Optimal control of coupled multiphysics problems: Guidelines for real-life applications demonstrated for a complex fuel cell model

A. Rund K. Chudej J. Kerler H.J. Pesch

K. Sternberg

This article is a preprint. It was published in GAMM-Mitteilungen by Wiley. The version of record is available at http://dx.doi.org/10.1002/gamm.201210011.

GAMM-Mitteilungen, 18 October 2012

Optimal Control of Coupled Multiphysics Problems: Guidelines for Real-life Applications Demonstrated for a Complex Fuel Cell Model

Armin Rund[1], **Kurt Chudej**[2], **Johanna Kerler**[3], **Hans Josef Pesch**[2,*], and **Kati Sternberg**[4]

[1] University of Graz, Institut für Mathematik und Wissenschaftliches Rechnen, Heinrichstraße 36, A 8010 Graz, Austria; formerly University of Bayreuth, Chair of Mathematics in Engineering Sciences, D 95440 Bayreuth, Germany

[2] University of Bayreuth, Chair of Mathematics in Engineering Sciences, D 95440 Bayreuth, Germany

[3] University of Augsburg, Institut für Mathematik, Angewandte Analysis mit Schwerpunkt Numerik; formerly University of Bayreuth, Chair of Mathematics in Engineering Sciences, D 95440 Bayreuth, Germany

[4] Central Medical Affairs, Merz Pharmaceuticals GmbH, Eckenheimer Landstraße 100, D 60318 Frankfurt am Main, Germany; formerly University of Bayreuth, Chair of Mathematics in Engineering Sciences, D 95440 Bayreuth, Germany

Key words Optimal control of multiphysics problems, first optimize then discretize, first discretize then optimize, fuel cell systems, state constraints
MSC (2000) 40-01, 49M05, 49M15, 49M25, 49M37, 65M06, 65M20, 35M10

In practice one often is confronted with the simulation and optimization of dynamical processes of multiphysics systems depending on time and space. This paper shall serve as a guidebook for practioneers who wants to proceed from simulation to optimization. We are specially concerned in this paper with optimal control problems involving partial differential equations. After providing an insight into the theory of optimal control problems with partial differential equations, the focus of the paper lies on their numerical treatment to obtain approximate optimal solutions, or more precisely approximate candidate optimal solutions. The two basic approaches, first optimize then discretize (FOTD) and first discretize then optimize (FDTO) are first described in an abstract setting and then evaluated by means of the fuel cell example of this paper which can serve as a prototype problem for multiphysics processes. The example describes the dynamical behaviour inside a certain type of high temperature fuel cell where gas flows, heat distribution, and potential fields as well as chemical reactions are modelled by hyperbolic and parabolic partial differential as well as ordinary differential algebraic equations. The underlying problem is an optimal control problem where up to 28 instationary partial differential algebraic equations in one, respectively two spatial dimensions are involved. In the conclusion pros and cons of the aforementioned approaches are discussed. Hereby, not only the numerical efficiency but also the investigation of human resources are balanced against each other.

* Corresponding author E-mail: hans-josef.pesch@uni-bayreuth.de, Phone: +49 921 55 7150, Fax: +49 921 55 7155

Copyright line will be provided by the publisher

1 Introduction

With the pioneering work of Lions [19] in the 1970s the theory of optimal control of partial differential equations (PDEs) has been constituted, first only for linear problems. Problems of practical relevance could not be solved at that time, since numerical methods were not developed far enough. Moreover realistic applications usually are semi-linear or quasi-linear. Recently some new textbooks have been published which cover the theoretical state of the art including semi-linear PDEs (see Tröltzsch [27]) as well as numerical methods and some applications including flow control and shape optimization (see Hinze et. al [17] and Borzi and Schulz [3]).

Nevertheless there is still a lack of advise for users like engineers who want to proceed from simulation to optimization for their often extremely complicated problems. For real-life applications one has to take into account not only the efficiency of algorithms from the viewpoint of algorithmic complexity but also the efficiency from the viewpoint of human resources that have to be invested for the solution of realistic problems.

In this paper the authors like to summarize their experiences made in an interdisciplinary project with engineers from academia and industry.[1] The target course of this project was concerned with the simulation, optimization, and control of a certain type of high temperature fuel cell system in order to help controlling the system in daily operation as part of a complex power plant, to optimize load changes, and to increase its life-span. The results obtained have stimulated various subsequent research activities also on the field of optimal control of coupled systems including several qualifying theses until today. From these works guidelines will now be extracted for practioneers for solving optimal control problems for nonlinear dynamical systems involving multiple simultaneous physical phenomena.

As examples we use an instationary spatially 1D, respectively 2D model for a stack of molton carbonate fuel cells (MCFCs). These models, out of a hierarchy of models of different degrees of complexity, include the gas flows through anode and cathode as well as burner and mixer, the chemical reactions within the gas flows and the pores of the cells, the transport, respectively the diffusion of temperature through the electrodes, respectively in the solid, and finally the arising electric fields; see Sundmacher et. al. [26], Heidebrecht [10], and [6] and [24]. All these phenomena have to be simultaneously taken into account in order to describe the reallity as precise as possible and necessary.

The goal of our investigation is the optimization of the dynamical behaviour within a single averaged fuel cell with respect to various performance indices. In addition, certain inequality constraints, e. g., on the solid temperature or its gradient have to be taken into account, in order to avoid material corrosion through thermal stresses. All in all, this multiple physical optimal control problem constitutes an enormous challenge and can serve as a benchmark problem for numerical methods for solving PDE constrained optimization problems.

[1] This joint research project entitled *Optimierte Prozessführung von Brennstoffzellen-Systemen mit Methoden der Nichtlinearen Dynamik* was financed by the Federal Ministry of Education and Research from May 2002 to Dec. 2005. Within this cooperation the partners of the Chair of Mathematics in Engineering Sciences of the University of Bayreuth were the Chair for Process Systems Engineering of the University of Magdeburg, the Max-Planck-Institute for the Dynamic of Complex Technical Systems, Magdeburg, and two companies, the IPF-Heizkraftwerksbetriebsgesellschaft mbH, Magdeburg, and the producer of the fuel cell system investigated here, the MTU CFC Solutions GmbH, Ottobrunn near Munich, later named MTU Onsite Energy of the Tognum group, Friedrichshafen.

The paper is organized as follows. After a glimps on theory and numerics of PDE optimal control, the structure of the fuel cell model is described in a rather compact notation. Different approaches for its numerical optimization are discussed and conclusions are finally drawn for tackeling general nonlinear dynamical systems of similar complexity.

2 A glimpse on theory and numerics

2.1 Theoretical treatment of optimal control problems involving PDEs

In order to survey the main theoretical ideas of PDE constrained optimal control we restrict ourselves to the so-called mother problem of optimal control for a linear elliptic PDE with a distributed control and homogeneous Neumann conditions and an objective functional of tracking type:

$$\min_{u\in U_{\rm ad}} J(y,u) := \frac{1}{2}\iint\limits_{\Omega} (y-y_d)^2\,{\rm d}x + \frac{\lambda}{2}\iint\limits_{\Omega} u^2\,{\rm d}x$$

subject to the constraints

$$-\Delta y + y = u \quad \text{in} \quad \Omega\subset\mathbb{R}^N\,,$$
$$\partial_n y = 0 \quad \text{on} \quad \Gamma=\partial\Omega\,.$$

The state variable y and the control variable u are defined on the bounded Lipschitz domain Ω. The set $U_{\rm ad} := \{u\in L^2(\Omega)\colon u_{\min}\le u\le u_{\max}\ \text{a. e.}\}$ of admissble controls is defined by functions $u_{\min}$ and $u_{\max}$ in $L^\infty(\Omega)$. The function y_d to be tracked is assumed to be of class $L^2(\Omega)$ and the regularisation parameter λ shall be positive, $\lambda>0$.

According to the Theorem of Lax-Milgram the ellipitic boundary value problem has a unique weak solution $y\in H^1(\Omega)$. Therefore, there exists a linear and continuous (control-to-state) solution operator $S\colon U_{\rm ad}\to H^1(\Omega)$ mapping the control space $U_{\rm ad}\subset L^2(\Omega)$ to the state space $H^1(\Omega)$. Herewith we formally can write the so-called reduced problem

$$\min_{U_{\rm ad}} f(u)$$

with $f:\ U_{\rm ad}\to\mathbb{R}$ via $f(u) := J(S(u),u)$. This constitutes an optimization problem in the Hilbert space $L^2(\Omega)$. Due to the convexity of this problem it has a unique optimal solution $\bar u$ which must fulfill the variational inequality

$$\big(S^*\,(S(\bar u)-y_d)+\lambda\,\bar u\,,\,u-\bar u\big)_{L^2(\Omega)}\ge 0 \qquad \forall\,u\in U_{\rm ad}\,,$$

where $(\cdot,\cdot)_{L^2(\Omega)}$ denotes the scalar product in $L^2(\Omega)$ and $S^*\colon H^1(\Omega)^*\to L^2(\Omega)^*=L^2(\Omega)$ the adjoint solution operator.

Because of the convexity of the problem, this necessary condition is sufficient, too.

By obeying $S(\bar u)-y_d=\bar y-y_d$ with the optimal state $\bar y$ associated with the optimal control $\bar u$ and by defining an adjoint state $p:=S^*\,(\bar y-y_d)$, the above variational inequality can be rewritten as

$$\big(p+\lambda\,\bar u\,,\,u-\bar u\big)_U\ge 0 \qquad \forall\,u\in U_{\rm ad}$$

with the adjoint state variable p satisfying the adjoint equation

$$-\Delta p + p = \bar y - y_d \quad \text{in} \quad \Omega\subset\mathbb{R}^N\,,$$

$$\partial_n p = 0 \quad \text{on} \quad \Gamma = \partial\Omega\,.$$

Moreover, the variational inequality can be replaced by the pointwise control law

$$\bar{u}(x) = P_{U_{\rm ad}}\left\{-\frac{1}{\lambda}\,p(x)\right\}$$

with the projection operator $P_{U_{\rm ad}}(z) := \min\{u_{\max}, \max\{u_{\min}, z\}\}$ onto the set of admissible controls.

Hence, the optimality system for our mother problem consists of a coupled two-dimensional elliptic system with a non-differentiable projection operator.

