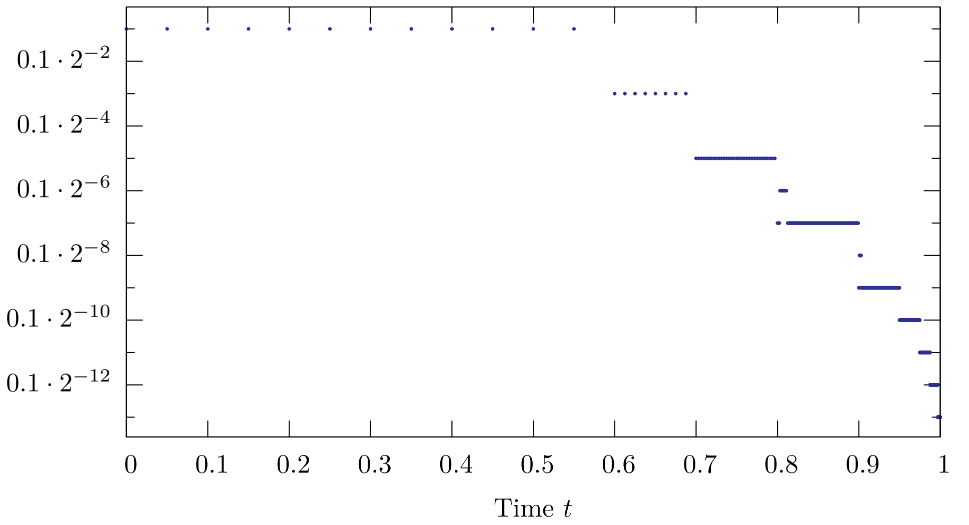
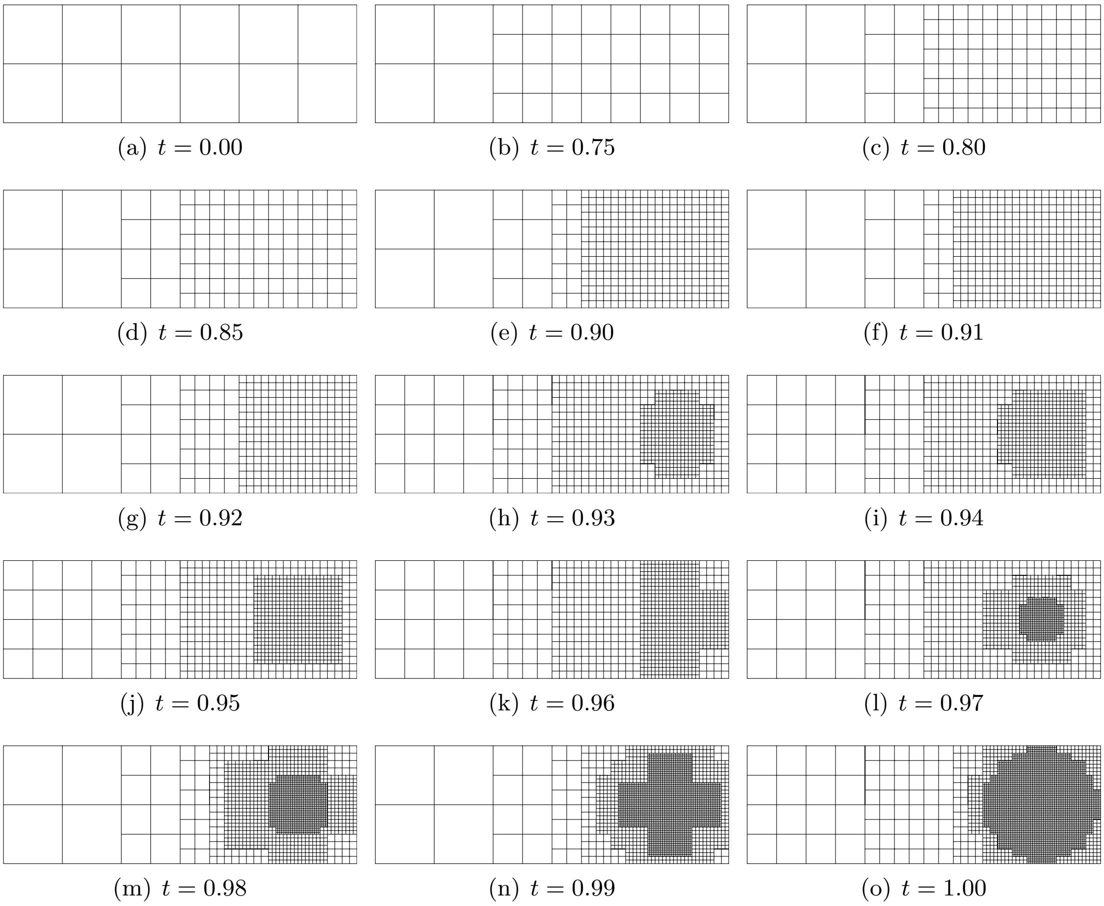


Figure 6.9. Visualization of the adaptively determined time step size k

Finally, we present in Figure 6.10 a selection of adaptively refined triangulations obtained from the computation on dynamically changing meshes. It is remarkable that due to the presented techniques of error estimation, the meshes near the final time $t = T$ are much more refined than near the initial time $t = 0$ although the optimal state exploits almost the same structure at $t = 0$ and $t = T$ (cf. Figures 6.4 and 6.5). Furthermore, we observe that three quarters of the time interval, that is for $t \in [0, 0.75]$, the grid is kept almost unrefined.



(a) $t = 0.00$ (b) $t = 0.75$ (c) $t = 0.80$

(d) $t = 0.85$ (e) $t = 0.90$ (f) $t = 0.91$

(g) $t = 0.92$ (h) $t = 0.93$ (i) $t = 0.94$

(j) $t = 0.95$ (k) $t = 0.96$ (l) $t = 0.97$

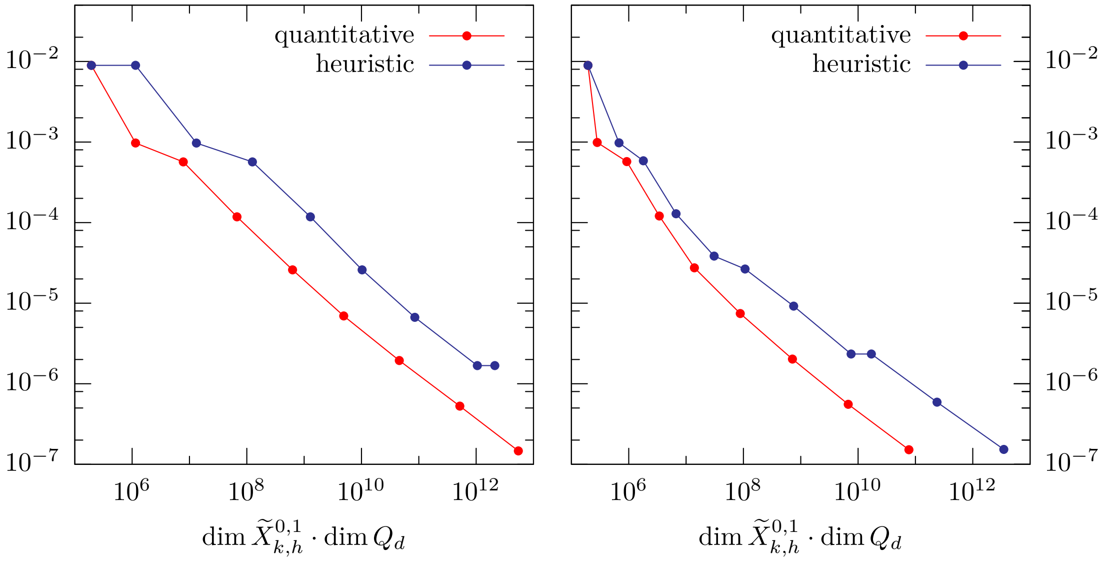
(m) $t = 0.98$ (n) $t = 0.99$ (o) $t = 1.00$

Figure 6.10. Spatial triangulations at certain time points

6.7.3 Comparison to a heuristic error estimator

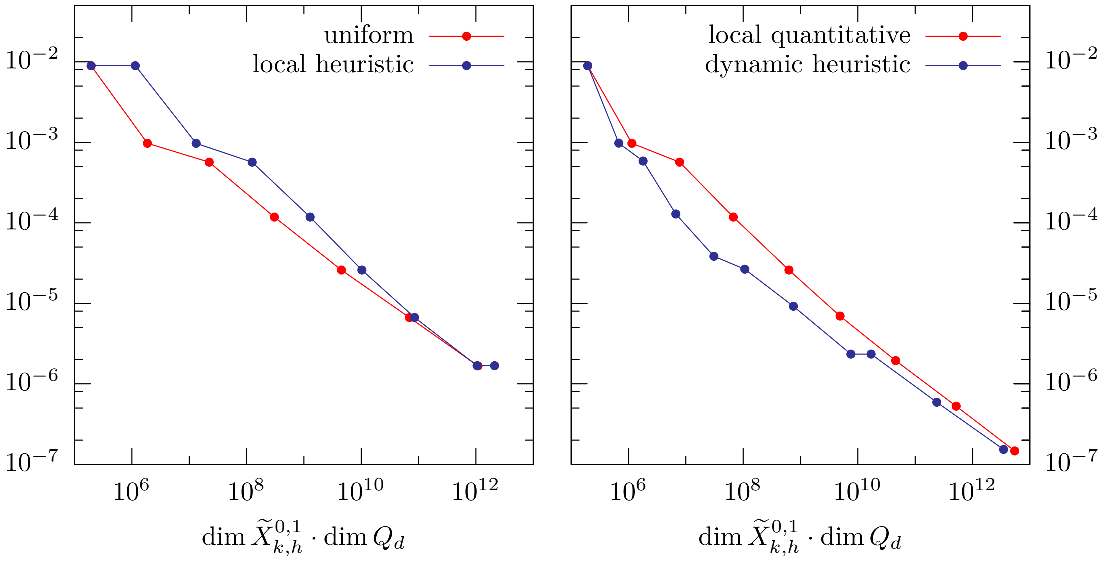
To compare the quality of the different discretizations obtained by the heuristic and quantitative error estimators, we consider the model configuration investigated in Section 6.7.2. Thereby, the configuration used for the numerical tests is the same as employed there for obtaining the results depicted in Figure 6.7. That is, we consider only refinement of the spatial triangulations for a fixed number of M time steps.

At first, we compare the resulting discretization errors for spatial refinement based on the quantitative error estimator η_h^J and based on the heuristic error estimator $\tilde{\eta}_h$ for both one fixed



(a) Local refinement on a fixed spatial triangulation (b) Local refinement on dynamically changing meshes

Figure 6.11. Comparison of the discretization errors $|e_h^J|$ obtained by the heuristic and quantitative error estimators



(a) Local refinement on a fixed spatial triangulation obtained by the heuristic estimator vs. uniform refinement

(b) Local refinement on dynamically changing meshes obtained by the heuristic estimator vs. local refinement on a fixed spatial triangulation obtained from the quantitative estimator

Figure 6.12. Comparison of the discretization error $|e_h^J|$ obtained by the heuristic error estimator to other types of refinements

spatial triangulation and using dynamically changing meshes. This comparison is depicted in Figure 6.11. We observe in both presented situations, that the mesh generated by η_h^J is more efficient at reducing the error e_h^J in terms of the cost functional. Moreover, we note especially in Figure 6.11(b) a "smoother" reduction of the error on meshes obtained by the quantitative estimator. Figure 6.11(b) also shows a slightly better order of convergence of the discretization error on meshes constructed by means of η_h^J than by means of $\tilde{\eta}_h$.

Nevertheless, one may conclude from these first observations, that there is no or only a minor benefit from using the quantitative error estimator. However, by inspection of the results given in Figure 6.12, this impression turns out to be wrong. Figure 6.12(a) compares the errors on a sequence of uniformly refined triangulations with the error on locally refined meshes obtained by means of the heuristic estimator. We observe, that during the whole computation the error on the heuristically refined meshes is larger than the error on the sequence of uniform refined meshes. Furthermore, Figure 6.12(b) shows that there is almost no benefit from using the heuristic estimator on dynamically changing meshes compared to the quantitative estimator on a fixed spatial triangulation.

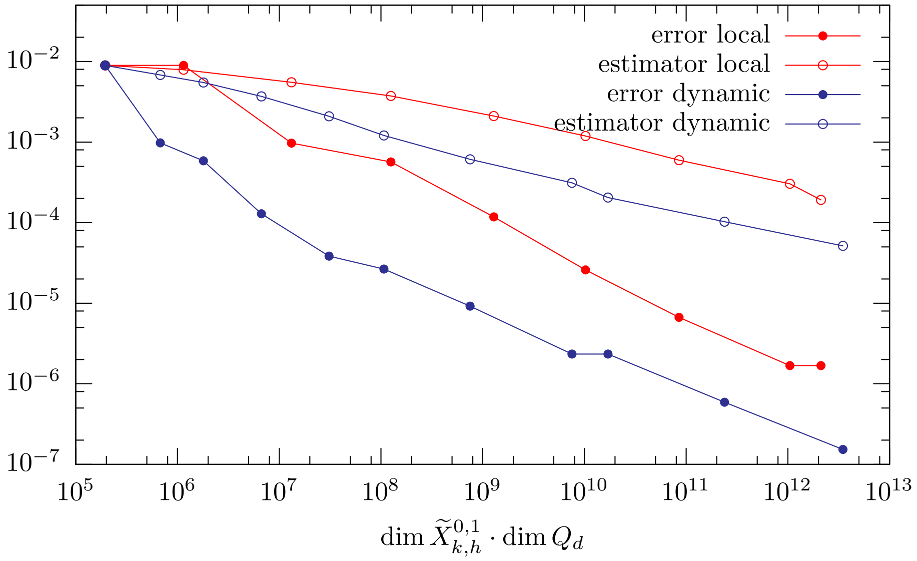
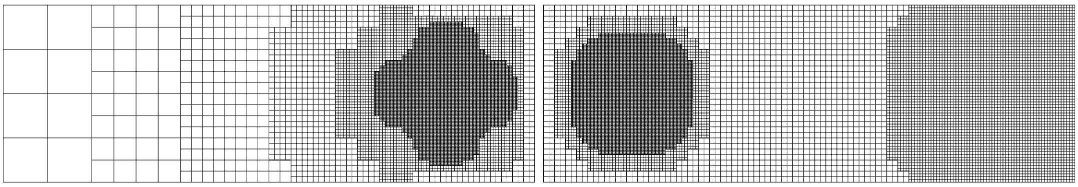


Figure 6.13. Comparison of the error $|e_h^J|$ and the heuristic estimator $|\tilde{\eta}_h|$ for local refinement of a fixed spatial triangulation as well as local refinement on dynamically changing meshes with $C_I \approx 9.64 \cdot 10^{-5}$

A further drawback of the heuristic error estimator $\tilde{\eta}_h$ is that it is not constant-free. That is, it contains the constant C_I whose value has to be determined suitably a priori. If—as in this model configuration—the value of the error is known, one can choose C_I such that at least on one mesh the relation $e_h^J \approx \tilde{\eta}_h$ holds. The development of the error $|e_h^J|$ and the estimator $|\tilde{\eta}_h|$ with the adjusted constant $C_I \approx 9.64 \cdot 10^{-5}$ is depicted in Figure 6.13 for local refinement on a fixed mesh and on dynamically changing meshes. We emphasize, that such a adjustment of C_I is virtually impossible in concrete applications. Consequently, the error estimation obtained from $\tilde{\eta}_h$ can in general not be used within a space-time adaptive algorithm where the refinement strategy necessitates reliable information on the sizes of the errors for equilibrating all involved discretization errors (cf. Section 6.5).

Furthermore, we observe in Figure 6.13 that even if C_I is adjusted properly, the assessment of the considered error in terms of the cost functional by the heuristic estimator $\tilde{\eta}_h$ is not satisfactory. The estimator exhibits a different order of convergence than the error. This is reasonable since $\tilde{\eta}_h$ is constructed to estimate the error in terms of the $H^1(\Omega)$-seminorm and in the considered case the functional acts on $L^2(I, L^2(\Omega))$.