In the following, we consider more general semi-linear parabolic optimal control problems with both a distributed control $v \in V_{\rm ad} := \{v \in L^2(Q)\colon v_{\min} \le v \le v_{\max} \text{ a. e.}\}$ with $v_{\min}$ and $v_{\max}$ in $L^\infty(Q)$ and a boundary control $u \in U_{\rm ad} := \{u \in L^2(\Sigma)\colon u_{\min} \le u \le u_{\max} \text{ a. e.}\}$ with $u_{\min}$ and $u_{\max}$ in $L^\infty(\Sigma)$. Here the space-time cylinder Q is defined by $Q := \Omega \times (0,T)$ with $\Omega \subset \mathbb{R}^N$ bounded and Lipschitzian. Its boundary Σ is defined by $\Sigma := \Gamma \times (0,T)$ with $\Gamma = \partial\Omega$. The objective function shall be of type

$$\begin{aligned}\min_{u\in U_{\rm ad}, v\in V_{\rm ad}} J(y,v,u) := &\iint\limits_{\Omega} \Theta(x, y(x,T))\,\mathrm{d}x \\ &+ \iint\limits_{Q} \Phi(x,t,y(x,t),v(x,t))\,\mathrm{d}x\,\mathrm{d}t + \iint\limits_{\Sigma} \Psi(x,t,y(x,t),u(x,t))\,\mathrm{d}s\,\mathrm{d}t\,.\end{aligned}$$

Under suitable assumptions on the nonlinearities (see [27], pp. 266ff), the following optimality system holds:

$$\begin{aligned}
\bar{y}_t + \mathcal{A}\,\bar{y} + d(x,t,\bar{y}) &= \bar{u} \quad \text{in} \quad Q\,,\\
\partial_{\nu_\mathcal{A}}\bar{y} + b(x,t,\bar{y}) &= \bar{v} \quad \text{in} \quad \Sigma\,,\\
\bar{y}(\cdot,0) &= y_0 \quad \text{in} \quad \Omega\,,\\
-p_t + \mathcal{A}\,p + d_y(x,t,\bar{y})\,p &= \Phi_y(x,t,\bar{y},\bar{v}) \quad \text{in} \quad Q\,,\\
\partial_{\nu_\mathcal{A}} p + b_y(x,t,\bar{y})\,p &= \Psi_y(x,t,\bar{y},\bar{u}) \quad \text{in} \quad \Sigma\,,\\
p(\cdot,T) &= \Theta_y(x,\bar{y}(x,T)) \quad \text{in} \quad \Omega\,,\\
\iint\limits_{Q} \left(p + \Phi_v(x,t,\bar{y},\bar{v})\,(v-\bar{v})\right)\,\mathrm{d}x\,\mathrm{d}t \ge 0 &\quad \forall v \in V_{\rm ad}\,,\\
\iint\limits_{\Sigma} \left(p + \Psi_u(x,t,\bar{y},\bar{u})\,(u-\bar{u})\right)\,\mathrm{d}s\,\mathrm{d}t \ge 0 &\quad \forall u \in U_{\rm ad}\,.
\end{aligned}$$

Here $\mathcal{A}\,y(x,t) := -\sum_{i,j=1}^N D_i\left(a_{i,j}(x,t)\,D_j y(x,t)\right)$ with $a_{i,j} \in L^\infty(Q)$ denotes a self-adjoint uniformly elliptic differential operator in divergence form and $\partial_{\nu_\mathcal{A}}$ its associated derivative in the direction of the conormal $\nu_\mathcal{A} := A\,\nu$ with $A := (a_{i,j})$ where D_i respectively ν denote spacial partial derivatives, respectively the outer normal on Σ. The appropriate function spaces for state and adjoint variables now are $W(0,T) \cap C(\bar{Q})$, respectively $W(0,T) \cap L^\infty(Q)$; see [27], pp. 268ff.

Moreover, $\operatorname{grad} f := (p + \Phi_v(x,t,y,v)\,,\; p + \Psi_u(x,t,y,u))$ can be identified with the gradient of the reduced functional $f(u,v) := J(S(v,u),v,u)$.

Note that all nonlinear terms can be easily differentiated by symbolic or automatic differentiation independently of their complexity. Moreover the differential operators in the adjoint system can be easily determined; they remain unchanged compared to the associated operator term of the state equation in case of (self-adjoint) linear second order operators while first order operator terms change their signs. This is due to the number of partial integrations that have to be performed for the associated weak form of the PDE. For more details on the derivation of necessary conditions, it is referred to the textbook of Tröltzsch [27].

Such optimality systems can be comfortably set up by the so-called formal Lagrange technique, here demonstrated for the simple elliptic mother problem. Define the Lagrangian $\mathcal{L}$ and integrate twice by parts

$$\begin{aligned}
&\mathcal{L}(y,u,p)\\
&\quad := J(y,u) - \iint_\Omega (-\Delta y + y - u)\; p \,\mathrm{d}x - \int_{\partial\Omega} \partial_n y\, p \,\mathrm{d}s\\
&\quad = J(y,u) - \iint_\Omega \nabla y \cdot \nabla p + (y-u)\, p \,\mathrm{d}x\\
&\quad = J(y,u) - \iint_\Omega (-\Delta p + p)\; y - u\, p \,\mathrm{d}x - \int_{\partial\Omega} \partial_n p\, y \,\mathrm{d}s\,.
\end{aligned}$$

Then a differentiation in the direction of $h \in H^1(\Omega)$, respectively of $u - \bar{u}$, $u \in U_{\mathrm{ad}}$, yields

$$\begin{aligned}
&D_y\mathcal{L}(\bar{y},\bar{u},p)\, h = 0 \;\forall h \in H^1(\Omega)\\
&\quad \Longrightarrow -\Delta p + p = \bar{y} - y_d\,,\; \partial_n p = 0\,,\\
&D_u\mathcal{L}(\bar{y},\bar{u},p)\,(u-\bar{u}) \geq 0 \;\forall u \in U_{\mathrm{ad}}\\
&\quad \Longrightarrow \big(p + \lambda\,\bar{u}\,,\; u - \bar{u}\big)_U \geq 0 \;\forall\, u \in U_{\mathrm{ad}}\,.
\end{aligned}$$

Again, using Riesz Theorem we can identify $f'(u) = D_u\mathcal{L}(y,u,p)$. To have an explicit expression for the gradient will be important for the design of numerical methods.

Treatable generalizations are mixed control-state constraints or pure state constraints; see, e. g., [27], [17], [8]. Control constraints and some mixed constraints can be treated efficiently by primal-dual active set strategies; see [2], [12]. Especially pure state constraints are very involved and ask for regularization techniques, see [13], [14] and [20]. Also other differential operators (not self-adjoint, nonlinear of first order), more general objective functionals, or special control models, such as $u(x,t) := f(x)\,v(t)$ (time varying with prescribed spacial profile) can be treated, too. The latter generalizations are particularly treatable by the formal Lagrange technique.

2.2 Concepts of discretization and optimization for optimal control problems involving PDEs

In order to discuss the various approaches for the solution of PDE optimal control problems let us write a general problem of PDE constrained optimal control in the following abstract

form:

$$(P) \qquad \min_{u \in U_{\rm ad}\,,\, y \in Y_{\rm ad}} J(y,u) \quad \text{s.t.} \quad PDE(y) = B(u)\,.$$

Here PDE stands for an arbitrary differential equation with its associated initial and boundary condition for the state vector function y, and B is an operator acting on the control vector function u. The aim in solving those problems efficiently is to keep the ratio of the amount of computation for the optimization versus the amount of computation for the simulation of the problem below a small constant, which obviously cannot undershoot 2 in view of the above theoretical results. More important may be however the time for human resources for transferring a practical problem into a computable form.

The two essential numerical procedures are known as direct, respectively *first discretize then optimize* method (FDTO), and indirect, respectively *first optimize then discretize* method (FOTD).[2] With view on time dependent PDEs also intermediate versions may apply.

A FDTO-method can be described as follows:

$$(P_h) \min_{u_h \in U_{\rm ad}^h\,,\, y_h \in Y_{\rm ad}^h} J_h(y_h,u_h) \text{ s.t. } PDE_h(y_h) = B_h(u_h)$$

where the index h indicates the fineness of the discetization. Crucial are the choices of the finite-dimensional spaces $U_{\rm ad}^h$ and $Y_{\rm ad}^h$ for the state, respectively control variables as well as the discretization of the PDEs. This approach leads to a generally huge nonlinear programming problem (NLP) for which even today's high performance software for large scale nonlinear programming may fail unless the problems are of moderate size.

In contrast, FOTD-methods exploit the first order necessary conditions: The minimization of the reduced functional $f(u) := J(S(u),u)$ requires the solution of an optimality system of the form $f'(u) = 0$, i. e.,

$$\begin{aligned} PDE(y) &= B(u)\,, \\ ADJ(p) &= C(y,u)\,, \\ VI(y,p,u) &\geq 0\,, \end{aligned}$$

where ADJ stands for the adjoint system[3] and VI for a system of variational inequalities. This optimality system has then to be discretized, say as $F_h(u_h) = 0$. Here an appropriate finite-dimensional ansatz space for the adjoint state has to be additionally chosen.

In summary, *first discretize then optimize* means: replace all quantities of the infinite-dimensional optimization problem by finite-dimensional substitutes and solve an NLP:

$$\min f(u) = J(S(u),u) \Longrightarrow \min f_h(u_h) = J_h(S_h(u_h),u_h)\,,$$

[2] The classification direct vs. indirect methods is not commonly used in the PDE community where FDTO vs. FOTD is prefered. In the ODE community, an indirect method is a method which exploits the necessary conditions of optimal control theory, i. e., the maximum principle of Pontryagin et. al. Here the user must provide the adjoint system. In contrast, direct methods do not require it. Therefore, they are more appropriate for the "unaware" user. Moreover direct NLP-based methods usually have larger domains of convergence than Newton-like methods for the solution of the multipoint boundary value problems that result from the necessary conditions. This requires considerably fewer human resources since the effort for homotopy techniques can be reduced, if not avoided. However, with regard to more recent developments as tools for automatic differentiation or concepts where FDTO equals FOTD when specially taylored discretization schemes are employed, the traditional contrast is blured.

[3] It is adverted here to the dependence of the right side of the adjoint system on the control which is not the case for the usual textbook problems. This is included here with view on the fuel cell example investigated in Sec. 3.5.

while *first optimize then discretize* means: derive optimality conditions of the infinite-dimensional system, discretize the optimality system and find a solution of the discretized optimality system:

$$f'(u) = 0 \Longrightarrow F_h(u_h) = 0\,.$$

In ideal manner, one should strive for a discrete concept for which both approaches commute, which is, for example, the case if one uses Galerkin methods in space and time; see Richter, Springer, and Vexler [22].

For further variants we introduce the notation $\mathrm{FO}_\infty\mathrm{TD}_\mathrm{h}$ for **F**irst deriving the **O**ptimality system in an **infinite**–dimensional setting and **T**hen **D**iscretize it. The notation $\mathrm{FD}_\mathrm{h}\mathrm{TO}_\mathbb{N}$ is used for firstly discretizing the infinite-dimensional problem and then solving the resulting finite-dimensional nonlinear programming problem. So far the subscript h stands for a general discretization with N spatial degrees of freedom. With view on time dependent PDEs we will now distinguish between spatial and time discretization which is indicated by the subscripts h with N spatial degrees of freedom, respectively k with M grid points on the time axis. Note that the subscript $\mathbb{N}$ in $O_\mathbb{N}$ shall solely indicate the finite dimension of the NLP to be solved.

For time dependent PDEs we may first start with a semi-discretization in space leading to an optimal control problem for an ordinary differential algebraic system (DAE):

$$(P_h(t)) \quad \min_{u_h(t)\in U_\mathrm{ad}^h,\, y_h(t)\in Y_\mathrm{ad}^h} J_h(y_h(t), u_h(t)) \text{ s.t. } DAE_h(y_h(t)) = B_h(u_h(t))\,.$$

Now we again have the choice either to optimize or to discretize first, hence using the above notation $\mathrm{FD}_\mathrm{h}\mathrm{TO}_\infty\mathrm{TD}_\mathrm{k}$ or $\mathrm{FD}_\mathrm{h}\mathrm{TD}_\mathrm{k}\mathrm{TO}_\mathbb{N}$. The latter corresponds to a full discretization $\mathrm{FD}_\mathrm{hk}\mathrm{TO}_\mathbb{N}$.

These concepts will now be discussed by means of our real-life example.

Textbooks focussing on numerical methods for PDE constrained optimization are Hinze et. al. [17] and Borzi and Schulz [3]; see also the survey article of Herzog and Kunisch [11].

3 Optimization of molten carbonate fuel cells

Fuel cells allow an enviromentally friendly energy production based on a controlled oxy-hydrogen reaction. Fuel cell systems can be classified according to their range of application: mobile or stationary, to their operating temperature: low, moderate, or high, and to the material of the membrane of the fuel cell: polymer (PEMFC, PEFC, DMFC), carbonate (CFC), solid oxide (SOFC), and others.

Here, we exploit two models of a hierarchy of mathematical models for so-called Molten Carbonate Fuel Cells (MCFCs), which are utilized for stationary power-heat coupling in the Hotmodule of MTU CFC Solutions GmbH, Ottobrunn; see Fig. 1. These models have been developed by Heidebrecht [10] and later simplified in [24]. A major task in operating such MCFC plants is to control appropriate technical devices in an optimal way in order to perform a change in the external load. This gives rise to several optimal control problems, which we will investigate in this section.

3.1 Aspects of modelling

First, the functionality of MCFCs shall be described briefly. MCFCs are high temperature fuel cells that can produce the necessary hydrogen from the fuel gas (e. g., methane CH_4) because

Fig. 1 The Hotmodule under construction at MTU CFC Solutions GmbH, Ottobrunn near Munich: the stack of about 300 fuel cell is just pushed into the module

of their high operating temperature combined with the use of a catalyst. This technique is known as internal reforming and is one of the advantages of MCFCs, since they do not rely on pure hydrogen fuel gas, which is hard to store and does not occur naturally to a large extent, despite it is the most abundant chemical element on Earth.

One fuel cell out of a stack is schematically depicted in Fig. 2, respectively 3. The galvanic cell in the center produces electric current, if the anode is provided with hydrogen and the cathode with oxygen. Therefore, each electrode is accompanied by a gas channel. The two electrodes are separated by a special electrolyte, which only carbonate ions CO_3^{2-} can pass.

The anode gas channel (top) is fed with a mix of methane and water. Hydrogen is produced by internal reforming based on the upper two reforming reactions (Fig. 2), which are (together) endothermic, i. e., they consume energy. The hydrogen then diffuses into the pores of the anode electrode sustaining the two oxidation reactions (mid of Fig. 2), which need the carbonate ions out off the electrolyte and produce free electrons. The anode exhaust gas is completely oxidized in the burning chamber by inserting air. After passing the mixer, the gas enters the cathode gas channel and diffuses into the pores of the cathode electrode. There, the reduction reaction regenerates the carbonate ions, which pass through the electrolyte to the anode while the electrons are produced in the anode and consumed in the cathode. A part of the exhaust gas of the cathode can be refed into the catalytic burner to improve the oxidation in the burner chamber by rising the temperature therein. Note that the exhaust gas at the cathode outlet can be graded as air according to the strict German Federal Pollution Control Act.

The spatially 1D counter flow configuration of Fig. 2 forms the basis of the simplified 1D model of [24]. In contrast, Fig. 3 depicts a spatially 2D cross flow configuration, which is realized in the Hotmodule mentioned above and which is incorporated in the 2D model of [10], also investigated in [25] and [6].

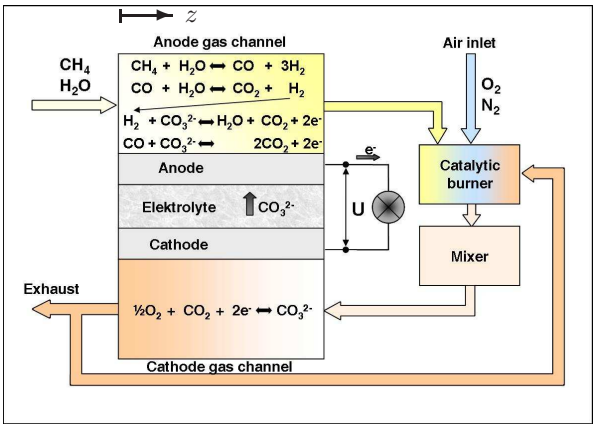

Fig. 2 Functionality of a molten carbonate fuel cell, here for the 1D counter flow design [25].

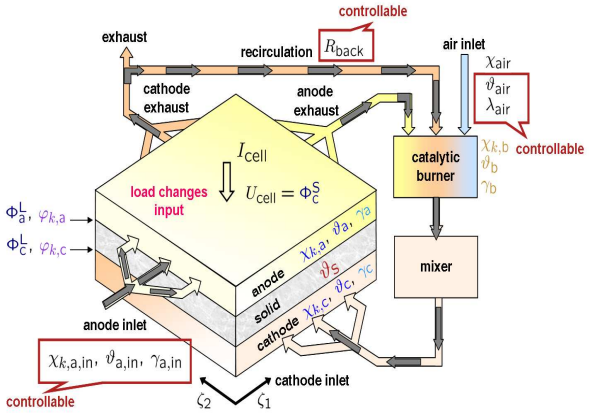

Fig. 3 Notation of variables, here for the 2D cross flow design [25]: χ = molar fraction, ϑ = temperature, γ = molar flow density, Φ = Potential, φ = partial pressure, U_{cell} = cell voltage, I_{cell} = cell current, R_{back} = fractional amount of recycled gas, λ_{air} = air number. The indices are (a,c,e,b,m) for the compartment and $k \in \mathcal{I}$ for the chemical compounds.

The variables appearing in the PDAE model due to [10] can be obtained from Fig. 3. In summary, seven chemical compounds indexed by $k \in \mathcal{I}$ appear: $\mathcal{I} := \{\mathrm{CH_4}, \mathrm{H_2O}, \mathrm{H_2}, \mathrm{CO}, \mathrm{CO_2}, \mathrm{O_2}, \mathrm{N_2}\}$. The fuel cell stack can be controlled by handling the gas and air inlet, as well as the amount of recycled gas.

The 1D and the 2D model share the same characteristic structure, which we discuss in the following by means of the 1D model depicted in Fig. 4. The model is given by a set of differential algebraic equations for the state variables. To shorten the notation, we abbreviate the complicated source terms according to their category as f^q, g^q, and h^q, where q stands

PDEs in gas channels and solid in (z, t):

$$\begin{aligned}
\frac{\partial \chi_{k,\mathrm{a}}}{\partial t} + \gamma_\mathrm{a}\, \vartheta_\mathrm{a}\, \frac{\partial \chi_{k,\mathrm{a}}}{\partial z} &= f^{\chi_{k,a}}\,, \\
\frac{\partial \chi_{k,\mathrm{c}}}{\partial t} - \gamma_\mathrm{c}\, \vartheta_\mathrm{c}\, \frac{\partial \chi_{k,\mathrm{c}}}{\partial z} &= f^{\chi_{k,c}}\,, \\
\frac{\partial \vartheta_\mathrm{a}}{\partial t} + \gamma_\mathrm{a}\, \vartheta_\mathrm{a}\, \frac{\partial \vartheta_\mathrm{a}}{\partial z} &= f^{\vartheta_\mathrm{a}}\,, \\
\frac{\partial \vartheta_\mathrm{c}}{\partial t} - \gamma_\mathrm{c}\, \vartheta_\mathrm{c}\, \frac{\partial \vartheta_\mathrm{c}}{\partial z} &= f^{\vartheta_\mathrm{c}}\,, \\
\frac{\partial (\gamma_\mathrm{a}\, \vartheta_\mathrm{a})}{\partial z} &= f^{\gamma_\mathrm{a}\vartheta_\mathrm{a}}\,, \\
-\frac{\partial (\gamma_\mathrm{c}\, \vartheta_\mathrm{c})}{\partial z} &= f^{\gamma_\mathrm{c}\vartheta_\mathrm{c}}\,, \\
c_\mathrm{p,s}\, \frac{\partial \vartheta_\mathrm{s}}{\partial t} - \frac{1}{\mathrm{Pe}}\, \frac{\partial^2 \vartheta_\mathrm{s}}{\partial z^2} &= f^{\vartheta_\mathrm{s}}\,,
\end{aligned}$$