(a) Quantitative estimator η_h^J

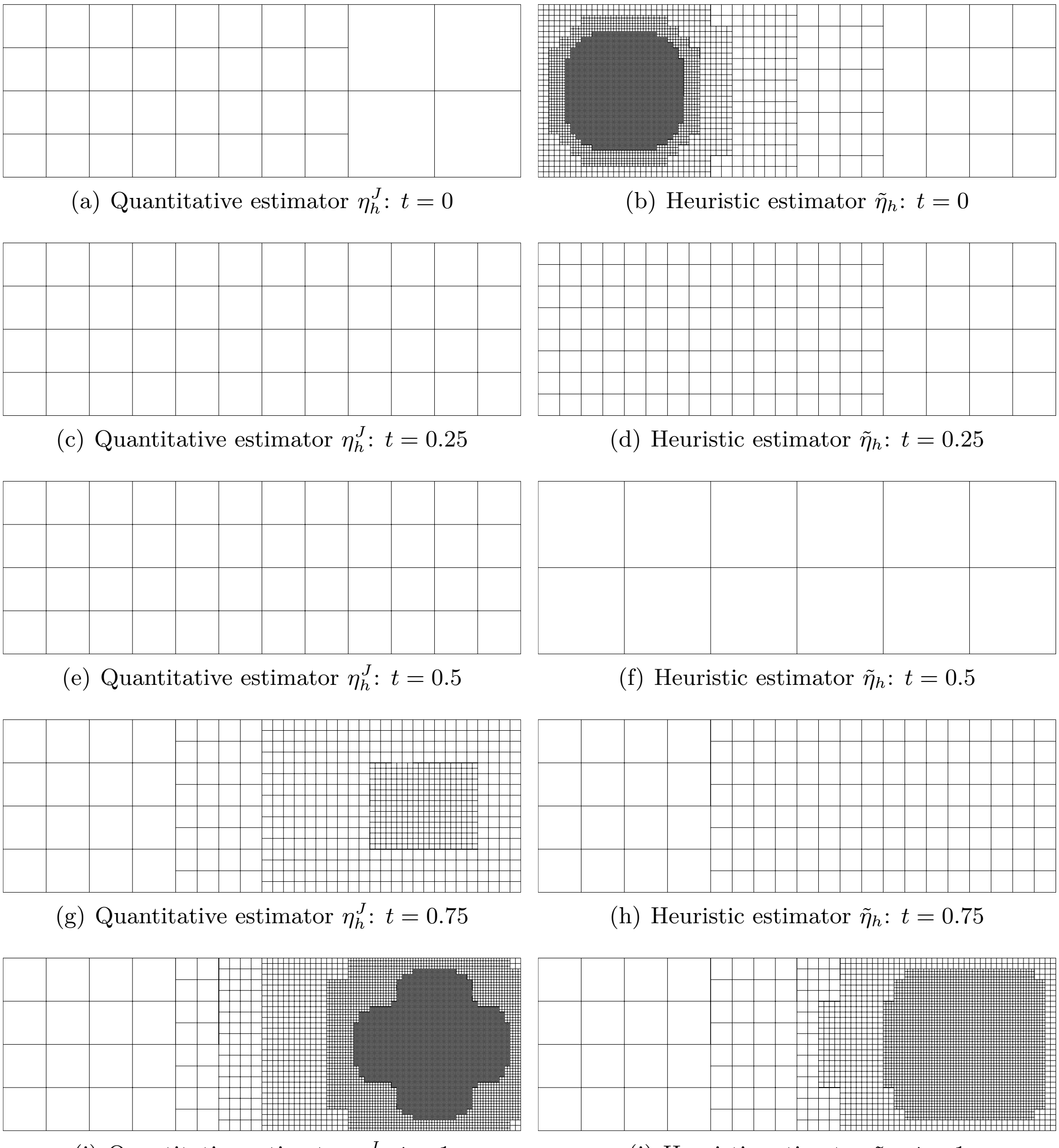
(b) Heuristic estimator $\tilde{\eta}_h$

Figure 6.14. Comparison of the local refinement obtained by the two considered estimators after six refinement cycles using a fixed spatial triangulation

A hint for this altogether poor behavior of the heuristic error estimator $\tilde{\eta}_h$ can be found when considering the produced locally refined triangulations given in the Figures 6.14 and 6.15. Figure 6.14 shows the two meshes obtained by means of the quantitative and heuristic error estimators η_h^J and $\tilde{\eta}_h$ using one fixed spatial triangulation for the whole time interval I. We note, that the mesh produced by η_h^J (see Figure 6.14(a)) is only refined on the right part of the domain. In contrast, estimator $\tilde{\eta}_h$ advises to refine both the left and the right part of the domain resulting in the mesh depicted in Figure 6.14(b). Thereby, the refinement on the right part is at least one level of refinement coarser than on the left part. This stronger concentration on the left part of the domain (leading to the poor error reduction which is even worse than uniform refinement) can be explained by the rather irregular behavior of the optimal state u at initial time. The quantitative estimator compensates this situation by its multiplicative structure: It couples the roughness of u multiplicatively with information about the adjoint state z which is rather smooth at this time.

Similar observations can be made by consideration of Figure 6.15 where a selection of spatial triangulations from the computations on dynamically changing meshes is depicted. Whereas the meshes obtained by the two different refinement indicators are comparatively similar at final time $t = 1$ (cf. the Figures 6.15(i) and 6.15(j)) they differ totally at initial time $t = 0$ (cf. the Figures 6.15(a) and 6.15(b)). At this time, the mesh obtained from using the quantitative error estimator is at most one level finer than the initial mesh. Like in the situation of one fixed mesh, the heuristic based refinement yields here a mesh for $t = 0$ which is strongly refined in the left half of the domain. The mesh is here even finer than the mesh at final time $t = 1$. Again, this stands in sharp contrast to the more efficient behavior of the proposed quantitative estimator.

We close this chapter with the conclusion that the construction of temporal and spatial discretizations for the efficient reduction of the error in terms of a quantity of interest is only practicable when based on systematic approaches to quantitative error estimation like the one derived in the first sections of this chapter.



(a) Quantitative estimator η_h^J: $t = 0$

(b) Heuristic estimator $\tilde{\eta}_h$: $t = 0$

(c) Quantitative estimator η_h^J: $t = 0.25$

(d) Heuristic estimator $\tilde{\eta}_h$: $t = 0.25$

(e) Quantitative estimator η_h^J: $t = 0.5$

(f) Heuristic estimator $\tilde{\eta}_h$: $t = 0.5$

(g) Quantitative estimator η_h^J: $t = 0.75$

(h) Heuristic estimator $\tilde{\eta}_h$: $t = 0.75$

(i) Quantitative estimator η_h^J: $t = 1$

(j) Heuristic estimator $\tilde{\eta}_h$: $t = 1$

Figure 6.15. Comparison of the local refinement obtained by the two considered estimators after six refinement cycles using dynamically changing meshes

7 Applications

In this chapter, we apply the a posteriori error analysis and the adaptive refinement techniques derived in Chapter 6 to two optimization problems motivated by concrete applications from engineering and chemistry.

As first application, we consider in Section 7.1 the optimal control of steel hardening induced by a laser beam. Thereby, the goal of the optimization is to adjust the laser intensity in such a way, that the thickness of the hardened part of the workpiece is close to a desired hardening profile. As second application, we investigate in Section 7.2 a model describing the propagation of laminar flames through a channel equipped with a cooled obstacle. The model is constructed using an Arrhenius law containing unknown parameters. In this setting, we consider the parameter identification problem of recovering one of these parameters from measurements in four spatial points at final time. The cost functional is given by a least-square formulation for penalizing the deviation from these measurements.

For the first configuration, the natural choice of measurement for error control is the cost functional itself. However, in the second example, the cost functional is just an artificial construction without any physical meaning. Thus, we aim there in estimating the discretization error directly with respect to the unknown parameter via a suitable choice of the quantity of interest.

7.1 Surface hardening of steel

We consider the optimal control of laser surface hardening of steel. In this process, a laser beam moves along the surface of a workpiece. The heating induced by the laser is accompanied by a phase transition, in which austenite, the high temperature phase in steel, is produced. Due to further phase transitions (which are not contained in the considered model) the desired hardening effect develops.

The goal is to control this hardening process such that a desired hardening profile is produced. Since in practical applications, the moving velocity of the laser beam is kept constant, the most important control parameter is the energy of the laser beam. Especially when there are large variations in the thickness of the workpiece or in regions near the boundaries of the workpiece, the proper adjustment of the laser energy is crucial to meet the given hardening profile.

7.1.1 Formulation of the problem

The configuration of the control problem to be investigated in this section is mainly taken from Fuhrmann and Hömberg [36] and Hömberg and Volkwein [46]. Accordingly to Leblond and Devaux [52], the formation of austenite is described therein by the initial value problem

$$\begin{aligned} \partial_t a &= \frac{1}{\tau(\theta)}\big[a_{\mathrm{eq}}(\theta) - a\big]_+ && \text{in } \varOmega \times I, \\ a &= 0 && \text{on } \varOmega \times \{\,0\,\}, \end{aligned} \tag{7.1}$$

where a is the volume fraction of austenite, a_{eq} is the equilibrium volume fraction of austenite, and τ is a time constant. Both a_{eq} and τ depend on the temperature θ. The brackets

$$[v]_+ \coloneqq \frac{v + |v|}{2}$$

denote the non-negative part of v.

The temperature distribution θ in the workpiece is described by the following heat equation:

$$\begin{aligned} \rho c_{\mathrm{p}} \partial_t \theta - \varepsilon \Delta \theta &= -\rho L \partial_t a + q \varLambda && \text{in } \varOmega \times I, \\ \partial_n \theta &= 0 && \text{on } \partial\varOmega \times I, \\ \theta &= \theta_0 && \text{on } \varOmega \times \{\,0\,\}. \end{aligned} \tag{7.2}$$

Here, the density ρ, the heat capacity c_{p}, the heat conductivity ε, and the latent heat L are assumed to be positive constants. The term $q(t)\varLambda(x,t)$ describes the volumetric heat source due to laser radiation, where q acts as time-dependent control variable. Thus, the optimal control is searched for in $Q = L^2(I, \mathbb{R})$.

For theoretical as well as computational reasons, the term $\big[a_{\mathrm{eq}}(\theta) - a\big]_+$ from (7.1) is regularized as

$$\big[a_{\mathrm{eq}}(\theta) - a\big]_+ \approx (a_{\mathrm{eq}}(\theta) - a)\mathcal{H}_\delta(a_{\mathrm{eq}}(\theta) - a), \tag{7.3}$$

where $\mathcal{H}_\delta$ is a monotone regularization of the Heaviside function, given for instance by

$$\mathcal{H}_\delta(s) \coloneqq \begin{cases} 1 & \text{for } s \geq \delta, \\ 10\Big(\dfrac{s}{\delta}\Big)^6 - 24\Big(\dfrac{s}{\delta}\Big)^5 + 15\Big(\dfrac{s}{\delta}\Big)^4 & \text{for } \delta > s \geq 0, \\ 0 & \text{for } s < 0 \end{cases}$$

with a parameter $\delta > 0$.

Thus, as governing state equation, we consider the combination of (7.1) and (7.2) together with the approximation (7.3):

$$\begin{aligned} \partial_t a &= \frac{1}{\tau(\theta)}(a_{\mathrm{eq}}(\theta) - a)\mathcal{H}_\delta(a_{\mathrm{eq}}(\theta) - a) && \text{in } \varOmega \times I, \\ \rho c_{\mathrm{p}} \partial_t \theta - \varepsilon \Delta \theta &= -\rho L \partial_t a + q \varLambda && \text{in } \varOmega \times I, \\ \partial_n \theta &= 0 && \text{on } \partial\varOmega \times I, \\ a &= 0 && \text{on } \varOmega \times \{\,0\,\}, \\ \theta &= \theta_0 && \text{on } \varOmega \times \{\,0\,\}. \end{aligned} \tag{7.4}$$

In [46], it is proven that under some smoothness conditions for the data a_{eq} and τ, and under the assumptions $\theta_0 \in H^1(\Omega)$, $\Lambda \in L^\infty(I, L^\infty(\Omega))$, $q \in L^2(I, \mathbb{R})$ the state equation (7.4) admits a unique solution $u := (a, \theta) \in W^{1,\infty}(I, L^\infty(\Omega)) \times X$. Here, because of the Neumann boundary conditions for θ, the space X is defined using $V = H^1(\Omega)$ and $H = L^2(\Omega)$.

As cost functional to be minimized, we choose

$$J(q, u) = \frac{\beta}{2} \int_I \|a(t) - \hat{a}(t)\|^2_{L^2(\Omega)} \, dt + \frac{\alpha}{2} \int_I q(t)^2 \, dt,$$

where $\hat{a}$ is a given desired volume fraction of austenite. In [46], the authors considered observation located only at final time T. Since we treat already an example with observation at final time in the following section, we choose the objective functional distributed over time and space. The numerical results (cf. the Figures 7.2 and 7.3 in the following subsection) confirm that also this choice leads to the desired hardening profile especially at final time T.

For the computations, we choose the physical parameters for the heat equation accordingly to [46] as

$$\rho c_{\mathrm{p}} = 1.17, \qquad \varepsilon = 0.153, \qquad \text{and} \qquad \rho L = 150.$$

The equilibrium volume fraction a_{eq} and the time constant τ are constructed by cubic spline interpolation of the values from Table 7.1. The resulting spline approximations are depicted in Figure 7.1.

Table 7.1. Data for a_{eq} and τ

θ	$a_{\mathrm{eq}}(\theta)$	$\tau(\theta)$
730	0	1
830	0.91	0.2
840	1	0.18
900	1	0.05

The parameter δ in the definition of the regularized Heaviside function is chosen as $\delta = 0.15$, and the initial condition for the temperature is set to $\theta_0 = 20$. The laser source Λ is modeled by

$$\Lambda(x, t) = \frac{4\kappa A}{\pi D^2} \exp\Big(-\frac{2(x_1 - vt)^2}{D^2}\Big) \exp(\kappa x_2), \quad x = (x_1, x_2)^T,$$

where the values of the parameters are taken from [46] as $D = 0.47$, $\kappa = 60$, $A = 0.3$, and $v = 1.15$.

For the numerical computations, we choose the domain Ω to be $(0, 5) \times (-1, 0)$ and determine the final time T such that the laser, which moves from $(0, 0)$ to $(5, 0)$, reaches the boundary at $(5, 0)$ at time T. Thus, we set $T = {}^5\!/_v \approx 4.34782$. The desired volume fraction $\hat{a}$ is chosen as

$$\hat{a}(x) := \begin{cases} 1 & \text{for } 0 \geq x_1 \geq -\frac{1}{8} \\ 0 & \text{for } -\frac{1}{8} > x_1 \geq -1 \end{cases}, \quad x = (x_1, x_2)^T,$$

and for the parameters α and β from the definition of the objective functional J we take $\alpha = 10^{-4}$ and $\beta = 3500$.

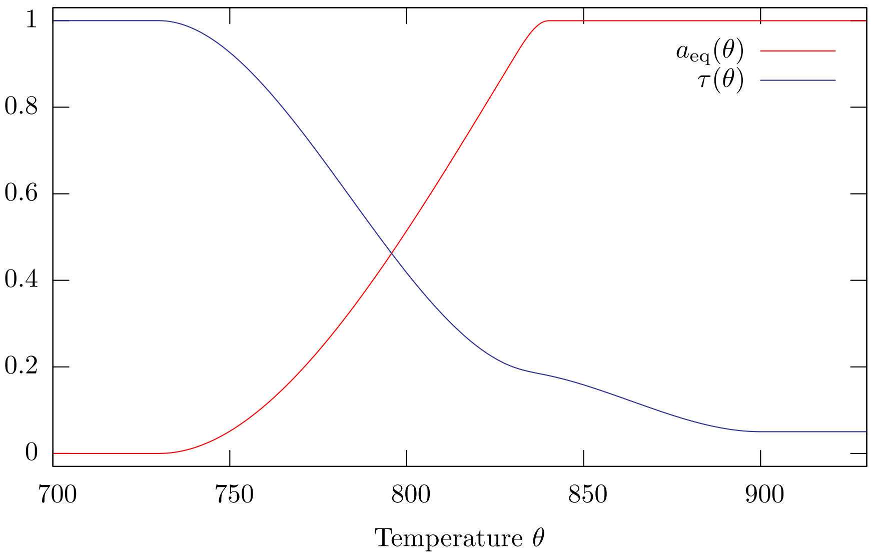


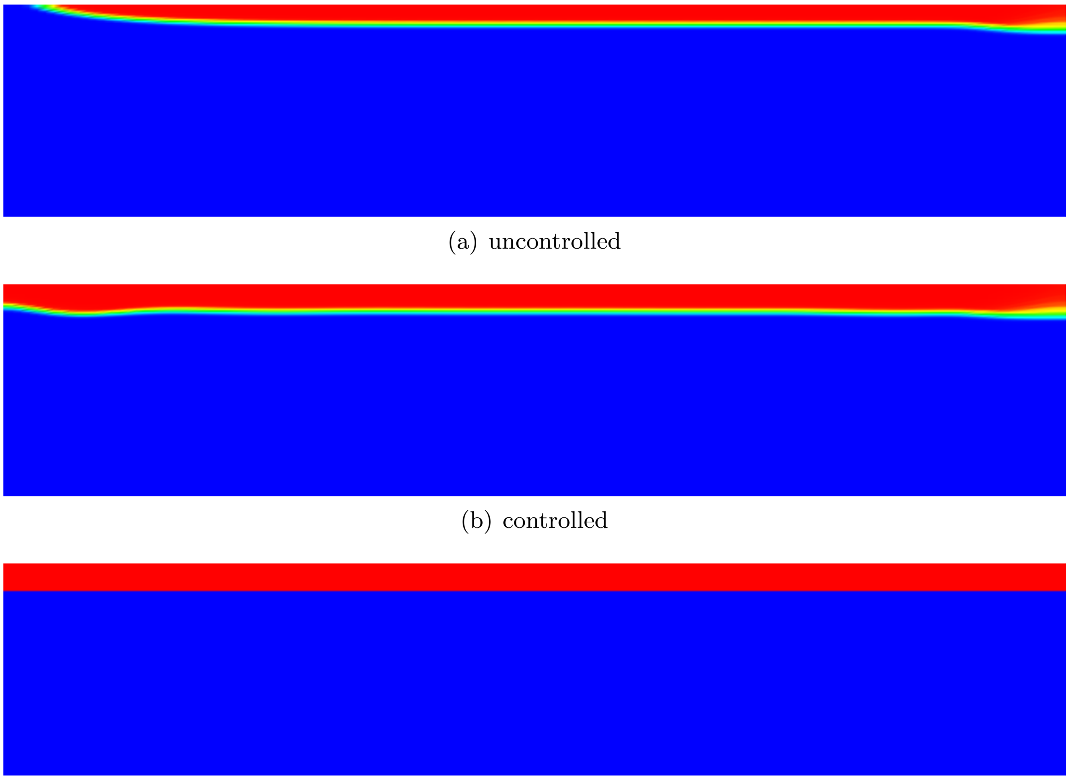
Figure 7.1. Spline interpolation of the data for a_{eq} and τ

7.1.2 Numerical results

For discretizing the state space, we employ the cG(1)dG(0) discretization, that is, the discrete state is piecewise bilinear in space and piecewise constant in time. Since the control space is given by $Q = L^2(I, \mathbb{R})$, we have to discretize the controls only in time. Correspondingly to the state space, we choose a dG(0) discretization based on a possibly coarser step size than the step size used for discretizing the state space; cf. the discussion in Example 3.1.

At first, we investigate the qualitative behavior of the optimization algorithm. Figure 7.2 presents the distribution of austenite at final time T before (a) and after (b) the optimization on a fine discretization of the state space. To compare with, the desired state is depicted in Figure 7.2(c). Figure 7.3 proves the gain of optimization by showing the pointwise error between the desired state and the uncontrolled (a) and controlled (b) volume fraction of austenite at final time.

As next step, we verify the properties of the error estimator with respect to the temporal discretization error of the state variable. That is, we consider the error $e_k^J := J(q_{hd}, u_{hd}) - J(q_\sigma, u_\sigma)$ and the corresponding error estimator η_k^J. Thereby, an approximation of the values for $J(q_{hd}, u_{hd})$ is computed by extrapolation of the values obtained on fine time discretizations. The development of this error for uniform refinement of the temporal discretization and for local refinement based on the information obtained from η_k^J is depicted in Figure 7.4. We observe almost no difference between the two types of refinement and consequently there is no gain due to the local refinement at all. This can be explained by the global structure of the problem: Both the functional and the laser beam act on the whole time interval. Nevertheless, just the knowledge of the size of the temporal discretization error provided by the error estimation leads to remarkable savings of computational costs when using this information within a coupled refinement of all involved discretizations.

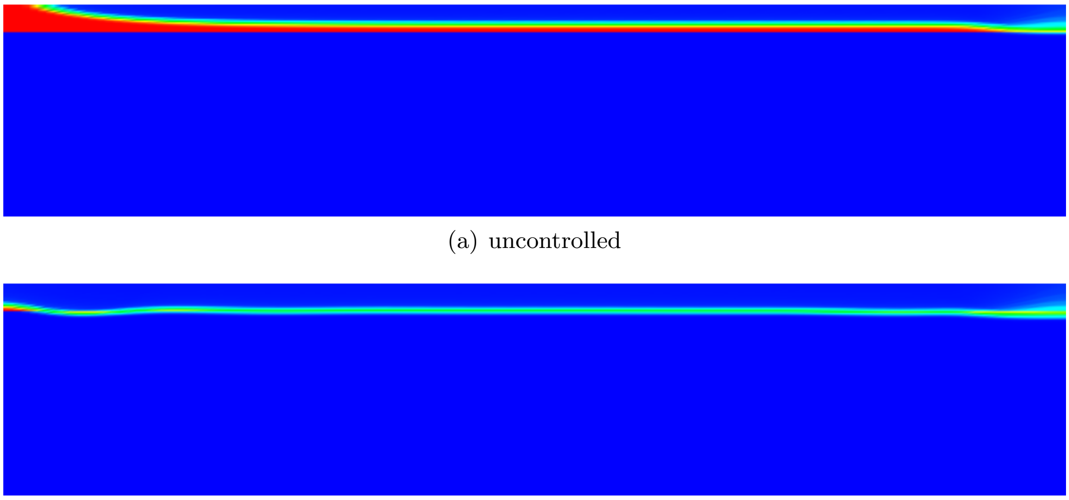


(a) uncontrolled

(b) controlled

(c) desired

Figure 7.2. Distribution of austenite at final time T



(a) uncontrolled

(b) controlled

Figure 7.3. Discrepancy between the distribution of austenite and the desired state at final time T

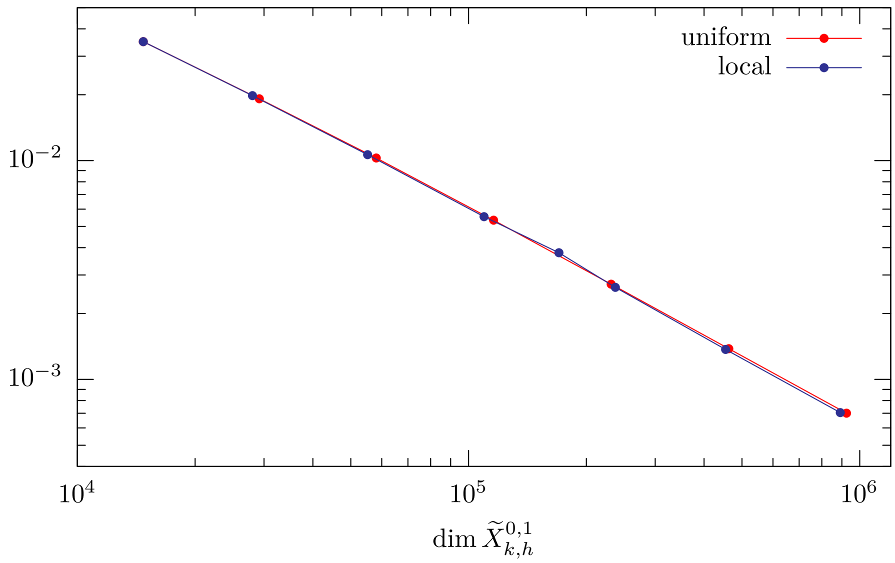


Figure 7.4. Comparison of the relative error $|e_k^J|/J$ for uniform and local refinement of the time steps

In Figure 7.5, we present a comparison of different refinement strategies for the spatial discretization. We depict the development of the error $e_h^J := J(q_{kd}, u_{kd}) - J(q_\sigma, u_\sigma)$ caused by the spatial discretization of the state space. For testing the temporal error estimator, an approximation of the value for $J(q_{kd}, u_{kd})$ is obtained by extrapolation. Thereby, we consider the following three types of refinement:

- Uniform refinement
- Local refinement based on the error indicator η_h^J with one fixed mesh for all time steps
- Local refinement based on η_h^J but allowing separate spatial meshes for each time interval by using dynamically changing meshes

We observe that by the usage of local refinement the number of grid points can be reduced from $N = 16{,}641$ to $N = 5{,}271$. Moreover, if we allow dynamically changing meshes, we only need $N_{\max} = 3{,}873$ grid points. The total number of degrees of freedom in the space discretization ($\dim \widetilde{X}_{k,h}^{0,1}$) is reduced even by a factor of 5.7 when employing local refinement on dynamic meshes.

In the Figures 7.6 and 7.7, a selection from the sequence of locally refined meshes is given. Thereby, we detect a strong refinement at the position where the laser currently acts and at the region around the transition from hardened to not hardened steel. In this region, the optimal distribution of austenite as well as the desired hardening profile exhibits spatial discontinuities.

We now couple the temporal and spatial estimators by the equilibration strategy described in Section 6.5. Since we do not benefit from local refinement in time, we allow only uniform refinement of the time steps. However, in space we allow the adaptation procedure to use

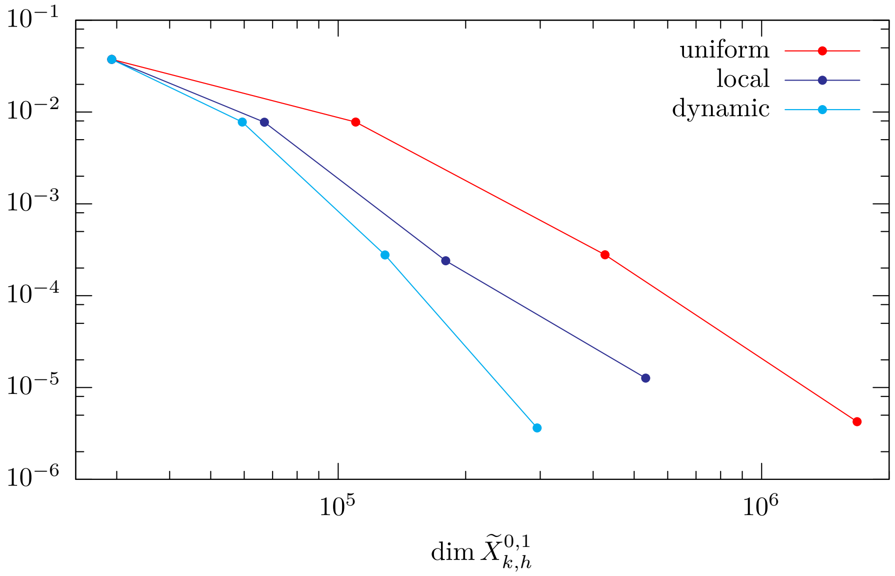


Figure 7.5. Comparison of the relative error $|e_h^J|/J$ for uniform and local refinement of the triangulation using dynamically changing meshes

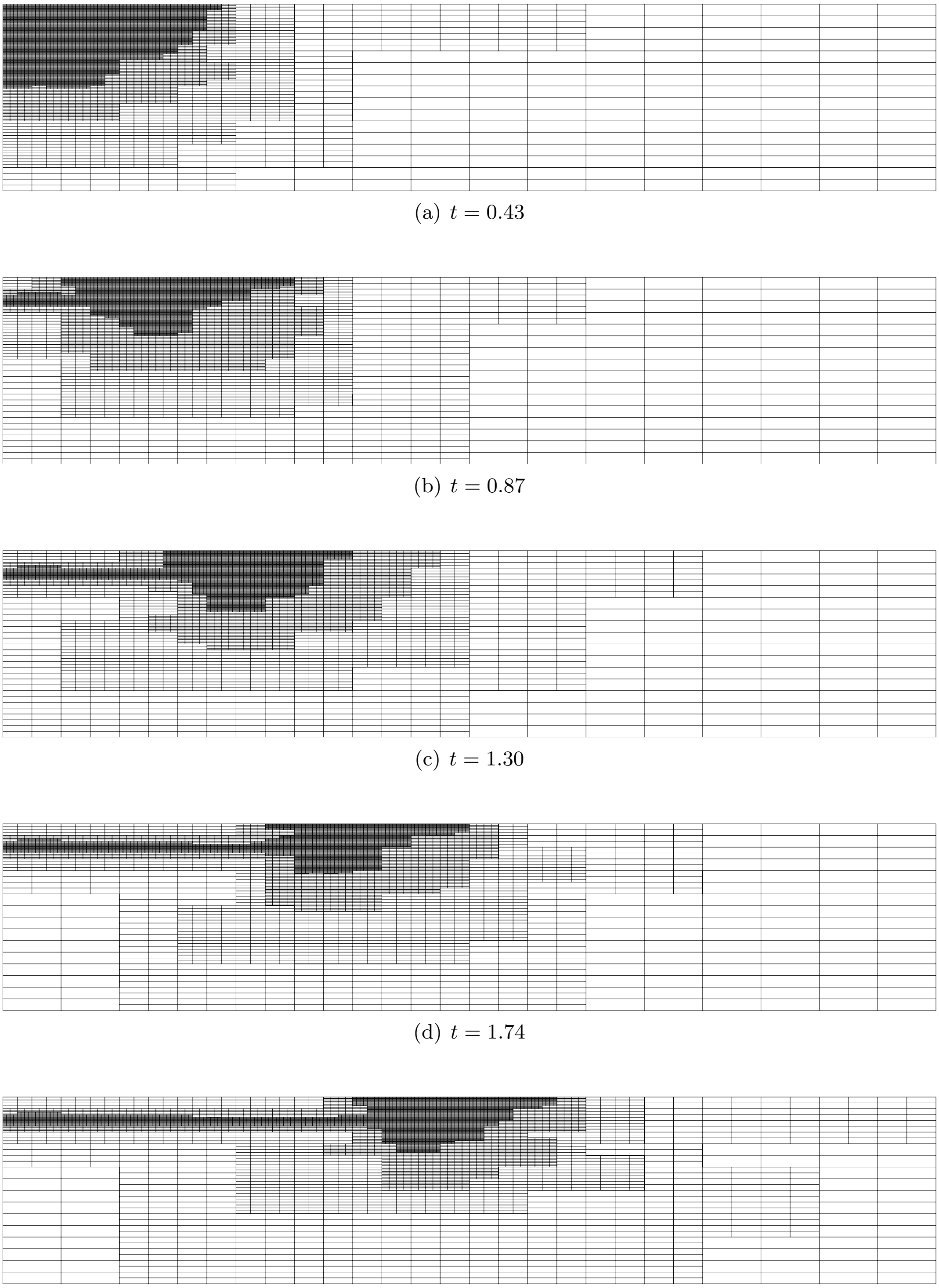
dynamically changing meshes. Results of this computation are given in Table 7.2. Therein, we observe that the contribution from the spatial discretization error to the overall error is much smaller than the contribution from the temporal discretization error. Consequently, the equilibration procedure decides for example to keep the spatial meshes fixed while increasing the number of time steps from 200 over 400 to 800 time steps.