DAEs of burner and mixer (in t):

$$\begin{aligned}
\chi_{k,\mathrm{b}} &= g^{\chi_{k,\mathrm{b}}}\,, \\
\vartheta_\mathrm{b} &= g^{\vartheta_\mathrm{b}}\,, \\
\gamma_\mathrm{b} &= g^{\gamma_\mathrm{b}}\,, \\
\gamma_\mathrm{air} &= g^{\gamma_\mathrm{air}}\,, \\
\frac{\mathrm{d}\chi_{k,\mathrm{m}}}{\mathrm{d}t} &= g^{\chi_{k,\mathrm{m}}}\,, \\
\frac{\mathrm{d}\vartheta_\mathrm{m}}{\mathrm{d}t} &= g^{\vartheta_\mathrm{m}}\,, \\
\gamma_\mathrm{m} &= g^{\gamma_\mathrm{m}}\,,
\end{aligned}$$

currents and the potential fields:

$$\begin{aligned}
I_\mathrm{a}(t) &= \int_0^1 i_\mathrm{a}(z,t)\,\mathrm{d}z\,, & i_\mathrm{a}(z,t) &= f^{i_\mathrm{a}}\,, & \frac{\mathrm{d}U_\mathrm{cell}}{\mathrm{d}t}(t) &= d^{U_\mathrm{cell}}\,, \\
I_\mathrm{c}(t) &= \int_0^1 i_\mathrm{c}(z,t)\,\mathrm{d}z\,, & i_\mathrm{c}(z,t) &= f^{i_\mathrm{c}}\,, & \frac{\partial \Phi_\mathrm{a}^\mathrm{L}}{\partial t}(z,t) &= f^{\Phi_\mathrm{a}^\mathrm{L}}\,, \\
I_\mathrm{e}(t) &= \int_0^1 i_\mathrm{e}(z,t)\,\mathrm{d}z\,, & i_\mathrm{e}(z,t) &= f^{i_\mathrm{e}}\,, & \frac{\partial \Phi_\mathrm{c}^\mathrm{L}}{\partial t}(z,t) &= f^{\Phi_\mathrm{c}^\mathrm{L}}\,, \\
& & i(z,t) &= f^i + d^i\,.
\end{aligned}$$

Fig. 4 1D model of the MCFC. Each equation of the molar fractions $\chi_{k,\cdot}$ appears sevenfold marked by the index $k \in \mathcal{I}$. Initial and boundary conditions are not depicted. The nonlinear source terms are abbreviated by f^q (time and space dependent) or g^q, d^q (time dependent) with q denoting the corresponding state variable. In detail, f^q depends on all space (and time) dependent state variables from $Z := \{\vartheta_\mathrm{a}, \vartheta_\mathrm{c}, \chi_{k,\mathrm{a}}, \chi_{k,\mathrm{c}}, \vartheta_\mathrm{s}, \gamma_\mathrm{a}, \gamma_\mathrm{c}, i_\mathrm{a}, i_\mathrm{c}, i_\mathrm{e}, i, \Phi_\mathrm{a}^\mathrm{L}, \Phi_\mathrm{c}^\mathrm{L}\}$ and additionally on $U_\mathrm{cell}(t)$. g^q depends on the state variables in $B := \{\vartheta_\mathrm{a}(1,t), \chi_{k,\mathrm{a}}(1,t), \gamma_\mathrm{a}(1,t), \vartheta_\mathrm{c}(0,t), \chi_{k,\mathrm{c}}(0,t), \gamma_\mathrm{c}(0,t), \vartheta_\mathrm{b}(t), \chi_{k,\mathrm{b}}(t), \gamma_\mathrm{b}(t), \vartheta_\mathrm{m}(t), \chi_{k,\mathrm{m}}(t), \gamma_\mathrm{m}(t), \gamma_\mathrm{air}(t)\}$ and additionally on the controls from $u \in U$. There are additional source terms $d^q = d^q(I_\mathrm{a}(t), I_\mathrm{c}(t), I_\mathrm{e}(t), I_\mathrm{cell}(t))$ depending on the currents in the equations of U_cell and i. $c_\mathrm{p,s}$ and Pe are constants.

for the notation of state variables. The terms f^q describe the reactions as well as the mass and heat exchange with neighbouring compartments. Mathematically f^q may depend on all space-dependent state variables as well as on the cell voltage $U_\mathrm{cell}(t)$. In burner and mixer, the respective source terms g^q are functions of solely time-depentent variables: on the control variables, the output values of the gas channels as well as the state variables inside the burner

and the mixer. The terms d^q depends on the three currents $I_{\mathrm{a|c|e}}(t)$ as well as the cell current $I_{\mathrm{cell}}(t)$.

The complete system of equations in Fig. 4 consists of a set of quasi-linear hyperbolic reaction advection equations for the molar fractions $\chi_{k,\mathrm{a}}$ of all chemical compounds in the anode gas channel indexed by $k \in \mathcal{I}$, respectively $\chi_{k,\mathrm{c}}$ in the cathode gas channel, furthermore for the anode and cathode gas temperatures ϑ_{a} and ϑ_{c} as well as for the molar flow densities γ_{a} and γ_{c}, and, moreover, of a semi-linear heat equation for the solid temperature ϑ_{s}. Two of the equations degenerate, i. e., do not depend on partial derivatives with respect to the (dimensionless) time t. The spatial variables are denoted by $z \in [0,1]$ for the 1D counter flow design (Fig. 2) and by ζ_1 and ζ_2 for the 2D cross flow design (Fig. 3). The potentials at the ion layers $\Phi^{\mathrm{L}}_{a|_c}$ are modelled by ordinary differential equations (ODEs) in each spatial point, and the cell voltage U_{cell} by one scalar ODE. Furthermore, there are non-local dependencies induced by the currents $I_{a|c|e}$, which are the integrals over the current densities $i_{a/c/e}$. The currents enter solely the two source terms d^i and $d^{U_{\mathrm{cell}}}$. The system exhibits multiple couplings through the nonlinear source terms f^q and g^q (the d^qs are both linear) and the nonlinear convection terms. Depending on the complexity of the model to be used, the diffusion into the pores of the electrodes can be taken into account by additional algebraic equations (AE) for the partial pressures $\varphi_{k,\mathrm{a}}$ and $\varphi_{k,\mathrm{c}}$; see [10]. Finally, I_{cell} is an input variable that describes the load of the fuel cell system. We will model sudden load changes by means of a Heaviside function in I_{cell}.

Each PDE and ODE possesses an initial condition. The PDE variables must additionally fulfill boundary conditions: inflow conditions for the gas channels (Dirichlet conditions) and a symmetry, respectively insulation condition for the solid temperature. The inflow condition at the anode inlet are controled by time dependent boundary control variables; see Fig. 3. The inflow condition of the cathode gas channel depend on the outflow values of both the anode and cathode gas channel. Their relations are computed by a system of ordinary DAEs which describes the dynamics of burner and mixer, where again boundary controls act on the air inlet; see again Fig. 3.

Complete models with up to 28 state variables can be found for the 2D cross flow model in [10] and [6] and for the 1D counter flow model in [24] and [23].

This dynamical system can be controlled by up to seven controls $u \in U := \{\gamma_{\mathrm{a,in}}(t)$, $\vartheta_{\mathrm{a,in}}(t)$, $\chi_{\mathrm{CH_4,a,in}}(t)$, $\chi_{\mathrm{H_2,a,in}}(t)$, $\lambda_{\mathrm{air}}(t)$, $\vartheta_{\mathrm{air}}(t)$, $R_{\mathrm{back}}(t)\}$[4] for the configuration of the gas flows; see again Fig. 3.

Optimal load changes will be the major task for the optimization of these complicated PDAE models. These can be acchieved by tracking the cell voltage $U_{\mathrm{cell}}(t)$ to its new stationary value $U_{\mathrm{ref}}(t)$ including control costs

$$\min \frac{1}{2}\int_0^{t_{\mathrm{f}}} (U_{\mathrm{cell}}(t) - U_{\mathrm{ref}}(t))^2 \mathrm{d}t + \frac{\lambda}{2}\sum_{u\in U}\int_0^{t_{\mathrm{f}}} u^2(t)\mathrm{d}t\,,$$

with $\lambda > 0$.[5] As constraints, the entire dynamical system has to be taken into account as well as technically motivated box constraints for each control variable $u \in U$. In addition,

[4] The values of the other molar fractions of the anode gas inlet are set to zero except for $\chi_{\mathrm{H_2O,a,in}} = 1 - \chi_{\mathrm{CH_4,a,in}} - \chi_{\mathrm{H_2,a,in}}$.

[5] Alternatively, each control could also possess its own cost factor $\lambda_i > 0$.

large variations of the solid temperature ϑ_s must be avoided, because hot spots lead finally to material corrosion. This requirement can be included, for example, by state constraints; see Sec. 3.4.2.

3.2 Semi-discretization in space as first step of FDTO

Using a semi-discretization in space, e. g., with finite differences, the PDAE model is transferred into a large system of DAEs. This method is known as (vertical) method of lines (MOL). In the following, this is exemplarily demonstrated for the 1D model with an equidistant grid $z^{(n)} = (n-1)h,\ n = 1, \ldots, N$, with stepsize $h = 1/(N-1)$ using Finite Difference Methods (FDM).[6] For the convection terms upwind schemes must be applied to preserve the underlying conservation laws. Here onesided spatial difference quotients must to be chosen against the flow direction while all coefficients must be evaluated at the current discretization point except molar flow densities, which must be evaluated at the neighboring discretization point against the flow direction. Moreover, upwind schemes must also be applied directly to the product of molar flow densities and gas temperatures. According to our computations, shocks did not appear. Note that, in the anode gas channel the characteristic curves spread out because $\frac{\partial(\gamma_\mathrm{a}\vartheta_\mathrm{a})}{\partial z} > 0$. In the cathode gas channel, however, we have a slight compression.

The integrals are approximated by quadrature formulas, e. g., by a composite trapezoidal or simpson rule with weights α_n.

The resulting spatially semi-discretized state equations constitute a DAE system in time for the discrete states $\chi_{k,\mathrm{a}}^{(n)}(t), n = 1, \ldots, N$, etc.; see Fig. 5. It is of index 1; see [5]. For solving the DAEs one relies on good choices for the initial conditions, particularly for the algebraic variables. Hereby, consistent initial conditions cannot be chosen independently for each line, since they must approximate the consistent initial conditions of the original PDAE. Good choices can be obtained from stationary solutions of the state equations: repeated simulations over sufficiently long time intervals and successive refinement of the discretization combined with interpolation on these finer grids usually yield appropriate consistent initial data.