Table 7.2. Local refinement on dynamic meshes with equilibration

N_{tot}	$N_{\max}$	M	η_h^J/J	η_k^J/J	$\eta_h^J/J + \eta_k^J/J$	e_{kh}^J/J	I_{eff}
14739	289	50	$-7.4 \cdot 10^{-03}$	$2.3 \cdot 10^{-02}$	$1.622 \cdot 10^{-02}$	$-4.916 \cdot 10^{-03}$	-0.30
59325	675	100	$-2.8 \cdot 10^{-03}$	$1.3 \cdot 10^{-02}$	$1.049 \cdot 10^{-02}$	$7.828 \cdot 10^{-03}$	0.74
257867	1659	200	$-3.9 \cdot 10^{-04}$	$6.3 \cdot 10^{-03}$	$6.040 \cdot 10^{-03}$	$7.445 \cdot 10^{-03}$	1.23
515115	1659	400	$-3.9 \cdot 10^{-04}$	$3.2 \cdot 10^{-03}$	$2.827 \cdot 10^{-03}$	$3.454 \cdot 10^{-03}$	1.22
1029611	1659	800	$-4.3 \cdot 10^{-04}$	$1.6 \cdot 10^{-03}$	$1.193 \cdot 10^{-03}$	$1.424 \cdot 10^{-03}$	1.19
4721397	3911	1600	$9.8 \cdot 10^{-06}$	$7.8 \cdot 10^{-04}$	$8.143 \cdot 10^{-04}$	$9.375 \cdot 10^{-04}$	1.15

This implies that uniform refinement of the time and space discretizations without the knowledge of the size of the different error contributions can not be competitive. For the efficient equilibration—and thus the efficient reduction of the error—estimations of the size of each involved discretization errors are essential. Furthermore, the table demonstrates that the estimator $\eta_h^J/J + \eta_k^J/J$ is in very good agreement with the relative error $|e_{kh}^J|/J$ Thereby, the error e_{kh}^J is defined as $e_{kh}^J := J(q_d, u_d) - J(q_\sigma, u_\sigma)$ with an approximation of $J(q_d, u_d)$ obtained by extrapolation. For normalizing the errors and the estimators we use J, which denotes an approximation of the exact value of the cost functional $J(q, u)$.

We pass on a graphical comparison of this results with the results of a computation using



(a) $t = 0.43$

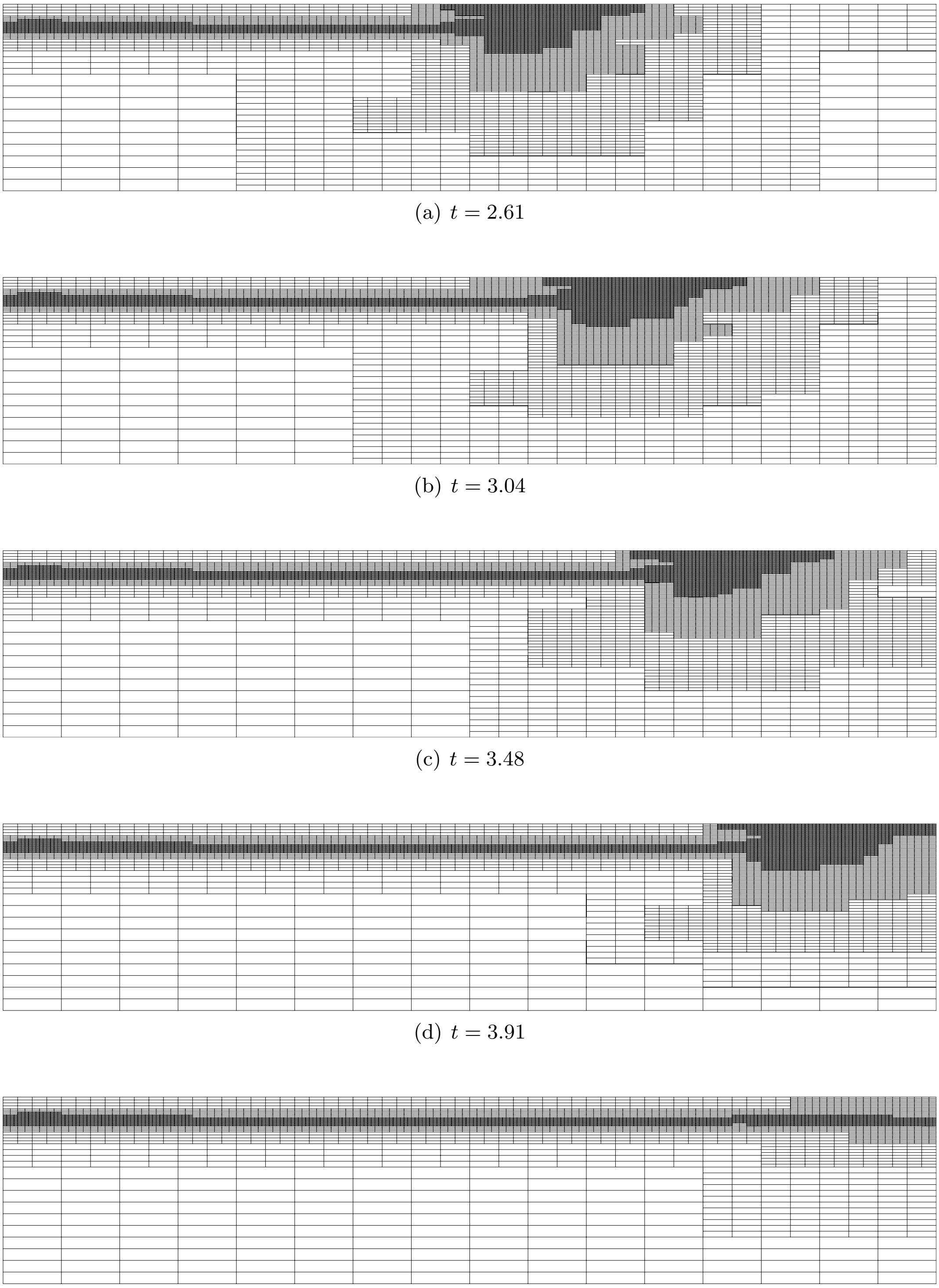
(b) $t = 0.87$

(c) $t = 1.30$

(d) $t = 1.74$

(e) $t = 2.17$

Figure 7.6. Locally refined meshes for $t \in \{ 0.43, 0.87, 1.30, 1.74, 2.17 \}$



(a) $t = 2.61$

(b) $t = 3.04$

(c) $t = 3.48$

(d) $t = 3.91$

(e) $t = 4.35$

Figure 7.7. Locally refined meshes for $t \in \{ 2.61, 3.04, 3.48, 3.91, 4.35 \}$

uniform refinement with and without equilibration. As mentioned, when not having the possibility to decide which type of discretization contributes the majority to the discretization error, one could not solve this problem efficiently at all. If equilibration is employed, the gain from the local refinement in space (cf. Figure 7.5) carries over directly to the space-time adaptive computation.

Next, we show in Figure 7.8 a series of optimal controls obtained by dG(0) approximations on refined discretizations of Q. We choose here for both the discretization of the state and of the control the same number of time intervals. That is, we use $M \in \{ 10, 20, 40, 80 \}$ time steps and consequently finite-dimensional subspaces Q_d with $\dim Q_d \in \{ 10, 20, 40, 80 \}$. This is motivated by the fact that for instance for linear-quadratic problems the discretization error caused by the discretization of the control space is zero if the control discretization and the discretization of the state variable fit together (cf. Remark 6.3). However, for the considered problem, the optimality condition can not be fulfilled pointwise, and thus the error due to the discretization of the control can at no time be neglected.

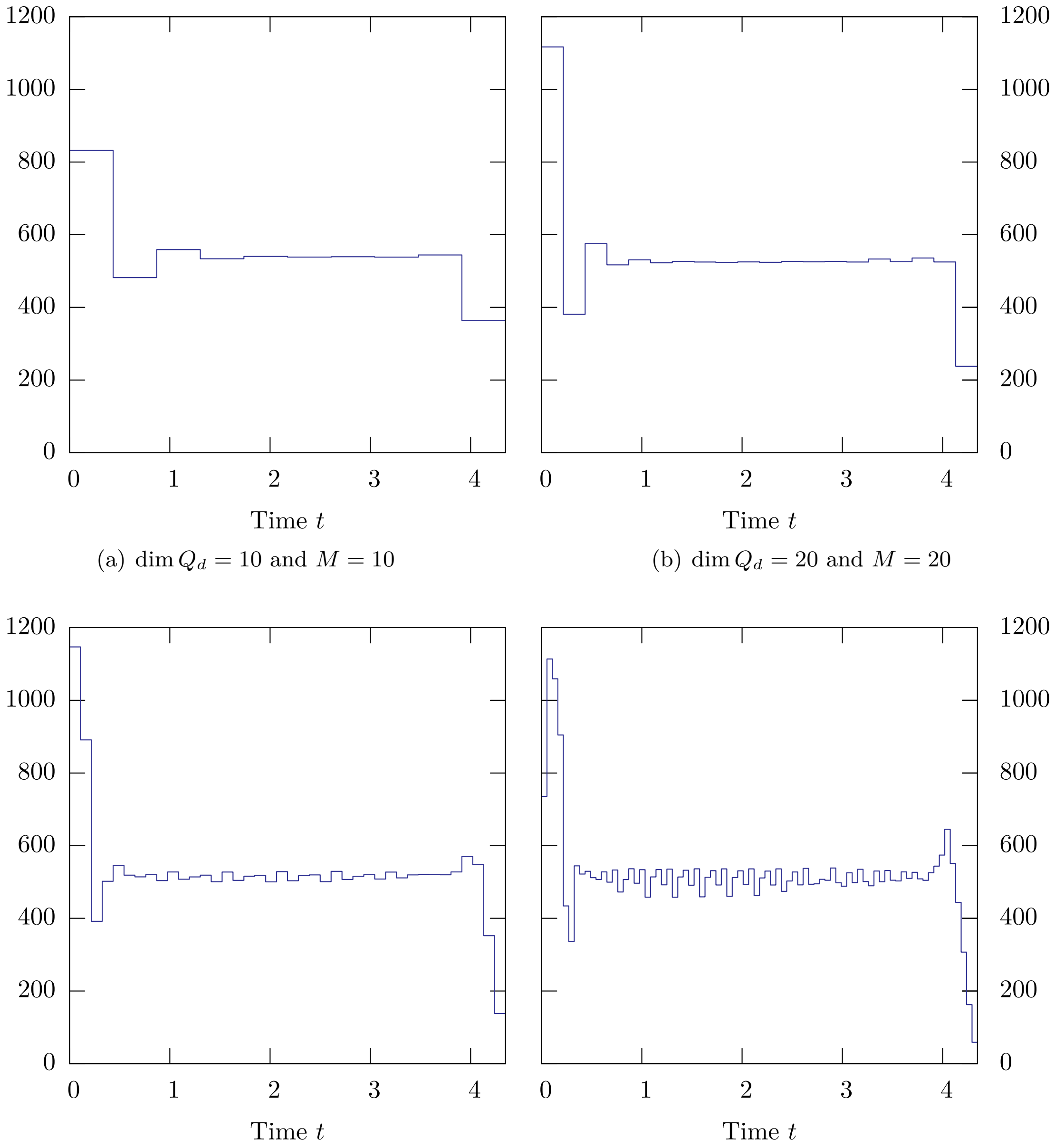
In this configuration, we observe that the computed optimal controls expose increasing instabilities when enlarging simultaneously the number of performed time steps and the dimension of the discrete control space. These instabilities may arise from the low regularity of the volume fraction a and the discontinuous desired state $\hat{a}$. As usual, the optimal control can be smoothened by enlarging the regularization parameter α. However, when doing so, one has to face that the focus of the optimization is moved more and more from minimizing the deviation of the state to the desired state to minimizing the norm of the control. Consequently, the quality of approximating the desired state by the optimization deteriorates. Hence, if not given by properties of the problem, the choice of the regularization parameter is quite delicate. A possible way out could be the usage of strategies for finding an optimal regularization parameter proposed in the field of inverse problems.

Another possibility, which we favor here, is “stabilization by discretization”. That is, we want to keep the discretization of Q relatively coarse to avoid the increasing instabilities in the optimal control. This is qualitatively justified by the results given in Figure 7.9. There, we depict also the optimal control for $\dim Q_d \in \{ 10, 20, 40, 80 \}$ but now for a fixed number of $M = 160$ time steps for the state. Here, we observe in contrast Figure 7.8, that the instabilities appear first when using a discrete control space with $\dim Q_d = 80$ degrees of freedom.

Table 7.3. Estimated errors due to time discretization of the state and the control space during refinement of the control discretization

M	$\dim Q_d$	η_k^J/J	η_d^J/J
160	10	$7.7\cdot 10^{-03}$	$-7.6\cdot 10^{-04}$
160	20	$8.4\cdot 10^{-03}$	$-4.4\cdot 10^{-05}$
160	40	$8.4\cdot 10^{-03}$	$-6.1\cdot 10^{-05}$
160	80	$8.3\cdot 10^{-03}$	$-2.4\cdot 10^{-05}$

However, the a priori selection of a suitable step size for the control discretization leading to accurate results and stable computations is virtually impossible. Hence, one can not do so without an estimation of the error due to the discretization of the control and the corresponding error indicators for the suitable adaptive refinement. For the concrete problem



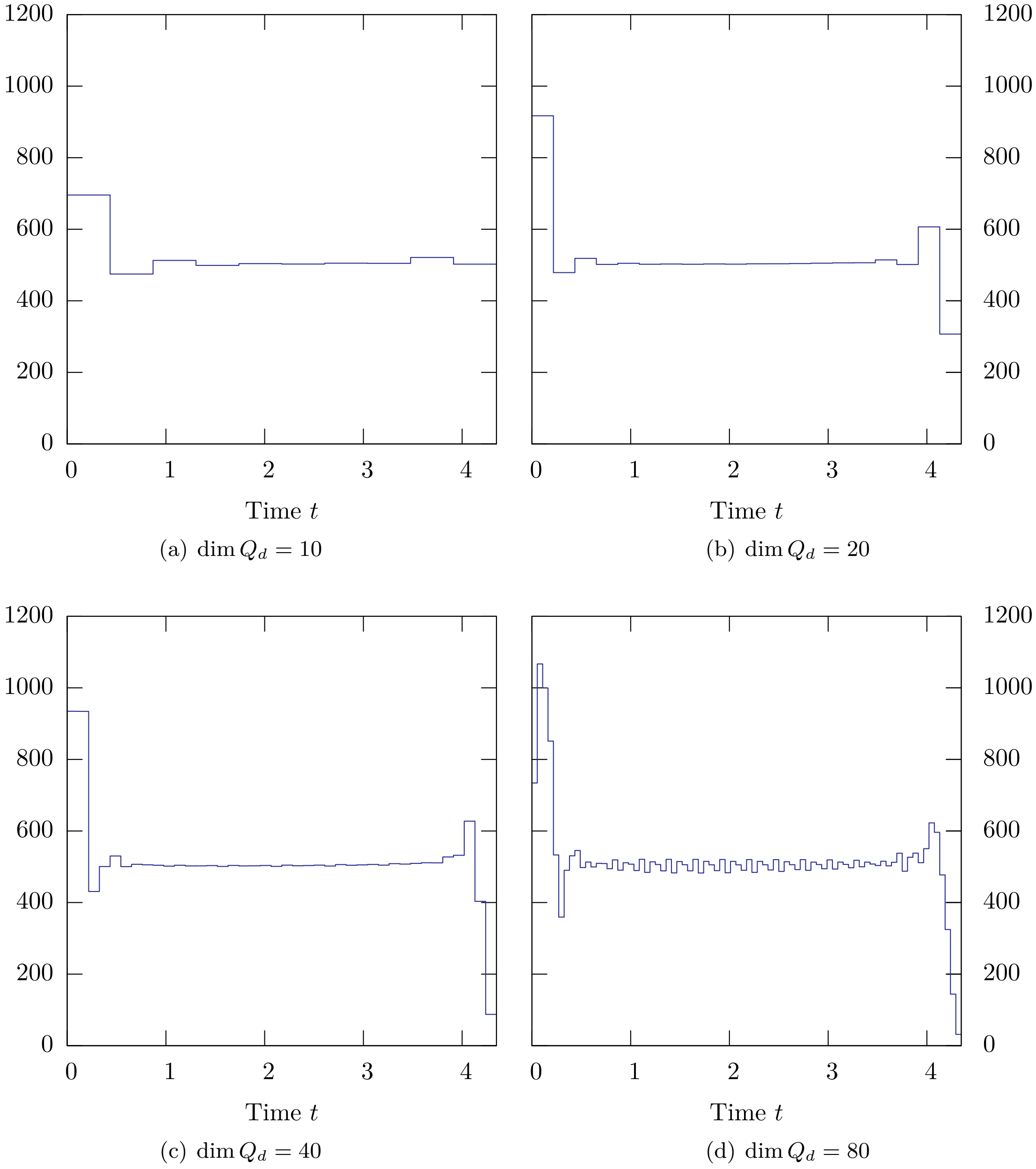
(a) $\dim Q_d = 10$ and $M = 10$

(b) $\dim Q_d = 20$ and $M = 20$

(c) $\dim Q_d = 40$ and $M = 40$

(d) $\dim Q_d = 80$ and $M = 80$

Figure 7.8. Optimal control q_σ for $N = 4{,}225$ spatial nodes and different levels of refinements of Q_d and different numbers of time steps



(a) $\dim Q_d = 10$

(b) $\dim Q_d = 20$

(c) $\dim Q_d = 40$

(d) $\dim Q_d = 80$

Figure 7.9. Optimal control q_σ for $N = 4{,}225$ spatial nodes and different levels of refinements of Q_d and a fixed number of $M = 160$ time steps

under consideration, Table 7.3 provides the information that the estimated error due to the time discretization of the state using $M = 160$ time steps is at least ten times larger than the estimated error due to the time discretization of the control on $\dim Q_d = 10$ time intervals. Hence, the discretization of the control space can be kept rather coarse here. This is confirmed by the results given in Table 7.4 where we compare the values of the estimators for the time discretization of the state space and the control space for $\dim Q_d = 10$ and $M \in \{ 160, 320, 640, 1280 \}$. We observe that the error induced by the state discretization dominates the error induced by the control discretization even for $M = 1280$ time steps. That is, also for this choice of state discretization, a discretization of the control space with $\dim Q_d = 10$ is sufficient. This does not only save computational costs but it also enhances the stability properties of the problem.