3.3 Numerical results for the 2D cross flow configuration via $\mathrm{FD_{hk}TO_{\mathbb{N}}}$

In the thesis of Sternberg [25] models have been investigated for the 2D cross flow configuration. In particular optimal trajectories were computed for fast and safe load changes including sensitivity studies. The thesis follows the FDTO approach. After a 2D semi-discretization in space similar to Sec. 3.2, the code NUDOCCCS of Büskens [4] has been applied to solve the resulting DAE optimal control problem. Inside NUDOCCCS the sparse NLP solver SNOPT was employed; see Gill, Murray, and Saunders [9]. Note that this approach is the discrete version of the reduced functional approach of Sec. 2.1, as SNOPT is working purely on the control variables.

Due to the complexity of the system and the long process time (up to 3 h), but mainly due to the non-efficient application of numerical differentiation for the computation of the gradient of the objective function with respect to the discretized control variables only coarse discretizations can be handled. Nevertheless, the numerical results were sufficiently trustworthy thanks

[6] Adaptivity in space is not an issue here, since equidistant grids perform sufficiently well; see [24]. Thereby, one avoids the book-keeping of handling adaptive grids for each of the 21 space-dependent space variables at several time instances.

DAEs in gas channels and solid:

$$\frac{\partial \chi_{k,\mathrm{a}}^{(n)}}{\partial t} + \gamma_{\mathrm{a}}^{(n-1)}\,\vartheta_{\mathrm{a}}^{(n)}\,\frac{\chi_{k,\mathrm{a}}^{(n)} - \chi_{k,\mathrm{a}}^{(n-1)}}{h} = f^{\chi_{k,a}}\,,$$

$$\frac{\partial \chi_{k,\mathrm{c}}^{(n)}}{\partial t} - \gamma_{\mathrm{c}}^{(n+1)}\,\vartheta_{\mathrm{c}}^{(n)}\,\frac{\chi_{k,\mathrm{c}}^{(n+1)} - \chi_{k,\mathrm{c}}^{(n)}}{h} = f^{\chi_{k,c}}\,,$$

$$\frac{\partial \vartheta_{\mathrm{a}}^{(n)}}{\partial t} + \gamma_{\mathrm{a}}^{(n-1)}\,\vartheta_{\mathrm{a}}^{(n)}\,\frac{\vartheta_{\mathrm{a}}^{(n)} - \vartheta_{\mathrm{a}}^{(n-1)}}{h} = f^{\vartheta_{\mathrm{a}}}\,,$$

$$\frac{\partial \vartheta_{\mathrm{c}}^{(n)}}{\partial t} - \gamma_{\mathrm{c}}^{(n+1)}\,\vartheta_{\mathrm{c}}^{(n)}\,\frac{\vartheta_{\mathrm{c}}^{(n+1)} - \vartheta_{\mathrm{c}}^{(n)}}{h} = f^{\vartheta_{\mathrm{c}}}\,,$$

$$\frac{\gamma_{\mathrm{a}}^{(n)}\,\vartheta_{\mathrm{a}}^{(n)} - \gamma_{\mathrm{a}}^{(n-1)}\,\vartheta_{\mathrm{a}}^{(n-1)}}{h} = f^{\gamma_{\mathrm{a}}\vartheta_{\mathrm{a}}}\,,$$

$$-\frac{\gamma_{\mathrm{c}}^{(n+1)}\,\vartheta_{\mathrm{c}}^{(n+1)} - \gamma_{\mathrm{c}}^{(n)}\,\vartheta_{\mathrm{c}}^{(n)}}{h} = f^{\gamma_{\mathrm{c}}\vartheta_{\mathrm{c}}}\,,$$

$$c_{\mathrm{p,s}}\,\frac{\partial \vartheta_{\mathrm{s}}^{(n)}}{\partial t} - \frac{\vartheta_{\mathrm{s}}^{(n+1)} - 2\,\vartheta_{\mathrm{s}}^{(n)} + \vartheta_{\mathrm{s}}^{(n-1)}}{Pe \cdot h^2} = f^{\vartheta_{\mathrm{s}}}\,,$$

DAEs of burner and mixer:

$$\chi_{k,\mathrm{b}} = g^{\chi_{k,\mathrm{b}}}\,,$$

$$\vartheta_{\mathrm{b}} = g^{\vartheta_{\mathrm{b}}}\,,$$

$$\gamma_{\mathrm{b}} = g^{\gamma_{\mathrm{b}}}\,,$$

$$\gamma_{\mathrm{air}} = g^{\gamma_{\mathrm{air}}}\,,$$

$$\frac{\mathrm{d}\chi_{k,\mathrm{m}}}{\mathrm{d}t} = g^{\chi_{k,\mathrm{m}}}\,,$$

$$\frac{\mathrm{d}\vartheta_{\mathrm{m}}}{\mathrm{d}t} = g^{\vartheta_{\mathrm{m}}}\,,$$

$$\gamma_{\mathrm{m}} = g^{\gamma_{\mathrm{m}}}\,,$$

currents and the potential fields:

$$I_{\mathrm{a}}(t) = \sum_{n=1}^{N} \alpha_n \cdot i_{\mathrm{a}}^{(n)}\,, \qquad i_{\mathrm{a}}^{(n)}(z,t) = f^{i_{\mathrm{a}}}\,, \qquad \frac{\mathrm{d}U_{\mathrm{cell}}}{\mathrm{d}t}(t) = d^{U_{\mathrm{cell}}}\,,$$

$$I_{\mathrm{c}}(t) = \sum_{n=1}^{N} \alpha_n \cdot i_{\mathrm{c}}^{(n)}\,, \qquad i_{\mathrm{c}}^{(n)}(z,t) = f^{i_{\mathrm{c}}}\,, \qquad \frac{\partial \Phi_{\mathrm{a}}^{\mathrm{L}(n)}}{\partial t}(z,t) = f^{\Phi_{\mathrm{a}}^{\mathrm{L}}}\,,$$

$$I_{\mathrm{e}}(t) = \sum_{n=1}^{N} \alpha_n \cdot i_{\mathrm{e}}^{(n)}\,, \qquad i_{\mathrm{e}}^{(n)}(z,t) = f^{i_{\mathrm{e}}}\,, \qquad \frac{\partial \Phi_{\mathrm{c}}^{\mathrm{L}(n)}}{\partial t}(z,t) = f^{\Phi_{\mathrm{c}}^{\mathrm{L}}}\,,$$

$$i^{(n)}(z,t) = f^{i} + d^{i}\,.$$

Fig. 5 1D model of the MCFC after semi-discretization in space. Now the terms f^q and g^q depend on the lines associated with the state variables combined by Z, e. g., $\vartheta_{\mathrm{a}}^{(n)}(t)$, $n = 1, \ldots, N$, etc.. Analogously, the boundary lines enter $g^q = g^q(\vartheta_{\mathrm{a}}^{(N)}(t), \ldots, \vartheta_{\mathrm{c}}^{(1)}(t), \ldots)$. Here the index range is $n = 2, \ldots, N$ in the anode, $n = 1, \ldots, N-1$ in the cathode, and $n = 2, \ldots, N-1$ for ϑ_{s}, since the boundary lines are omitted here. They differ according to their boundary conditions. For more details see [23]. Finally, the coefficients α_n are the weights of the quadrature formulas.

to the smoothing influence of the heat equation. However, the amount of computation went up to days depending on the fineness of the grid.

Figure 6 shows the optimization of a fast load change from $I_{\mathrm{cell}} = 0.7$ to $I_{\mathrm{cell}} = 0.6$. Hereby, the stationary value of the cell voltage U_{cell} associated with the new load is tracked in the L^2-norm; see Sec. 3.1. Because of the additivity of the integral functional the entire

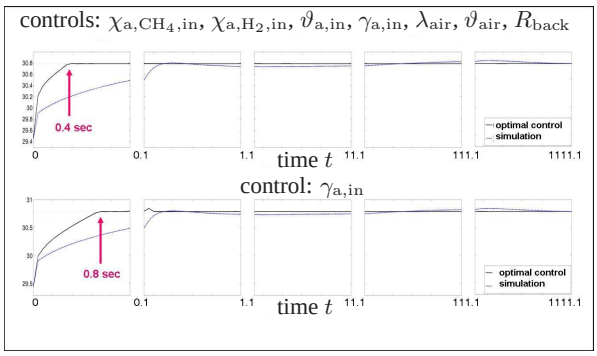

Fig. 6 Fast load change by tracking the cell voltage using either the full set of controls (top) or solely the molar flow density $\gamma_{\mathrm{a,in}}$ (bottom) at anode inlet; cf. [25].

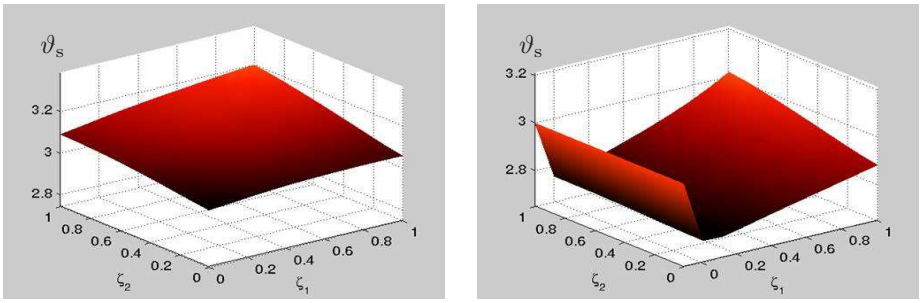

Fig. 7 Left: Solid temperature ϑ_{s} at final time [25]. Note the gas flow directions: in the anode from $\zeta_1 = 0$ to $\zeta_1 = 1$, in the cathode from $\zeta_2 = 1$ to $\zeta_2 = 0$. A hotspot occurs at the anode outlet, respectively at the cathode inlet, paricularly at $\zeta_1, \zeta_2) = (1, 0)$ where the gas stream takes a short cut due to the cross flow configuration. Right: Anode gas temperature ϑ_{a} at final time [25]. In flow direction ζ_1, first the endothermic reforming, then the exothermic oxidation dominates the temperature distribution.

time interval can be partioned in a pseudo-logarithmic scale to compensate for the dynamical behaviour of fast variables at the beginning of the process. The optimization was then performed on the resulting subintervals in a suboptimal manner to reduce the computational effort caused by the long process time. Since the cell voltage reacts relatively fast, the new stationary point of the MCFC can be achieved in less than 1 sec of process time. Moreover it turns out that the inflow $\gamma_{\mathrm{a,in}}$ of the molar flow density alone is sufficient for controlling fast load changes. Figure 7 depicts the solid temperature ϑ_{s} and the gas temperature ϑ_{a} in the anode gas channel at the final time t_{f}.

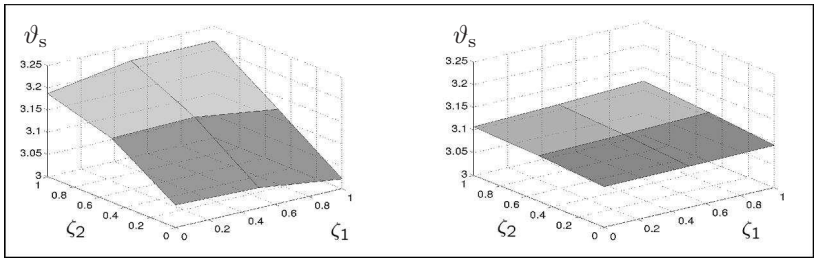

Fig. 8 Solid temperature ϑ_s at final time [6]. When choosing the aforementioned objective functional there is no necessity to pose pointwise state constraints on the solid temperature (right) in contrast to the choice $\varepsilon = 0$ for the entire time interval (left).

More results can be found in [25] and [6]. In the latter, fast and *safe* load changes have been obtained by modifying the objective functional

$$(P_\varepsilon) \quad \min \int_{t_0}^{t_f} \big(1-\varepsilon(t)\big)\left(U_{\text{cell}}(t)-U_{\text{ref}}\right)^2 + \varepsilon(t)\left(\iint_\Omega \left(\vartheta_s(\zeta_1,\zeta_2,t)-\vartheta_{\text{ref}}\right)^2 \mathrm{d}\zeta_1\,\mathrm{d}\zeta_2\right)\mathrm{d}t$$

with

$$\varepsilon = 0 \text{ on } \begin{cases} [0.0,0.1] \\ [0.1,1.1] \\ [1.1,11.1] \\ [11.1,111.1] \end{cases} \text{ and } \varepsilon = 1 \text{ on } [111.1, 1111.1]\,.$$

Herewith the fast reacting cell voltage U_{cell} is initially to be tracked towards $U_{\text{cell}}^{\text{to-be}}$ and, at the end, the functional shall switch to the slowly varying solid temperature ϑ_s. This choice avoids too large and therefore risky variations in the solid temperature without employing a pointwise state constraint on it or, more suitably, a constraint on the gradient of the solid temperature which constitutes quite a challenge for such a complicated model; see Sec. 3.4.2. For the numerical results of this scalarized Parato optimal control problem, see Fig. 8.

Despite the partition into a few time intervals we can still treat only coarse discretizations in time and space, which is unsatisfactory from the viewpoint of numerical analysis. Hence we will now investigate different optimization methods that either allow for adaptive time discretization and much more time steps, or avoid the numerical differentiation to reduce the number of PDAE solves drastically. In order to compare different approaches within acceptable computing times, we restrict ourselves to the 1D model from now on.

3.4 Numerical results for the 1D counter flow configuration via $\mathrm{FD}_{\mathrm{hk}}\mathrm{TO}_{\mathbb{N}}$

The simplified 1D version of the MCFC model allows us to compare several optimization methods on standard desktop PCs. In the following, we will compare three different methods: FDTO working on the reduced objective, FDTO working on the full objective using automatic differentiation (AD) and FOTD again for the reduced objective.

3.4.1 Method 1: $\mathrm{FD}_{\mathrm{h}}\mathrm{D}_{\mathrm{k}}\mathrm{TO}_{\mathbb{N}}$

First, we investigate a method which again aims at solving the reduced objective by using time stepping methods after a semi-discretization in space as described before. In [24] the resulting DAE system for the semi-discretized state variables of Fig. 5 has been solved by means of a fully implicit variable-order adaptive-time-stepping stiff BDF-method. The arising nonlinear systems are solved by a simplified Newton method, where the Jacobians are reused in neighbouring time steps. As suggested before, the integration routine here plays the role of a discrete analogue of the control-to-state operator.

Replacing the fully implicit scheme by semi-implicit methods in order to save computing time did not work here due to instabilities. Though, one could investigate decoupling methods or account for the different time scales, but we do not follow this approach here.

The implementation was done in MATLAB with the Quasi-Newton method `fmincon` as optimization routine. Note that the only optimization variables here are the few discretized boundary control variables which only depend on time. Each call of the objective function requires a DAE solve for the semi-discretized state system. The gradient of the reduced objective function is obtained by numerical differentiation inside `fmincon`.

The state solver turns out to be reliable and efficient, since the stepsize control accounts for the different time scales appropriately; see the error analysis in [24]. Nevertheless, the optimization routine gets lost if we release the controls so that they can be independently chosen at each of the approximately 1000 time steps, at which the states are computed. Because of many jumps in the control variables instabilities would occur. However, we can exploit that the optimal control has a higher regularity. Therefore, we choose linear splines as ansatz space for the controls and a coarser control grid with only 81 equidistant time steps. Herewith, optimal tracking of the cell voltage for a load change of from $I_{\mathrm{cell}} = 0.4$ to $I_{\mathrm{cell}} = 0.5$ at $t_s = 0.1$ can be carried through using only one of the control variables, namely the temperature of the air inlet $\vartheta_{\mathrm{air}}(t)$; see Fig. 9. We see, that the new stationary state is reached approximately at $t = 240$ compared to $t = 1000$ for the uncontrolled case. For more results obtained by this methods, for example using all possible control variables, see [24].

As advantage we note, that `fmincon` has to deal only with a few time-dependent controls, instead of many space- and time-dependent state variables in case of a full discretization (see Sec. 3.4.2). Furthermore, the state solver is very efficient and precise due to adaptive time-stepping methods and the potential of taking many time steps. In contrast, the numerical differentiation is inefficient and inexact: each gradient evaluation needs M function evaluations of the objective, which each requires one nonlinear state solve. This leads to poor convergence behavior and considerable computing times.

There are several remedies to cope with this. For the short term, the convergence behavior can be enhanced by alternating steps of optimizing and refinement of the control discretization or by methods of instantaneous control (see, e. g., [15]). Especially receding horizon methods with small horizons (1 or 2 intervals) yield good initial guesses in our computations with low

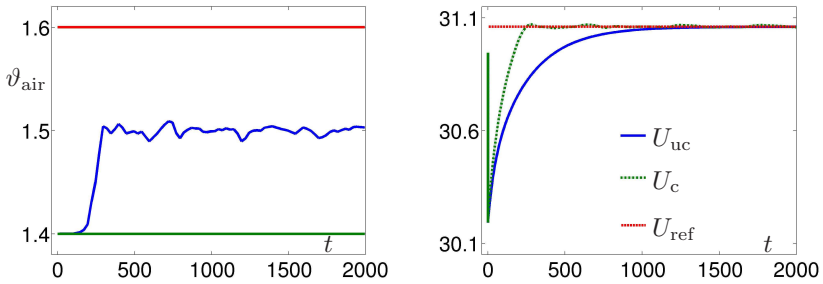

Fig. 9 Optimal tracking of the cell voltage for the 1D model for a load change from $I_{\mathrm{cell}} = 0.4$ to $I_{\mathrm{cell}} = 0.5$ at $t_s = 0.1$ using only the air temperature $\vartheta_{\mathrm{air}}(t)$ as control variable. Optimal control (left) and the transient oszillation of $U_{\mathrm{cell}}(t)$ to the desired state U_{ref} in the uncontrolled (uc), respectively the controlled case (c) (right). Data used: $N = 41$ lines, $t_{\mathrm{f}} = 2000$ and $\lambda = 10^{-5}$.

amount of computation. In the long term, one should avoid the numerical differentiation by applying automatic differentiation (AD) or adjoint calculus (FOTD).

A change to automatic differentiation (AD), for example with ADiMat of Bischof et. al. [1] based on source transformation, is very involved for adaptive implicit state solver. The first step, the computation of the Jacobians of the implicit state solver can easily be done, but the second step, the computation of the gradient of the reduced objective is challanging. It involves not only a second differentiation of a certain part of the code (for the computation of the Jacobians), which currently is not possible in MATLAB, but also a differentiation of an adaptive solver, which cannot be done using an AD tool as black-box. Instead, we will provide the gradient by adjoint calculus in Sec. 3.5. For a black-box use of AD we have to change our framework significantly, which we discuss next.

3.4.2 Method 2: $\mathrm{FD}_{\mathrm{hk}}\mathrm{TO}_{\mathbb{N}}$

In the following, we alternatively investigate a method, that allows a comfortable black-box use of AD for the first and second derivatives. Therefore, we will control our preferred NLP solver via the modelling language AMPL of Fourer et. al. [7] in order to benefit from exact first and second order derivative information that is provided by AMPL. For that purpose, we have to change two major ingredients: firstly, we must iterate on the state and the control variables and hence, we apply a full simultaneous discretization in space and time for the state equations. Secondly, we have to drop adaptive grid techniques. Therefore, we employ a fixed, pseudo-adaptive, logarithmically scaled time grid that reflects the different time scales of the variables with special emphasis on the initial time and the time of the load change, cf. the results in [24]. We stick to the equidistant spatial grid, that already performed well in former investigations. The spatial discretization follows the lines of Sec. 3.2, which yields a semi-explicit stiff DAE system of index 1. Hence we have to use an implicit time discretization, here the implicit Euler scheme.

Again we are led to a now even larger NLP which can, in principle, be solved by high performance NLP solvers such as the interior point solver IPOPT of Wächter and Biegler [28]. Note that this approach is now equivalent to the numerical approximation of the non-reduced

optimization problem. Control and state variables play an equal role here. Indeed we have to deal with 23 spatially and time dependent variables and 18 only time dependent variables. On the other hand, an advantage of this approach is that pointwise inequality constraints can be easily added. Also changes of the model can be more easily accomplished, which is a frequent task in interdisciplinary cooperations.

In contrast to Method 1, trajectories at fixed spatial points are no longer absolute continuous functions in time now. Moreover, it turns out that the determination of initial conditions leading to convergence is quite a challenge here, although IPOPT is said to be able to deal with non-admissible initial guesses. For details see [28].