Table 7.4. Estimated errors due to time discretization of the state and the control space during refinement of the state discretization

M	$\dim Q_d$	η_k^J/J	η_d^J/J
160	10	$7.6 \cdot 10^{-03}$	$-7.5 \cdot 10^{-04}$
320	10	$4.0 \cdot 10^{-03}$	$-5.8 \cdot 10^{-04}$
640	10	$2.0 \cdot 10^{-03}$	$-5.1 \cdot 10^{-04}$
1280	10	$1.0 \cdot 10^{-03}$	$-4.8 \cdot 10^{-04}$

Remark 7.1. In general, we have to confess that it is usually not possible to implement infinite-dimensional controls in practice. Actually, in the optimization of real applications only a finite number of controls should be considered. However, also in this case, the estimation of the error due to practical constraints on the control space can be of interest. Based on these estimates, one can for example advise to invest into a finer resolved control to reduce the gap between the continuous optimal solution and the discrete one.

7.2 Propagation of laminar flames

In this section, we consider a parameter estimation problem arising from chemistry. We aim at the identification of an unknown parameter in a reaction mechanism governed by an Arrhenius law. This formulation is employed to model the propagation of laminar flames through a channel. The channel is narrowed by two heat absorbing obstacles influencing the traveling of the flame.

The identification of the unknown parameter is done employing measurements of the solution components at four spatial points at final time. Using these values, the cost functional is constructed by means of a least-squares formalism.

7.2.1 Formulation of the problem

The governing equation for the considered problem is taken from an example given in Lang [50]. It describes the major part of gaseous combustion under the low Mach number hypothesis. In this approach, the dependency of the fluid density on the pressure is eliminated while the temperature dependence remains. If additionally the dependence on the temperature is neglected, the motion of the fluid becomes independent on the temperature and the species concentration. Hence, one can solve the temperature and the species equation alone specifying any solenoidal velocity field v. In particular, $v = 0$ is an interesting case.

Introducing the dimensionless temperature θ, denoting by Y the species concentration, and assuming constant diffusion coefficients yields the system of equations

$$\begin{aligned}
\partial_t\theta - \Delta\theta &= \omega(Y,\theta) && \text{in } \Omega \times I,\\
\partial_t Y - \frac{1}{\mathrm{Le}}\Delta Y &= -\omega(Y,\theta) && \text{in } \Omega \times I,\\
\theta &= \theta_0 && \text{on } \Omega \times \{\,0\,\},\\
Y &= Y_0 && \text{on } \Omega \times \{\,0\,\},
\end{aligned} \tag{7.5}$$

where the Lewis number Le is the ratio of diffusivity of heat and diffusivity of mass. We use a simple one-species reaction mechanism governed by an Arrhenius law given by

$$\omega(Y,\theta) = \frac{\beta^2}{2\mathrm{Le}} Y \mathrm{e}^{\frac{\beta(\theta-1)}{1+\alpha(\theta-1)}}, \tag{7.6}$$

in which an approximation for large activation energy has been employed.

Here, we consider a freely propagating laminar flame described by (7.5) and its response to a heat absorbing obstacle, a set of cooled parallel rods with rectangular cross section (cf. Figure 7.10). The computational domain has width $H = 16$ and length $L = 60$. The obstacle covers half of the width and has length $L/4$. The boundary conditions are chosen as

$$\begin{aligned}
\theta &= 1 \quad \text{on } \Gamma_{\mathrm{D}} \times I, & \partial_n\theta &= 0 \quad \text{on } \Gamma_{\mathrm{N}} \times I, & \partial_n\theta &= -\kappa\theta \quad \text{on } \Gamma_{\mathrm{R}} \times I,\\
Y &= 0 \quad \text{on } \Gamma_{\mathrm{D}} \times I, & \partial_n Y &= 0 \quad \text{on } \Gamma_{\mathrm{N}} \times I, & \partial_n Y &= 0 \quad \text{on } \Gamma_{\mathrm{R}} \times I,
\end{aligned}$$

where the heat absorption is modeled by boundary conditions of Robin type on Γ_{R}.

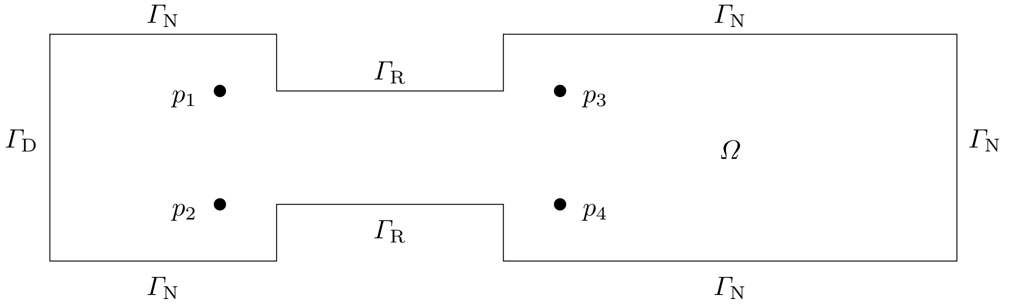


Figure 7.10. Computational domain Ω and measurement points p_i

The initial condition is the analytical solution of a one-dimensional right-traveling flame in the limit $\beta \to \infty$ located left of the obstacle:

$$\theta_0(x) = \begin{cases} 1 & \text{for } x_1 \le \tilde{x}_1 \\ \mathrm{e}^{\tilde{x}_1 - x_1} & \text{for } x_1 > \tilde{x}_1 \end{cases},$$

$$Y_0(x) = \begin{cases} 0 & \text{for } x_1 \le \tilde{x}_1 \\ 1 - \mathrm{e}^{\mathrm{Le}(\tilde{x}_1 - x_1)} & \text{for } x_1 > \tilde{x}_1 \end{cases}.$$

For the computations, the occurring parameters are set as in [50] to

$$\mathrm{Le} = 1, \qquad \beta = 10, \qquad \kappa = 0.1, \qquad \tilde{x}_1 = 9,$$

whereas the temperature ratio α, which determines the gas expansion in non-constant density flows, is the objective of the parameter estimation.

To use the same notation as in the theoretical parts of this work, we define the pair of solution components $u := (\theta, Y) \in \tilde{u} + X^2$ and denote the parameter α to be estimated by $q \in Q := \mathbb{R}$. For the definition of the state space X, we use here the spaces V and H given as

$$V = \left\{ v \in H^1(\Omega) \;\middle|\; v\big|_{\Gamma_\mathrm{D}} = 0 \right\} \qquad \text{and} \qquad H = L^2(\Omega).$$

The function $\tilde{u}$ is defined to fulfill the prescribed Dirichlet data as $\tilde{u}\big|_{\Gamma_\mathrm{D}} = (1, 0)$.

The unknown parameter α is estimated here using information from pointwise measurements of θ and Y at four points $p_i \in \Omega$, $i = 1, 2, 3, 4$, at final time $T = 60$. This parameter identification problem can be formulated by means of a cost functional of least-squares type, that is

$$J(q, u) = \frac{1}{2} \sum_{i=1}^{4} \bigl(\theta(p_i, T) - \hat{\theta}_i\bigr)^2 + \frac{1}{2} \sum_{i=1}^{4} \bigl(Y(p_i, T) - \hat{Y}_i\bigr)^2.$$

The values of the artificial measurements $\hat{\theta}_i$ and $\hat{Y}_i$, $i = 1, 2, 3, 4$, are obtained from a reference solution computed on fine space and time discretizations.

The consideration of point measurements does not fulfill the assumption on the cost functional in (2.4), since the point evaluation is not bounded as a functional on H. Therefore, the point functionals here have to be understood as regularized functionals defined on H. An a priori analysis of parameter estimation problems governed by elliptic equations and using such types of point functionals can be found in Rannacher and Vexler [67].

For the considered type of parameter estimation problems, one is usually not interested in reducing the discretization error measured in terms of the cost functional J. The focus is rather on the error in the parameter q itself. Hence, we define the quantity of interest E as

$$E(q, u) = q,$$

and apply the techniques presented in Section 6.2 for estimating the discretization error with respect to E. Since the control space in this application is given by $Q = \mathbb{R}$, it is not necessary to discretize Q. Thus, there is no discretization error due to the control discretization and the a posteriori error estimator η^E consists only of η^E_k and η^E_h.

7.2.2 Numerical results

For the computations, the state is discretized using the cG(1)dG(0) approach, that is by using piecewise constant polynomials in time and piecewise bilinear polynomials in space. We define the temporal and spatial discretization errors e^E_k and e^E_h as

$$e^E_k := E(q_h, u_h) - E(q_{kh}, u_{kh}) \qquad \text{and} \qquad e^E_h := E(q_k, u_k) - E(q_{kh}, u_{kh}).$$

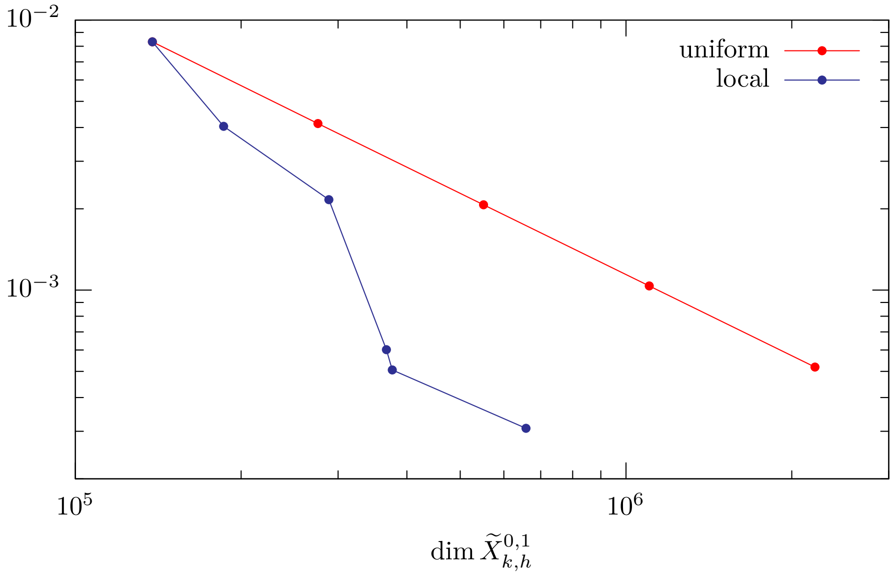


Figure 7.11. Comparison of the error $|e_k^E|$ for uniform and local refinement of the time steps

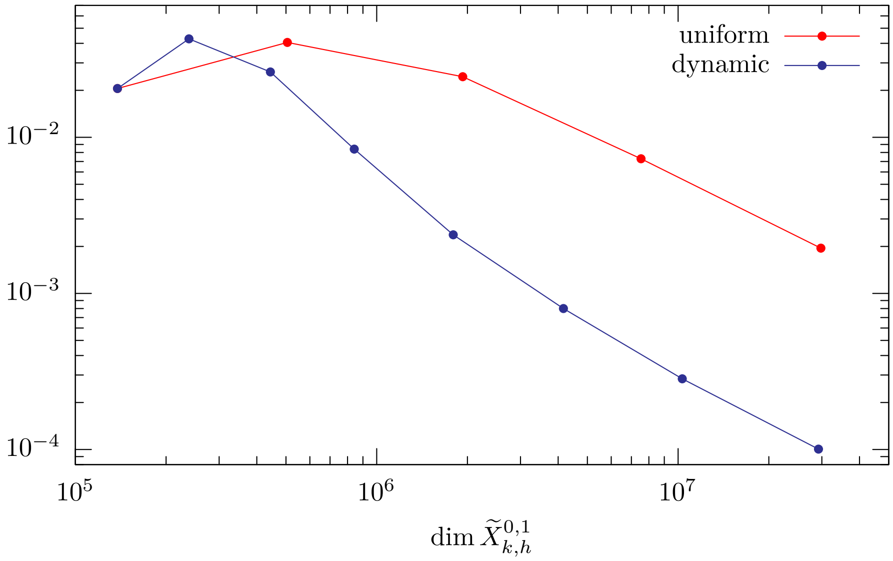


Figure 7.12. Comparison of the error $|e_h^E|$ for uniform and local refinement of the triangulation using dynamically changing meshes

The values of $E(q_h, u_h)$ and $E(q_k, u_k)$ are extrapolated from computations on a sequence of fine time and space discretizations, respectively. Since we have $E(q, u) = q = \alpha \approx 0.8$, there is no difference between the consideration of relative or absolute errors.

At first, we consider the case of refining only the time or only the space discretization. Thereby, we compare the behavior of the temporal discretization error for uniform and local refinement of the time grid and the behavior of the spatial discretization error for uniform and local refinement of the spatial triangulations using dynamically changing meshes. The results of these comparisons are depicted in the Figures 7.11 and 7.12. To reach for example an error $|e_k^E| \approx 5 \cdot 10^{-4}$, we gain a reduction of the number of time steps from $M = 8{,}192$ for uniform refinement to $M = 1{,}398$ for refinement due to the error indicator η_k^E. Correspondingly, to reach the error $|e_h^E| \approx 2 \cdot 10^{-3}$, we need at most $N_{\max} = 5{,}005$ grid points of the spatial discretizations when using dynamic meshes instead of $N = N_{\max} = 58{,}049$ grid points when using a fixed mesh with uniform refinement.

Table 7.5. Local refinement on a fixed mesh with equilibration

N	M	η_h^E	η_k^E	$\eta_h^E + \eta_k^E$	e^E	I_{eff}
269	512	$4.3 \cdot 10^{-02}$	$-8.4 \cdot 10^{-03}$	$3.551 \cdot 10^{-02}$	$-2.889 \cdot 10^{-02}$	-0.81
635	512	$5.5 \cdot 10^{-03}$	$-9.1 \cdot 10^{-03}$	$-3.533 \cdot 10^{-03}$	$-4.851 \cdot 10^{-02}$	13.72
1847	722	$-1.5 \cdot 10^{-02}$	$-3.6 \cdot 10^{-03}$	$-1.889 \cdot 10^{-02}$	$-3.024 \cdot 10^{-02}$	1.60
5549	1048	$-6.5 \cdot 10^{-03}$	$-2.5 \cdot 10^{-03}$	$-9.074 \cdot 10^{-03}$	$-1.097 \cdot 10^{-02}$	1.20
14419	1088	$-2.4 \cdot 10^{-03}$	$-2.5 \cdot 10^{-03}$	$-5.064 \cdot 10^{-03}$	$-5.571 \cdot 10^{-03}$	1.10
43343	1102	$-8.5 \cdot 10^{-04}$	$-2.5 \cdot 10^{-03}$	$-3.453 \cdot 10^{-03}$	$-3.693 \cdot 10^{-03}$	1.06

The next computations are done using simultaneous refinement of the space and time discretizations. Thereby, the refinements are coupled by the equilibration strategy introduced in Section 6.4. The Tables 7.5 and 7.6 demonstrate the effectivity of the error estimator $\eta_h^E + \eta_k^E$ on locally refined discretizations using fixed and dynamically changing spatial triangulations.