By the following strategy, this hurdle can be overcome in many, but not all situations. First compute a stationary solution on the entire space-time cylinder by Method 1 and store the values of all state and control variables on the space-time grid. This yields initial data which may lead to convergence. If the objective functional $(P_{\varepsilon=0})$ (tracking of the cell voltage) is minimized, this generally works quite well. However, the functional $(P_{\varepsilon=1})$ (tracking of the solid temperature) can generally not be treated this way. Since the solid temperature is a slow variable, the functional $(P_{\varepsilon=1})$ is extremely insensitive for the chosen time horizon with respect to variations in input data such as the discretized controls. ¿From these sensitivities, we however have to extract the necessary information to propel the optimization iteration. Maybe, a similar homotopy technique as in Sec. 3.3 could help here, too.

Figure 10 shows numerical results for a load change from $I_{\mathrm{cell}} = 0.4$ to $I_{\mathrm{cell}} = 0.45$ at $t = 10$. Here the functional $(P_{\varepsilon=0})$ is taken including the regularisation term with parameter $\lambda = 0.001$. The figure depicts components of the optimal solution without respectively with a pointwise constraints on the solid temperature. The controls at the anode as well as the feedback rate are effected by the state constraint considerably while the influence of the state constraint on the solid temperature remains moderate. Despite the reduction of the solid temperature the maximal gradients increase slightly.

Further results obtained by Method 2 are summarized in the following from [18]. To acchieve convergence it is usually necessary to set bounds on the optimization variables to keep the iteration process in a physically resonable region or to guarantee that all terms can be evaluated. Moreover, it turns out that convergence is easierly obtained if all possible controls are considered in the optimization iteration, although only two controls ($\vartheta_{\mathrm{a,in}}$,$\gamma_{\mathrm{a,in}}$) are actually "used", i. e., they are not kept on its lower bounds caused by the regularisation term. The solid temperature can be bounded from below up to a level of 3.77.

Finally, it turns out that storage requirements were not the bottleneck, but the amount of computation which may run up to 32 h for an optimization iteration with up to 34,000 variables (4 GiB RAM, 32-bit). Due to the frequent search for appropriate initial data, this entails also a considerable amount of human resources.

When taking into account a constraint on the gradient of the solid temperature, Method 2 did not converge. The discretization does not allow a sufficiently precise computation of the gradient; higher order differences schemes should be applied then.

The black-box use of AD is very comfortable and AMPL exempts the (unaware) user from providing gradient information, e. g., via adjoint equations. Therefore, this approach reduces, at least at the first glance, human resources considerably. On the other side, it is often very difficult to acchieve convergence for large PDAE models, especially when state constraints are involved. Nevertheless, method 2 is the only method that allows a direct inclusion of state constraints. Therefore, especially smaller (diffusion dominated) PDAE models with state

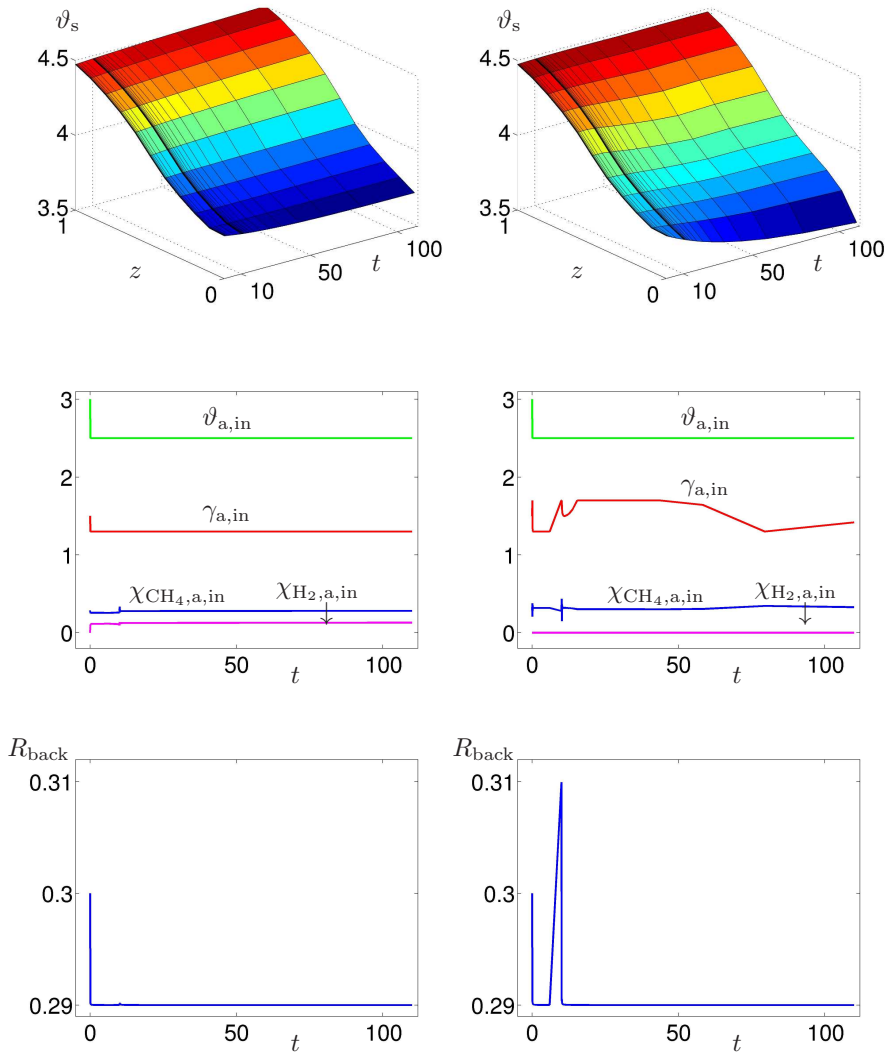

Fig. 10 Solid temperature (top), controls at anode inlet (middle), and feedback rate (bottom) for the state-unconstrained problem (left column) and the problem with the pointwise state constraint $\vartheta_s(z,t) \leq 4.48$ (right column), cf. [18].

constraints can be solved comfortably and with small implementational effort, e. g. the rocket car problem in [29].

We now proceed to an adjoint based approach which is known to be the most efficient and precise approach in general but also requires the highest level of skills of the user and of

preliminary work. Indeed, the complexity of the problem constitutes a severe challenge for FOTD.

3.5 Numerical results for the 1D counter flow configuration via $\mathrm{FO}_\infty\mathrm{TD}_{\mathrm{hk}}$

Optimization methods working on the reduced objective like the one in Sec. 3.4 benefit from two factors. Firstly, they iterate only on the control variables, which are typically only some time-dependent functions in optimal control problems arising from applications. Secondly, if the state equations require many or adaptive time steps in their integration due to different time scales or long process times, time stepping methods can handle this efficiently. In both cases, working on the reduced objective involves a much smaller number of optimization variables than iterating on all discretized state and control variables like in Sec. 3.4.2.

The fuel cell model at hand exhibits both properties. This motivates to enhance the FDTO method from Sec. 3.4.1. Its drawback is the inexact and costly computation of the gradient information by numerical differentiation. Since the use of AD is not an option as already mentioned, we will now strive for a precise and inexpensive way to compute the gradient information. We employ the adjoint calculus, hence turn to the FOTD approach. This has recently been applied to the 1D counter flow model of Fig. 4 in [23]. Thereby we are able to compute the gradient via one linear PDAE solve compared to M nonlinear PDAE solves using numerical differentiation. Here, M denotes the number of control variables. This reduces the amount of computation considerably while the gradient information can be computed more precisely. Therefore, we can increase the number of control variables, since it is not a bottleneck anymore. On the other side, a disadvantage is that we now have to derive the adjoint equations of that very complicated PDAE model with up to 28 equations. Therefore, we propose a structured approach and the use of AD tools for subtasks, otherwise the effort and the risk of errors may explode.

The main idea how to derive adjoint equations of such complicated multiphysics models can be carried through as follows: calculate only the differential operators and the coupling structure by analytical means, i. e., by hands, and then apply AD tools to obtain the many and lengthy nonlinear source terms. Since the differential operators are only short expressions, we are indeed able to derive the adjoint equations by applying the formal Lagrange technique, if we introduce appropriate abbreviations for all the source terms. In order to demonstrate the modus operandi, the starting point, the Lagrangian, is given here explicitely:[7]

[7] To avoid some product rules in the derivation of the optimality conditions, we replace the molar flow rates γ by the velocities $v := \gamma\,\vartheta$.