Table 7.6. Local refinement on dynamic meshes with equilibration

N_{tot}	$N_{\max}$	M	η_h^E	η_k^E	$\eta_h^E + \eta_k^E$	e^E	I_{eff}
137997	269	512	$4.3 \cdot 10^{-02}$	$-8.4 \cdot 10^{-03}$	$3.551 \cdot 10^{-02}$	$-2.889 \cdot 10^{-02}$	-0.81
238187	663	512	$3.5 \cdot 10^{-03}$	$-8.6 \cdot 10^{-03}$	$-5.192 \cdot 10^{-03}$	$-5.109 \cdot 10^{-02}$	9.84
633941	1677	724	$-1.6 \cdot 10^{-02}$	$-3.5 \cdot 10^{-03}$	$-2.015 \cdot 10^{-02}$	$-3.227 \cdot 10^{-02}$	1.60
1741185	2909	1048	$-7.3 \cdot 10^{-03}$	$-2.5 \cdot 10^{-03}$	$-9.869 \cdot 10^{-03}$	$-1.214 \cdot 10^{-02}$	1.23
3875029	4785	1098	$-2.2 \cdot 10^{-03}$	$-2.5 \cdot 10^{-03}$	$-4.792 \cdot 10^{-03}$	$-5.432 \cdot 10^{-03}$	1.13
9382027	10587	1140	$-7.9 \cdot 10^{-04}$	$-2.5 \cdot 10^{-03}$	$-3.301 \cdot 10^{-03}$	$-3.588 \cdot 10^{-03}$	1.08
23702227	25571	1160	$-2.8 \cdot 10^{-04}$	$-2.4 \cdot 10^{-03}$	$-2.756 \cdot 10^{-03}$	$-2.944 \cdot 10^{-03}$	1.06

In Figure 7.13, we compare uniform refinement of the space and time discretizations with local refinement of both discretizations on a fixed spatial triangulation and on dynamically changing triangulations. We gain an remarkable reduction of the required degrees of freedom for reaching a given tolerance. To meet for instance an error of $|e^E| \approx 10^{-2}$, the uniform refinement requires in total 15,056,225 degrees of freedom, the local refinement needs 5,820,901 degrees of freedom, and the dynamical refinement necessitates only 1,741,185 degrees of freedom. Thus, we gain a reduction of about 8.6.

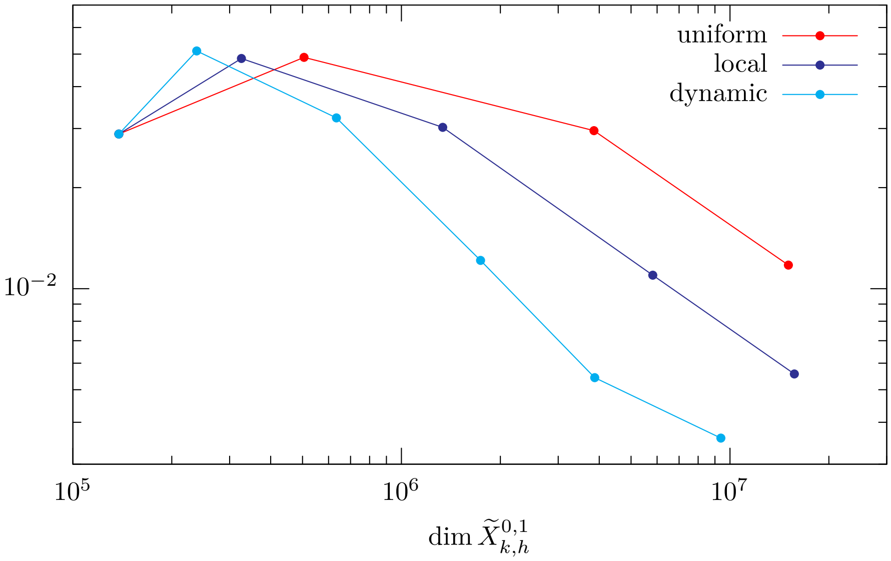


Figure 7.13. Comparison of the error $|e^E|$ for different refinement strategies

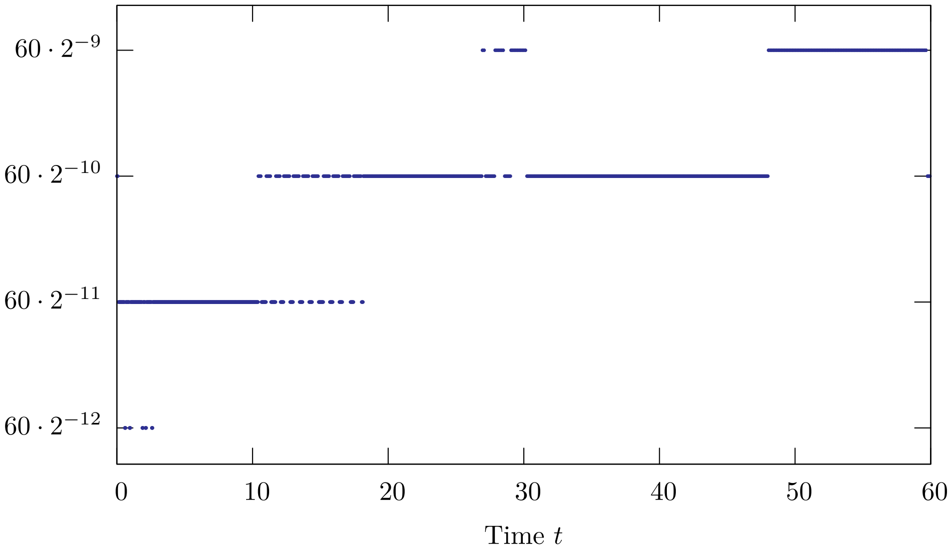


Figure 7.14. Visualization of the adaptively determined time step size k

Figure 7.14 depicts the distribution of the temporal step size k resulting from a fully adaptive computation on dynamic meshes. We observe a strong refinement of the time steps at the beginning of the time interval, whereas the time steps at the end are determined by the adaptation to be eight times larger.

Before presenting a sequence of dynamically changing meshes, we show in Figure 7.15 a typical locally refined mesh obtained by computations on a fixed spatial triangulation. We note, that the refinement is especially concentrated at the four reentrant corners and the two measurement points behind the obstacle. The interior of the region with restricted cross section is also strongly refined.

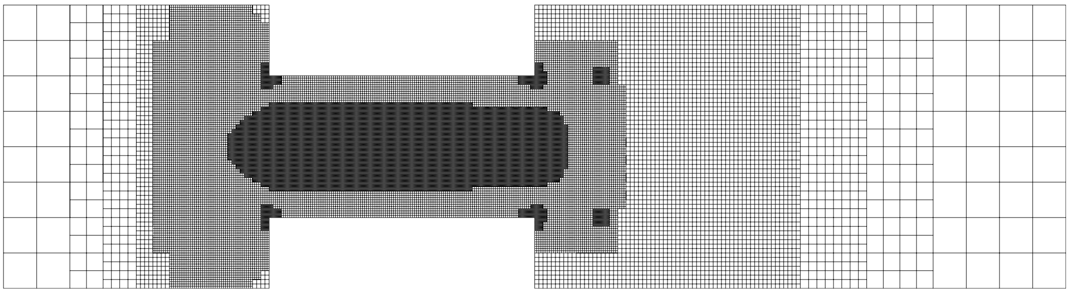
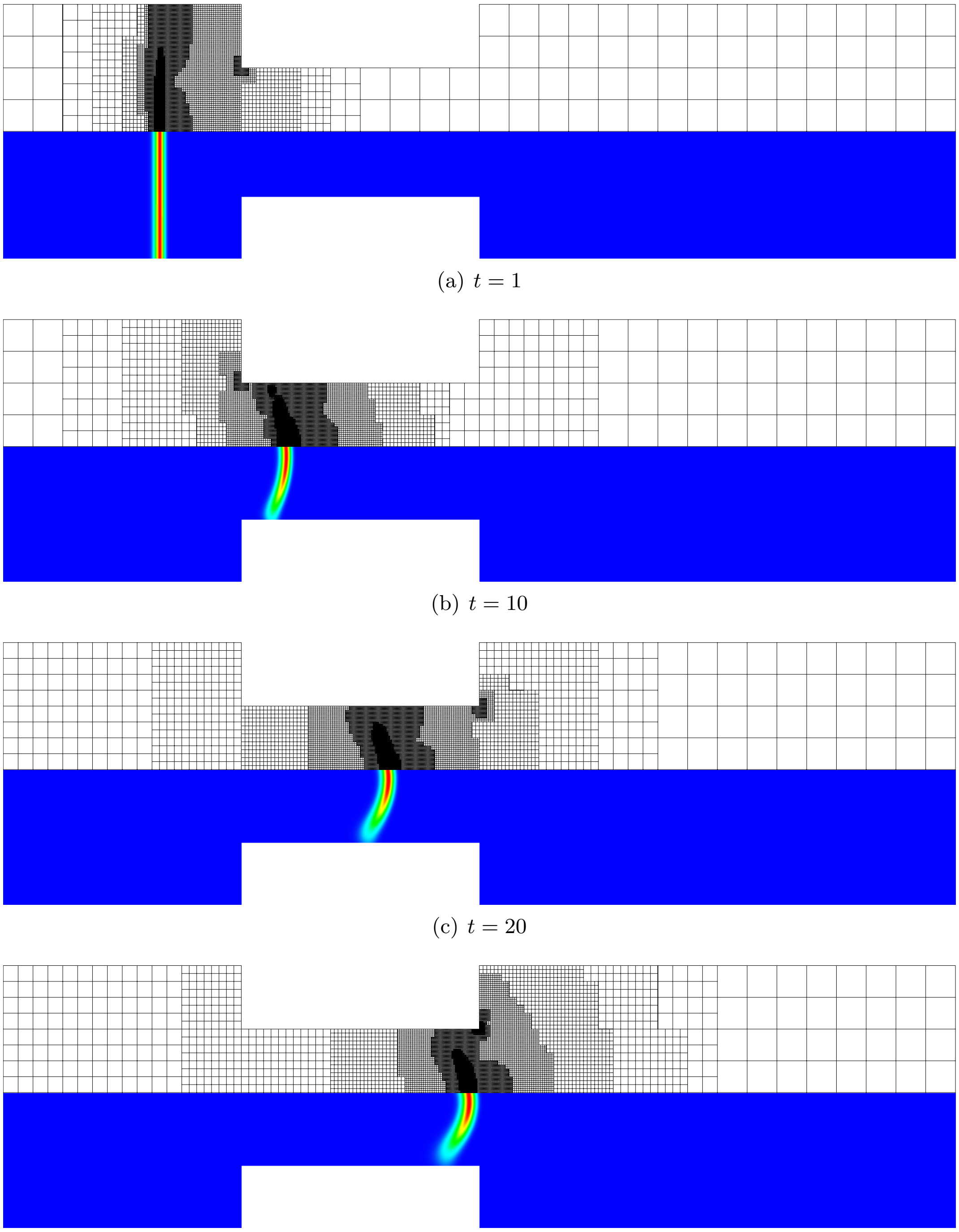


Figure 7.15. Locally refined fixed mesh

Finally, the Figures 7.16, 7.17, and 7.18 show the spatial triangulation and the reaction rate ω for certain selected time points. Thereby, ω is computed from the numerical solution by means of formula (7.6). We observe, that the refinement traces the front of the reaction rate ω until $t \approx 56$ (cf. Figure 7.17(d)). Afterwards, the mesh around the front becomes coarser and the refinement is concentrated at the four measurement points p_i. Compared to the usage of one fixed triangulation, the usage of dynamically changing meshes enables us here to reduce the discretization error in terms of the quantity of interest at lower computational costs; cf. Figure 7.13.



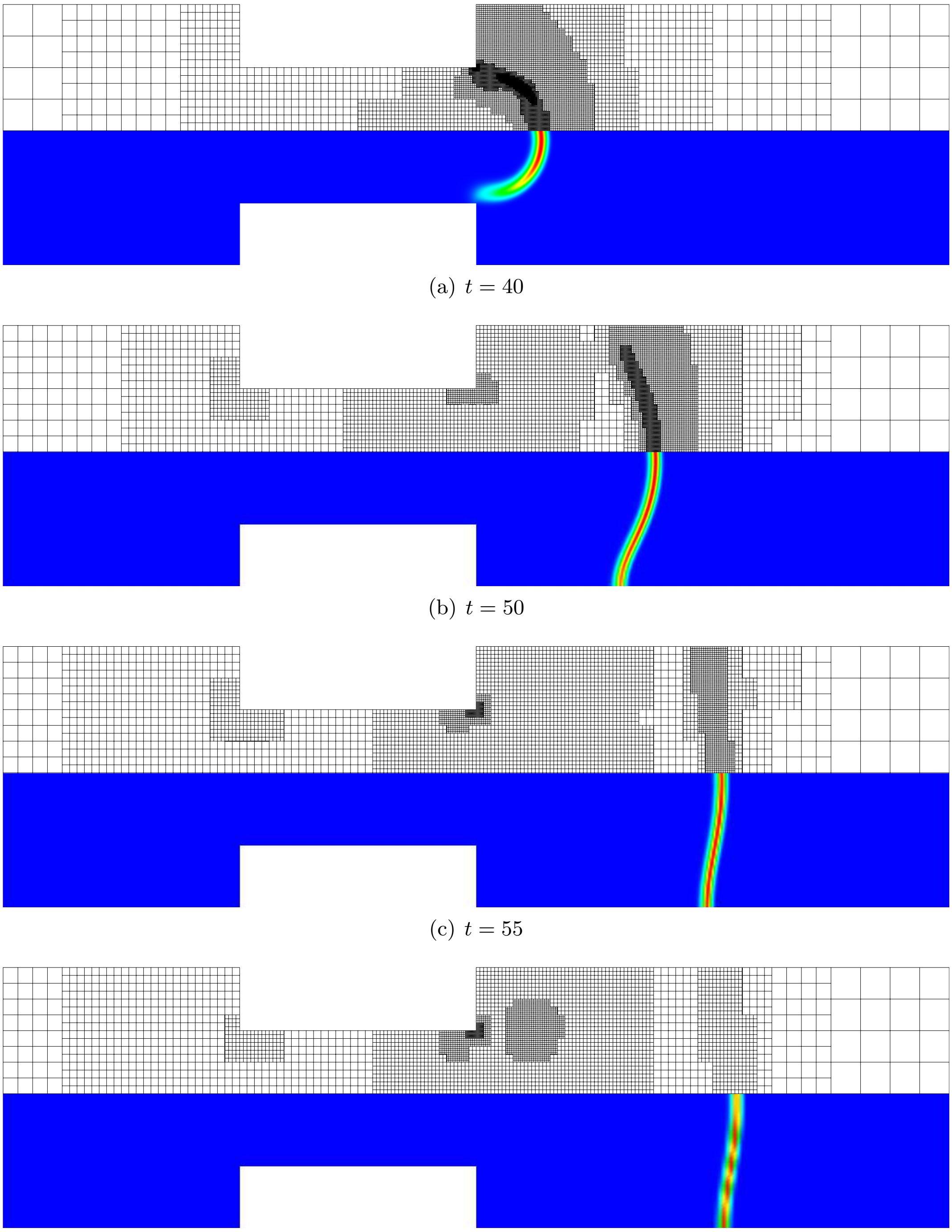
(a) $t = 1$

(b) $t = 10$

(c) $t = 20$

(d) $t = 30$

Figure 7.16. Locally refined meshes and reaction rate ω for $t \in \{1, 10, 20, 30\}$



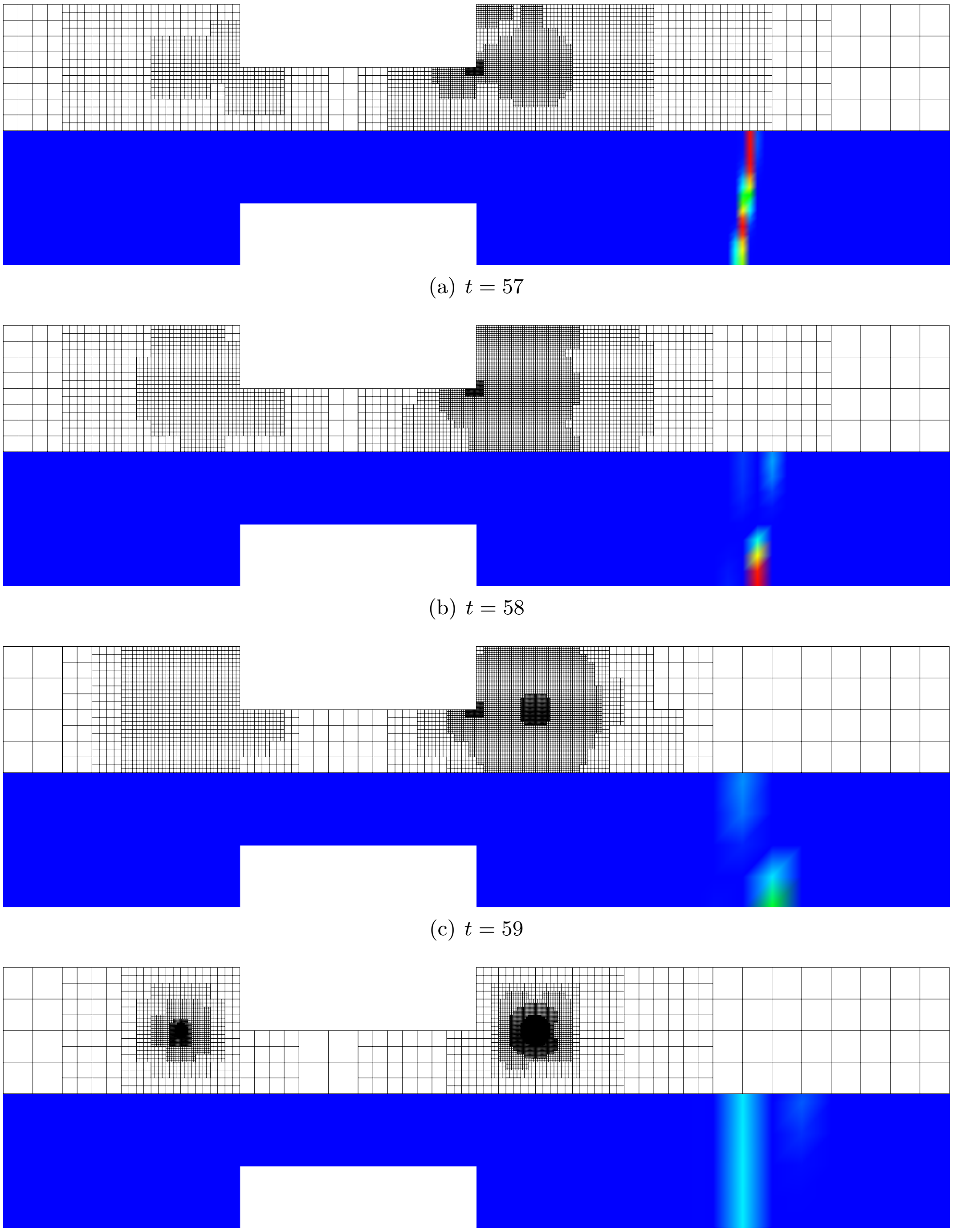
(a) $t = 40$

(b) $t = 50$

(c) $t = 55$

(d) $t = 56$

Figure 7.17. Locally refined meshes and reaction rate ω for $t \in \{ 40, 50, 55, 56 \}$



(a) $t = 57$

(b) $t = 58$

(c) $t = 59$

(d) $t = 60$

Figure 7.18. Locally refined meshes and reaction rate ω for $t \in \{57, 58, 59, 60\}$

8 Conclusions and Perspectives

In this thesis, we developed efficient numerical algorithms for solving a wide class of optimization problems governed by parabolic partial differential equations (PDEs). For the numerical treatment of such problems, we proposed to discretize the governing equations by Galerkin finite element methods in space and time. Also the control variable was approximated by Galerkin discretizations. For this combination of discretizations, we derived *a priori* and *a posteriori* estimates assessing the error caused by the discretization of the optimization problem. In the a priori analysis, we showed in the situation of a linear-quadratic parabolic optimal control problem optimal order of convergence for the error in the control, state, and adjoint state variable, as well as for the error in terms of the cost functional. Moreover, in the presented estimates, the influences of the involved discretizations on the total discretization error were clearly separated.

The main issue of the work at hand was the development of an adaptive refinement procedure aiming at the determination of efficient discretizations for the numerical solution of parabolic optimization problems. To this end, the techniques of a posteriori error estimation for optimization problems governed by elliptic PDEs were extended to estimate the error induced by the finite element discretization of parabolic optimization problems. Therefor, an error estimator was developed which is able to separately assess the errors caused by the temporal and spatial discretizations of the state and the control variables independently of each other. This allows for balancing the different errors by local refinement in space and time leading to efficient discretizations of the considered optimization problems. The presented a posteriori error estimates were derived in a general nonlinear setting for the error measured in terms of the cost functional and in terms of a given quantity of interest. The efficiency of the error estimation and the quality of the resulting discretizations were confirmed by several illustrative examples including comparisons to results obtained from a more heuristic smoothness-based error estimator.

In order to demonstrate the capabilities of our approach to a posteriori error estimation, we applied the developed methods to highly nonlinear optimal control and parameter estimation problems arising from concrete applications. In particular, we considered the optimal control of the intensity of a laser beam employed for the hardening of steel and the identification of an unknown parameter in an Arrhenius law utilized for modeling of traveling flames through a channel. Here, and also for some academic test configurations considered in this thesis, the application of the developed space-time adaptivity yielded a significant saving in terms of degrees of freedom and thus in computational time necessary to solve the problems up to a certain accuracy. Especially for the highly dynamic behavior of the solutions to the mentioned application problems, the usage of spatial discretizations changing in time (*dynamically changing meshes*) led to a further reduction of computational costs.

Based on these achievements, we regard the following aspects as promising for future developments:

In this thesis, we considered optimization problems in the absence of inequality constraints. However, many problems are characterized by additionally given constraints on the control or the state variable. Hence, the extension of the ideas and techniques presented here on a priori and a posteriori error analysis to the case of inequality constrained optimization problems is an important topic.

In the already submitted article [61], Boris Vexler and the author applied the techniques developed for the a priori analysis of linear-quadratic parabolic optimal control problems successfully to the case of problems with pointwise inequality constraints on the control variable. As mentioned, also the development of an a priori analysis for parabolic optimal control problems with pointwise state constraints is of interest. Here, one has to handle particularly the low regularity of the adjoint state. This leak of regularity carries over from the Lagrange multipliers for the state constraint appearing on the right-hand side of the adjoint equation, which are in general only regular Borel measures.

Also in the field of a posteriori error analysis, the derived approach has to be extended to the case of optimization problems with pointwise inequality constraints. For the extension to control constraints, it seems promising to combine the techniques presented in this thesis with the approach developed in Vexler and Wollner [84] for optimization problems governed by elliptic PDEs. In the presence of constraints on the state variable, which can be crucial for concrete applications, there are still some open questions concerning the efficient a posteriori error analysis. However, there exist some approaches to this topic in the case of elliptic state equations. Both control and state constrained optimization problems leak on regularity properties of the optimal solution and the corresponding Lagrange multipliers. This causes serious difficulties which have to be incorporated in the analysis and the practical evaluation of a posteriori error estimates.

A further extension of the derived a priori analysis could be the examination of other configurations of optimal control problems. In the thesis at hand, we considered a linear-quadratic problem with control by the right-hand side, and the cost functional acts distributed over space and time. A next possible step could be the extension to the case where the control enters the state equation via its initial values. Correspondingly, the case of terminal observation (the cost functional is located only at final time) is of interest. Here, for instance, difficulties arising from the low regularity of the state near the initial time have to be overcome. Also the development of comparable a priori estimates for semilinear parabolic optimal control problems with and without constraints is of major interest. Therefor, the estimates developed here have to be combined with techniques for the treatment of (for instance monotone) nonlinearities known from the analysis of semilinear parabolic equations.

Further, we want to apply the proposed error estimation and adaptive refinement strategies to more specific applications including for instance problems arising from biophysics or fluid dynamics.

Acknowledgments

At first, I would like to express my gratitude to my supervisor Rolf Rannacher for suggesting this interesting subject and for always being open for any questions. I would like to thank him for always pointing out the key question and essential objectives of my work, for the encouragement and the possibility to present my results at international conferences at home and abroad, for the help in gaining experience in research and teaching, and also for giving me some insight to the secrets of university administration.

I gratefully acknowledge the funding received towards my Ph.D. from the German Research Foundation (DFG) firstly through the now closed *Collaborative Research Center (SFB) 359 "Reactive Flows, Diffusion, and Transport"* and later through the *International Research Training Group (IGK) 710 "Complex Processes: Modeling, Simulation, and Optimization"* at the Interdisciplinary Center of Scientific Computing (IWR). I appreciate having been a research fellow of IGK 710 during the past three years, and I am grateful for the possibilities the membership of this group has offered.

My special thanks are addressed to Boris Vexler for his enduring support, for countless fruitful discussions, as well as for many valuable suggestions concerning my work. Besides, I want to thank him for the productive and agreeable collaboration in software development and paper writing.

Further, I thank Michael Schmich, with whom I have shared my office, for not hesitating in discussing all arising questions on mathematical and also all kinds of technical topics.

All numerical experiments in this work could not have been done without the optimization toolkit RoDoBo and the finite element library Gascoigne. Hence, I would like to thank all people who have contributed to the development of these software packages: Roland Becker, Olaf Benedix, Malte Braack, Thomas Dunne, Anke Griesbaum, Thomas Richter, Michael Schmich, Boris Vexler, and Winnifried Wollner.

I wish to thank the whole Numerical Analysis Group at the University of Heidelberg for always open doors, for the exchange of ideas, and for the nice atmosphere making my time at work most pleasant.

I thank all my friends and colleagues at the university, particularly Anke, Bärbel, Boris, Christian, Dirk, Frédéric, Helke, Michael, Sandra, Thomas, Tom, and Winni, for all the enjoyable time we spent together inside and outside the institute.

Last but by no means least, I like to thank my parents for their never-ending encouragement and ongoing support in all its forms.

Utilized Software Platforms

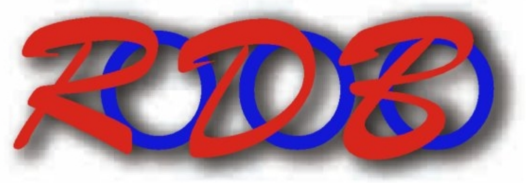
We intend to give here an overview over the software platforms utilized and enhanced for the numerical computations presented in this thesis:

- The optimization toolbox RoDoBo [70]. Here, all the techniques and algorithms for PDE-constrained optimization presented and developed in this thesis were implemented.
- The finite element toolbox Gascoigne [39]. It provides the discretization and solution capabilities for RoDoBo.
- The visualization software VisuSimple [85]. It was utilized to generate the images of the numerical solutions presented in the Chapters 6 and 7.

RoDoBo

Principal developers: R. Becker, D. Meidner, and B. Vexler

Development team: A. Griesbaum, O. Benedix, and W. Wollner



At the beginning of the author's Ph.D. research, Roland Becker, Boris Vexler, and the author initiated jointly the development of the optimization toolbox RoDoBo [70]. The motivation was driven by the matter of fact, that it seemed necessary to have a software environment where all developed ideas concerning PDE-constrained optimization can be tested and applied to concrete problems.

Since the part of RoDoBo dealing with the solution of partial differential equations bases on Gascoigne [39] (see the following section), all features of Gascoigne concerning for instance the types of equations that can be solved are inherited by RoDoBo. Besides the classical Poisson and heat equations, these are especially

- Systems of convection-diffusion-reaction equations
- Incompressible and compressible Navier-Stokes equations
- Reactive flow systems

Also inherited from GASCOIGNE are the capabilities of efficiently solving such problems. In particular, we emphasize the following features:

- Stabilization schemes
- Multigrid methods
- A posteriori error control
- Adaptive mesh refinement in 2D and 3D

For the efficient discretization of nonstationary equations (cf. Chapter 3) and the extension of the a posteriori error estimation techniques to this situation (cf. Chapter 6), RODOBO was enhanced by following features:

- Space-time finite element discretizations
- Dynamically changing meshes in time (in cooperation with M. Schmich)

Motivated by the very flexible user interface for describing the problems to be solved provided by GASCOIGNE, RODOBO is constructed to solve a wide class of optimization problems, that is optimal control and parameter estimation problems where the control may enter via the following parts:

- Equation (right-hand side, diffusion coefficients, reaction rates, ...)
- Boundary conditions of Dirichlet, Neumann, or Robin type
- Initial condition

Furthermore, state-of-the-art optimization algorithms were implemented. Additionally to the basic algorithms described in Chapter 4, these are

- Newton, Gauß-Newton, and Quasi-Newton methods
- Globalization techniques
- Primal-dual active set strategies
- Interior point methods

Besides this, RODOBO provides a flexible concept for evaluating the various formulas describing the derivatives of the reduced cost functional and for automatically assembling the different auxiliary problems to be solved during the optimization procedure.

RODOBO has become an extensively used software package at the Numerical Analysis Group at the University of Heidelberg and at RICAM in Linz for research and also teaching purposes. For instance, it has been used in software labs and for closed and ongoing diploma and Ph.D. theses as well as research projects on:

- Numerical Analysis and Discretization Strategies for Optimal Control Problems with Singularities (O. Benedix, B. Vexler)
- Efficient Computation of Regularization Parameters by Goal-Oriented Adaptive Discretization (A. Griesbaum, B. Kaltenbacher, B. Vexler)

- A Priori Error Estimations for Finite Element Discretization of an Elliptic Optimal Control Problem with a Bilinear State Equation (A. Kröner)
- A Priori Error Analysis for Finite Element Discretizations of Elliptic Optimal Control Problems with Dirichlet Control (S. May)
- Model Reduction by Adaptive Discretization in Optimal Control (R. Rannacher, W. Wollner)
- Adaptive Finite Element Methods for the Optimal Control of Elliptic PDEs with Control Constraints (W. Wollner)

Gascoigne

Principal developers: R. Becker and M. Braack

Development team: T. Dunne, D. Meidner, T. Richter, M. Schmich, B. Vexler, and W. Wollner

The object-oriented finite element toolkit Gascoigne [39] was initially created by Roland Becker and Malte Braack at the University of Heidelberg and at INRIA in Sophia-Antipolis. It has been further developed by various contributors—mainly current or former members of the Numerical Analysis Group at the University of Heidelberg.

The key features employed for solving partial differential equations are:

- Discretization by bi-/tri-linear (Q1) and bi-/tri-quadratic finite elements (Q2) in two and three space dimensions
- Newton's method for solving nonlinear problems
- Multigrid solver with ILU smoother on locally refined meshes
- Error estimation by goal-oriented weighted residual techniques
- Adaptive mesh refinement using quadrilaterals (2D) and hexahedrals (3D) with hanging nodes
- Stabilization by local projection (LPS), GLS, and SUPG

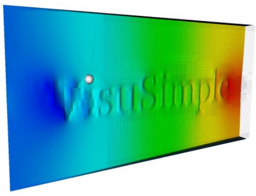
Besides the extensions for optimization purposes collected in RoDoBo, there are several other modules which base on the capabilities of Gascoigne. For example, a version made for dealing with complex chemical systems (by M. Braack), a parallelized version (by T. Richter), and an extension for the handling of dynamically changing meshes for solving nonstationary PDEs (by M. Schmich) exist.

Gascoigne has become the basis of many diploma and Ph.D. theses on a wide range of topics in the field of numerical analysis of partial differential equations. It has also been successfully used in software labs for teaching students the practical aspects of PDE numerics.

VisuSimple

Principal developers: R. Becker and R. Riviere

Development team: T. Dunne and D. Meidner



VisuSimple [85] is an interactive visualization and graphics/mpeg-generation program for two- or three-dimensional data in the VTK format—an easy to implement visual data format. The code of VisuSimple is provided freely to the public under an MIT-like license. Since VisuSimple is programmed using the scripting language Tcl and the excellent visual data processing toolkit VTK, modifications can be practically tested and integrated interactively into the source.

VisuSimple is a simple GUI for doing visualizations by means of VTK. Its main purpose is to speed up the usage of VTK for repetitive applications when sophisticated visualization algorithms are not needed and the size of data to be visualized is relatively small. For the moment, one can (among other things):

- Read structured or unstructured data files in VTK format
- Visualize the grid
- Visualize isolines and isosurfaces
- Visualize carpets of 2D scalar data
- Visualize vectorfields as arrows
- Make simple animations

The VTK library has practically become an industry standard as a visual data post-processing toolkit, so VisuSimple can be seen as an excellent learning opportunity to VTK.

A more detailed description of the capabilities of VisuSimple can be found in Dunne and Becker [27].

List of Figures

List of Tables

List of Algorithms

Bibliography

[1] N. Arada, E. Casas, and F. Tröltzsch. Error estimates for the numerical approximation of a semilinear elliptic control problem. *Comput. Optim. Appl.* 23(2), pp. 201–229, 2002.

[2] R. Becker. *Adaptive Finite Elements for Optimal Control Problems*. Habilitation thesis, Fakultät für Mathematik und Informatik, Universität Heidelberg, 2001.

[3] R. Becker. Estimating the control error in discretized PDE-constrained optimization. *J. Numer. Math.* 14(3), pp. 163–185, 2006.

[4] R. Becker and M. Braack. Multigrid techniques for finite elements on locally refined meshes. *Numer. Linear Algebra Appl.* 7(6), pp. 363–379, 2000.

[5] R. Becker, M. Braack, D. Meidner, R. Rannacher, and B. Vexler. Adaptive finite element methods for PDE-constrained optimal control problems. In *Reactive Flows, Diffusion and Transport*, edited by W. Jäger, R. Rannacher, and J. Warnatz, pp. 177–205. Springer-Verlag, Berlin, 2006.

[6] R. Becker and H. Kapp. Optimization in PDE models with adaptive finite element discretization. In *Numerical Mathematics and Advanced Applications*, edited by H. G. Bock, F. Brezzi, R. Glowinski, G. Kanschat, Y. A. Zuznetsov, J. Périaux, and R. Rannacher, pp. 147–155. World Scientific, London, 1998. Proceedings of ENUMATH 1997.

[7] R. Becker, H. Kapp, and R. Rannacher. Adaptive finite element methods for optimal control of partial differential equations: Basic concepts. *SIAM J. Control Optim.* 39(1), pp. 113–132, 2000.

[8] R. Becker, D. Meidner, and B. Vexler. Efficient numerical solution of parabolic optimization problems by finite element methods. *Optim. Methods Softw.* 22(5), pp. 813–833, 2007. doi:10.1080/10556780701228532.

[9] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *East-West J. Numer. Math.* 4(4), pp. 237–264, 1996.

[10] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation. In *Acta Numerica 2001*, edited by A. Iserles, volume 10, pp. 1–102. Cambridge University Press, 2001.

[11] R. Becker and B. Vexler. A posteriori error estimation for finite element discretizations of parameter identification problems. *Numer. Math.* 96(3), pp. 435–459, 2004.

[12] R. Becker and B. Vexler. Mesh refinement and numerical sensitivity analysis for parameter calibration of partial differential equations. *J. Comput. Phys.* 206(1), pp. 95–110, 2006.

[13] R. Becker and B. Vexler. Optimal control of the convection-diffusion equation using stabilized finite element methods. *Numer. Math.* 106(3), pp. 349–367, 2007.

[14] M. Berggren, R. Glowinski, and J.-L. Lions. A computational approach to controllability issues for flow-related models. (I): Pointwise control of the viscous burgers equation. *Int. J. Comput. Fluid Dyn.* 7(3), pp. 237–253, 1996.

[15] M. Braack. *An Adaptive Finite Element Method for Reactive Flow Problems*. Ph.D. thesis, Naturwissenschaftlich-Mathematische Gesamtfakultät, Universität Heidelberg, 1998.

[16] M. Braack and A. Ern. A posteriori control of modeling errors and discretization errors. *Multiscale Model. Simul.* 1(2), pp. 221–238, 2003.

[17] D. Braess. *Finite Elemente.* Springer-Verlag, Berlin, 3rd edition, 2003.

[18] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Springer-Verlag, Berlin, 2nd edition, 2002.

[19] G. F. Carey and J. T. Oden. *Computational aspects*, volume 3 of *Finite elements*. Prentice-Hall, Englewood Cliffs, 1984.

[20] C. Carstensen and R. Verfürht. Edge residuals dominate a posteriori error estimates for low order finite element methods. *SIAM J. Numer. Anal.* 30(6), pp. 1571–1587, 1999.

[21] E. Casas, M. Mateos, and F. Tröltzsch. Error estimates for the numerical approximation of boundary semilinear elliptic control problems. *Comput. Optim. Appl.* 31(2), pp. 193–219, 2005.

[22] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*, volume 40 of *Classics Appl. Math.* SIAM, Philadelphia, 2002.

[23] A. R. Conn, N. I. M. Gould, and P. L. Toint. *Trust-Region Methods.* MPS/SIAM Ser. Optim. SIAM, Philadelphia, 2000.

[24] B. Dacorogna. *Direct Methods in the Calculus of Variations*, volume 78 of *Appl. Math. Sci.* Springer-Verlag, Berlin, 1989.

[25] R. Dautray and J.-L. Lions. *Evolution Problems I*, volume 5 of *Mathematical Analysis and Numerical Methods for Science and Technology.* Springer-Verlag, Berlin, 1992.

[26] P. Deuflhard. *Newton Methods for Nonlinear Problems*, volume 35 of *Springer Ser. Comput. Math.* Springer-Verlag, Berlin, 2004.

[27] T. Dunne and R. Becker. VisuSimple: An interactive visualization utility for scientific computing. In *Reactive Flows, Diffusion and Transport*, edited by W. Jäger, R. Rannacher, and J. Warnatz, pp. 177–205. Springer-Verlag, Berlin, 2006.

[28] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. In *Acta Numerica 1995*, edited by A. Iserles, volume 4, pp. 105–158. Cambridge University Press, 1995.

[29] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. *Computational differential equations*. Cambridge University Press, Cambridge, 1996.

[30] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems I: A linear model problem. *SIAM J. Numer. Anal.* 28(1), pp. 43–77, 1991.

[31] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems II: Optimal error estimates in $L_\infty L_2$ and $L_\infty L_\infty$. *SIAM J. Numer. Anal.* 32(3), pp. 706–740, 1995.

[32] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems V: Long-time integration. *SIAM J. Numer. Anal.* 32(6), pp. 1750–1763, 1995.

[33] L. C. Evans. *Partial Differential Equations*, volume 19 of *Grad. Stud. Math.* AMS, Providence, 2002.

[34] R. S. Falk. Approximation of a class of optimal control problems with order of convergence estimates. *J. Math. Anal. Appl.* 44, pp. 28–47, 1973.

[35] M. Feistauer and K. Švadlenka. Space-time discontinuous Galerkin method for solving nonstationary convection-diffusion-reaction problems. Preprint MATH-knm-2005/2, Charles University Prague, 2005.

[36] J. Fuhrmann and D. Hömberg. Numerical simulation of the surface hardening of steel. *Internat. J. Numer. Methods Heat Fluid Flow* 9(6), pp. 705–724, 1999.

[37] A. V. Fursikov. *Optimal Control of Distributed Systems: Theory and Applications*, volume 187 of *Transl. Math. Monogr.* AMS, Providence, 1999.

[38] H. Gajewski, K. Gröger, and K. Zacharias. *Nichtlineare Operatorgleichungen und Operatordifferentialgleichungen*. Akademie-Verlag, Berlin, 1974.

[39] Gascoigne. The finite element toolkit. URL `http://www.gascoigne.uni-hd.de`.

[40] T. Geveci. On the approximation of the solution of an optimal control problem governed by an elliptic equation. *M2AN Math. Model. Numer. Anal.* 13, pp. 313–328, 1979.

[41] A. Griewank. Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation. *Optim. Methods Softw.* 1(1), pp. 35–54, 1992.

[42] P. Grisvard. *Elliptic Problems in Nonsmooth Domains*, volume 24 of *Pitman Monogr. Surv. Pure Appl. Math.* Longman Scientific & Technical, Harlow, Essex, 1985.

[43] V. Heuveline and A. Walther. Online checkpointing for adjoint computation in PDEs: Application to goal-oriented adaptivity and flow control. Preprint MATH-WR-07-2004, Technische Universität Dreseden, 2004.

[44] M. Hinze. A variational discretization concept in control constrained optimization: The linear-quadratic case. *Comput. Optim. Appl.* 30(1), pp. 45–61, 2005.

[45] M. Hinze and K. Kunisch. Second order methods for optimal control of time-dependent fluid flow. *SIAM J. Control Optim.* 40(3), pp. 925–946, 2001.

[46] D. Hömberg and S. Volkwein. Suboptimal control of laser surface hardening using proper orthogonal decomposition. Preprint 639, WIAS Berlin, 2001.

[47] J. Jahn. *Introduction to the Theory of Nonlinear Optimization.* Springer-Verlag, Berlin, 2nd edition, 1996.

[48] C. Johnson. *Numerical Solution of Partial Differential Equations by the Finite Element Method.* Cambridge University Press, Cambridge, 1987.

[49] L. V. Kantorovich and G. P. Akilov. *Functional Analysis in Normed Spaces*, volume 46 of *Int. Ser. Monogr. Pure Appl. Math.* The Macmillan Co., New York, 1964.

[50] J. Lang. *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems. Theory, Algorithm, and Applications*, volume 16 of *Lecture Notes in Earth Sci.* Springer-Verlag, Berlin, 1999.

[51] I. Lasiecka and K. Malanowski. On discrete-time Ritz-Galerkin approximation of control constrained optimal control problems for parabolic systems. *Control Cybernet.* 7(1), pp. 21–36, 1978.

[52] J.-B. Leblond and J. Devaux. A new kinetic model for anisothermal metallurgical transformations in steels including effect of austenite grain size. *Acta Metallurgica* 32(1), pp. 137–146, 1984.

[53] J.-L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*, volume 170 of *Grundlehren Math. Wiss.* Springer-Verlag, Berlin, 1971.

[54] W. Liu, H. Ma, T. Tang, and N. Yan. A posteriori error estimates for discontinuous Galerkin time-stepping method for optimal control problems governed by parabolic equations. *SIAM J. Numer. Anal.* 42(3), pp. 1032–1061, 2004.

[55] W. Liu and N. Yan. A posteriori error estimates for distributed convex optimal control problems. *Adv. Comput. Math* 15(1-4), pp. 285–309, 2001.

[56] W. Liu and N. Yan. A posteriori error estimates for control problems governed by nonlinear elliptic equations. *Appl. Num. Math.* 47(2), pp. 173–187, 2003.

[57] K. Malanowski. Convergence of approximations vs. regularity of solutions for convex, control-constrained optimal-control problems. *Appl. Math. Optim.* 8(1), pp. 69–95, 1981.

[58] R. S. McNight and W. E. Bosarge, jr. The Ritz-Galerkin procedure for parabolic control problems. *SIAM J. Control Optim.* 11(3), pp. 510–524, 1973.

[59] D. Meidner and B. Vexler. Adaptive space-time finite element methods for parabolic optimization problems. *SIAM J. Control Optim.* 46(1), pp. 116–142, 2007. doi:10.1137/060648994.

[60] D. Meidner and B. Vexler. A priori error estimates for space-time finite element discretization of parabolic optimal control problems I: Problems without control constraints. *SIAM J. Control Optim.* 2007. To appear.

[61] D. Meidner and B. Vexler. A priori error estimates for space-time finite element discretization of parabolic optimal control problems II: Problems with control constraints. *SIAM J. Control Optim.* 2007. To appear.

[62] C. Meyer and A. Rösch. Superconvergence properties of optimal control problems. *SIAM J. Control Optim.* 43(3), pp. 970–985, 2004.

[63] I. P. Mysovskikh. On the convergence of Newton's method. *Trudy Mat. Inst. Steklov* 28, pp. 145–147, 1949.

[64] P. Neittaanmäki and D. Tiba. *Optimal Control of Nonlinear Parabolic Systems*, volume 179 of *Monogr. Textbooks Pure Appl. Math.* Dekker, New York, 1994.

[65] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer Ser. Oper. Res. Springer-Verlag, Berlin, 1999.

[66] M. Picasso. Anisotropic a posteriori error estimates for an optimal control problem governed by the heat equation. *Numer. Methods Partial Differ. Equations* 22(6), pp. 1314–1336, 2006.

[67] R. Rannacher and B. Vexler. A priori error estimates for the finite element discretization of elliptic parameter identification problems with pointwise measurements. *SIAM J. Control Optim.* 44(5), pp. 1844–1863, 2005.

[68] T. Richter. *Funktionalorientierte Gitteroptimierung bei der Finite-Elemente-Approximation elliptischer Differentialgleichungen.* Diploma thesis, Fakultät für Mathematik und Informatik, Universität Heidelberg, 2001.

[69] T. Richter. *Parallel Multigrid Method for Adaptive Finite Elements with Application to 3D Flow Problems.* Ph.D. thesis, Naturwissenschaftlich-Mathematische Gesamtfakultät, Universität Heidelberg, 2005.

[70] RoDoBo. A C++ library for optimization with stationary and nonstationary PDEs with interface to Gascoigne [39]. URL `http://www.rodobo.uni-hd.de`.

[71] A. Rösch and B. Vexler. Optimal control of the Stokes equations: A priori error analysis for finite element discretization with postprocessing. *SIAM J. Numer. Anal.* 44(5), pp. 1903–1920, 2006.

[72] M. Schmich and B. Vexler. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. *SIAM J. Sci. Comput.* 2007. To appear.

[73] D. Schötzau. *hp-DGFEM for Parabolic Evolution Problems.* Ph.D. thesis, Swiss Federal Institute of Technology, Zürich, 1999.

[74] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.* 20(3), pp. 626–637, 1983.

[75] J. Sternberg. *Memory Efficient Approaches of Second Order for Optimal Control Problems.* Ph.D. thesis, Fakultät für Mathematik und Naturwissenschaften, Technische Universität Dresden, 2005.

[76] V. Thomée. *Galerkin Finite Element Methods for Parabolic Problems*, volume 25 of *Spinger Ser. Comput. Math.* Springer-Verlag, Berlin, 2nd edition, 2006.

[77] F. Tröltzsch. On the Lagrange-Newton-SQP method for the optimal control of semilinear parabolic equations. *SIAM J. Control Optim.* 38(1), pp. 294–312, 1999.

[78] F. Tröltzsch. *Optimale Steuerung partieller Differentialgleichungen.* Vieweg & Sohn, Wiesbaden, 2005.

[79] M. Ulbrich. *Nonsmooth Newton-like Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces.* Habilitation thesis, Fakultät für Mathematik, Technische Universität München, 2002.

[80] R. Verführt. A posteriori error estimation and adaptive mesh-refinement techniques. *J. Comput. Appl. Math.* 50(1–3), pp. 67–83, 1994.

[81] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques.* Wiley/Teubner, New York – Stuttgart, 1996.

[82] B. Vexler. *Adaptive Finite Element Methods for Parameter Identification Problems.* Ph.D. thesis, Naturwissenschaftlich-Mathematische Gesamtfakultät, Universität Heidelberg, 2004.

[83] B. Vexler. Finite element approximation of elliptic Dirichlet optimal control problems. *Numer. Funct. Anal. Optim.* 28(7-8), pp. 957–973, 2007.

[84] B. Vexler and W. Wollner. Adaptive finite elements for elliptic optimization problems with control constraints. *SIAM J. Control Optim.* 2007. To appear.

[85] VisuSimple. An interactive VTK-based visualization and graphics/mpeg-generation program. URL `http://www.visusimple.uni-hd.de`.

[86] A. Walther and A. Griewank. Advantages of binomial checkpointing for memory-reduced adjoint calculations. In *Numerical Mathematics and Advanced Applications*, edited by M. Feistauer, V. Dolejší, P. Knobloch, and K. Najzar, pp. 834–843. Springer-Verlag, Berlin, 2004. Proceedings of ENUMATH 2003.

[87] R. Winther. Error estimates for a Galerkin approximation of a parabolic control problem. *Ann. Math. Pura Appl. (4)* 117, pp. 173–206, 1978.

[88] J. Wloka. *Partial Differential Equations.* Cambridge University Press, Cambridge, 1987.