$$
\begin{aligned}
\mathcal{L}(...) = J &- \sum_{k\in\mathcal{I}} \iint\limits_{Q} \left(\frac{\partial \chi_{k,\mathrm{a}}}{\partial t} + v_{\mathrm{a}} \frac{\partial \chi_{k,\mathrm{a}}}{\partial z} - f^{\chi_{k,\mathrm{a}}} \right) \tilde{\chi}_{k,\mathrm{a}} \mathrm{d}z\mathrm{d}t \\
&- \int_0^{t_\mathrm{f}} \sum_{k\in\mathcal{I}} \Big(\chi_{k,\mathrm{a}}(0,t) - \chi_{k,\mathrm{a,in}} \Big) \hat{\chi}_{k,\mathrm{a}} \mathrm{d}t \\
&- \sum_{k\in\mathcal{I}} \iint\limits_{Q} \left(\frac{\partial \chi_{k,\mathrm{c}}}{\partial t} - v_{\mathrm{c}} \frac{\partial \chi_{k,\mathrm{c}}}{\partial z} - f^{\chi_{k,\mathrm{c}}} \right) \tilde{\chi}_{k,\mathrm{c}} \mathrm{d}z\mathrm{d}t \\
&- \int_0^{t_\mathrm{f}} \sum_{k\in\mathcal{I}} \Big(\chi_{k,\mathrm{c}}(1,t) - \chi_{k,\mathrm{m}} \Big) \hat{\chi}_{k,\mathrm{c}} \mathrm{d}t \\
&- \iint\limits_{Q} \left(\frac{\partial \vartheta_{\mathrm{a}}}{\partial t} + v_{\mathrm{a}} \frac{\partial \vartheta_{\mathrm{a}}}{\partial z} - f^{\vartheta_{\mathrm{a}}} \right) \tilde{\vartheta}_{\mathrm{a}} \mathrm{d}z\mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(\vartheta_{\mathrm{a}}(0,t) - \vartheta_{a,\mathrm{in}} \Big) \hat{\vartheta}_{\mathrm{a}} \mathrm{d}t \\
&- \iint\limits_{Q} \left(\frac{\partial \vartheta_{\mathrm{c}}}{\partial t} - v_{\mathrm{c}} \frac{\partial \vartheta_{\mathrm{c}}}{\partial z} - f^{\vartheta_{\mathrm{c}}} \right) \tilde{\vartheta}_{\mathrm{c}} \mathrm{d}z\mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(\vartheta_{\mathrm{c}}(1,t) - \vartheta_{\mathrm{m}} \Big) \hat{\vartheta}_{\mathrm{c}} \mathrm{d}t \\
&- \iint\limits_{Q} \left(\frac{\partial v_{\mathrm{a}}}{\partial z} - f^{v_{\mathrm{a}}} \right) \tilde{v}_{\mathrm{a}} \mathrm{d}z\mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(v_{\mathrm{a}}(0,t) - v_{\mathrm{a,in}} \Big) \hat{v}_{\mathrm{a}} \mathrm{d}t \\
&- \iint\limits_{Q} \left(-\frac{\partial v_{\mathrm{c}}}{\partial z} - f^{v_{\mathrm{c}}} \right) \tilde{v}_{\mathrm{c}} \mathrm{d}z\mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(v_{\mathrm{c}}(1,t) - \gamma_{\mathrm{m}} \vartheta_{\mathrm{m}} \Big) \hat{v}_{\mathrm{c}} \mathrm{d}t \\
&- \iint\limits_{Q} \left(c_{\mathrm{p,s}} \frac{\partial \vartheta_{\mathrm{s}}}{\partial t} - \frac{1}{\mathrm{Pe}} \frac{\partial^2 \vartheta_{\mathrm{s}}}{\partial z^2} - f^{\vartheta_{\mathrm{s}}} \right) \tilde{\vartheta}_{\mathrm{s}} \mathrm{d}z\mathrm{d}t - \frac{1}{\mathrm{Pe}} \int_0^{t_\mathrm{f}} \left[\frac{\partial \vartheta_s}{\partial z} \hat{\vartheta}_s \right]_0^1 \mathrm{d}t \\
&- \sum_{k\in\mathcal{I}} \int_0^{t_\mathrm{f}} \Big(\chi_{k,\mathrm{b}} - g^{\chi_{k,\mathrm{b}}} \Big) \tilde{\chi}_{k,\mathrm{b}} \mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(\vartheta_{\mathrm{b}} - g^{\vartheta_{\mathrm{b}}} \Big) \tilde{\vartheta}_{\mathrm{b}} \mathrm{d}t \\
&- \int_0^{t_\mathrm{f}} \Big(\gamma_{\mathrm{b}} - g^{\gamma_{\mathrm{b}}} \Big) \tilde{\gamma}_{\mathrm{b}} \mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(\gamma_{\mathrm{air}} - g^{\gamma_{\mathrm{air}}} \Big) \tilde{\gamma}_{\mathrm{air}} \mathrm{d}t \\
&- \sum_{k\in\mathcal{I}} \int_0^{t_\mathrm{f}} \left(\frac{\mathrm{d}\chi_{k,\mathrm{m}}}{\mathrm{d}t} - g^{\chi_{k,\mathrm{m}}} \right) \tilde{\chi}_{k,\mathrm{m}} \mathrm{d}t - \int_0^{t_\mathrm{f}} \Big(\gamma_{\mathrm{m}} - g^{\gamma_{\mathrm{m}}} \Big) \tilde{\gamma}_{\mathrm{m}} \mathrm{d}t \\
&- \int_0^{t_\mathrm{f}} \left(\frac{\mathrm{d}\vartheta_{\mathrm{m}}}{\mathrm{d}t} - g^{\vartheta_{\mathrm{m}}} \right) \tilde{\vartheta}_{\mathrm{m}} \mathrm{d}t - \int_0^{t_\mathrm{f}} \left(\frac{\mathrm{d}U_{\mathrm{cell}}}{\mathrm{d}t} - d^{u_{\mathrm{cell}}} \right) \tilde{U}_{\mathrm{cell}} \mathrm{d}t \\
&- \iint\limits_{Q} \Big(i - f^i - d^i \Big) \tilde{i} \mathrm{d}t\mathrm{d}z - \sum_{k\in\{\mathrm{a,c,e}\}} \iint\limits_{Q} \Big(i_k - f^{i_k} \Big) \tilde{i}_k \mathrm{d}t\mathrm{d}z \\
&- \sum_{k\in\{\mathrm{a,c,e}\}} \int_0^{t_\mathrm{f}} \left(I_k - \int_0^1 i_k \mathrm{d}z \right) \tilde{I}_k \mathrm{d}t \\
&- \iint\limits_{Q} \left(\frac{\partial \Phi_a^L}{\partial t} - f^{\Phi_a^L} \right) \tilde{\Phi}_a^L \mathrm{d}t\mathrm{d}z - \iint\limits_{Q} \left(\frac{\partial \Phi_c^L}{\partial t} - f^{\Phi_c^L} \right) \tilde{\Phi}_c^L \mathrm{d}t\mathrm{d}z \,.
\end{aligned}
$$

Note that each equation (PDE, ODE, AE) respectively each boundary condition has its own Lagrange multiplier, marked by a tilde respectively a hat. We obtain a total number of 67 multipliers. Hence, the tildes respectively hats indicate Lagrange multipliers associated either with state variables governed by PDEs or other types of equations, or respectively with boundary conditions. Then partial integrations with respect to time and space and directional derivatives with respect to the state variables yield the differential operators for the adjoint equations. This can be done easily analytically. For the linearization of the nonlinear source terms f^q, g^q and d^q, however, one should employ symbolic or automatic differentiation since more than 100 partial derivatives of complicated functions (e. g., due to the reaction kinetics) have to be provided. Compare the adjoint equation of the semi-linear model of Sec. 2.1. Additional partial integrations of the Lagrangian may finally alleviate to read out the adjoint system by means of standard variational arguments; compare again the example in Sec. 2.1.

The resulting adjoint equations are depicted in Fig. 11. Their source terms are now abbreviated by $\tilde{f}^w := \sum_{q \in Z_v} \frac{\partial f^q}{\partial q}\, \tilde{q}(z,t)$ with $Z_v := \{\vartheta_\mathrm{a},\ \vartheta_\mathrm{c},\ \chi_{k,\mathrm{a}},\ \chi_{k,\mathrm{c}},\ \vartheta_\mathrm{s},\ v_\mathrm{a},\ v_\mathrm{c},\ i_\mathrm{a},\ i_\mathrm{c},\ i_\mathrm{e},\ i,\ \Phi_\mathrm{a}^\mathrm{L},\ \Phi_\mathrm{c}^\mathrm{L}\}$ collecting all space-dependent state variables. Analogously we define $\tilde{g}^w := \sum_{q \in B_v} \frac{\partial g^q}{\partial q}\, \tilde{q}$, where the set B_v contains the state variables involved in the DAE of the burner and mixer. Note that some of the $\tilde{g}^w$ also depend on control variables due to products of control and state variables in the corresponding state equations. This gives rise to the dependency $ADJ(p) = C(y,u)$ in the description of the FOTD approach in Sec. 2.1.

In contrast to the semi-linear problem of Sec. 2.1, we now have to deal with quasi-linear reaction advection equations, too. Their differential operator is not self-adjoint, so we get more terms on the left side due to the product rule e. g., $\frac{\partial}{\partial z}\,(v\,\tilde{\chi}) = v\,\frac{\partial \tilde{\chi}}{\partial z} + \frac{\partial v}{\partial z}\tilde{\chi}$. Moreover, the state equations involve ODE-PDE couplings. Therefore, the formal Lagrange technique leads to integral expressions respectively spatially constant source terms in the adjoint currents $\tilde{I}$ respectively the adjoint current densities $\tilde{i}$. Furthermore, each adjoint variable has homogeneous terminal conditions. The coupling structure of the boundary conditions turns back as shown in Fig. 12, i. e., the adjoint anode gas channel has now Dirichlet boundary conditions at $z = 1$ instead at $z = 0$. The detailed adjoint system can be found in [23].

Note that the depicted adjoint equations are still incomplete. For the purpose of optimization, the derivative of the objective functional with respect to the state variables has to be added as inhomogenity; compare Sec .2.1. In case of tracking the cell voltage, an inhomogeneous term $U_\mathrm{cell}(t) - U_\mathrm{ref}(t)$ appears in the right hand side of the ODE for $\tilde{U}_\mathrm{cell}$.

The optimality system is completed by a set of nonlinear variational inequalities, see [23] and Fig. 12. Since some control variables enter the state equations nonlinearly and at two locations, these variational inequalities are much more involved than those of Sec. 2.1: some are fully coupled and can therefore not be formulated as projection formulas anymore. Fortunately, our implementation does not have to solve these complicated variational inequalities, instead they can be checked after convergence. However, we can identify the gradient of the reduced objective analogously to Sec. 2.1 by differentiating the Lagrangian with respect to the controls.

The adjoint equations can generally be solved analogeously to the state equations, since the type of equations stays unchanged. We have to modify the MATLAB-implementation used in Sec. 3.4 only slightly. However, we have to obey the change of the flux directions and the contrariwise coupling structure of the boundary conditions. The adaptive BDF-method must now run backward in time. We note, that since the state and adjoint system is solved independently

PDEs in gas channels and solid:

$$
\begin{aligned}
-\frac{\partial \tilde{\chi}_{k,\mathrm{a}}}{\partial t} - v_{\mathrm{a}} \frac{\partial \tilde{\chi}_{k,\mathrm{a}}}{\partial z} - \frac{\partial v_{\mathrm{a}}}{\partial z}\, \tilde{\chi}_{k,\mathrm{a}} &= \tilde{f}^{\chi_{k,a}}\,, \\
-\frac{\partial \tilde{\chi}_{k,\mathrm{c}}}{\partial t} + v_{\mathrm{c}} \frac{\partial \tilde{\chi}_{k,\mathrm{c}}}{\partial z} + \frac{\partial v_{\mathrm{c}}}{\partial z}\, \tilde{\chi}_{k,\mathrm{c}} &= \tilde{f}^{\chi_{k,c}}\,, \\
-\frac{\partial \tilde{\vartheta}_{\mathrm{a}}}{\partial t} - v_{\mathrm{a}} \frac{\partial \tilde{\vartheta}_{\mathrm{a}}}{\partial z} - \frac{\partial v_{\mathrm{a}}}{\partial z}\, \tilde{\vartheta}_{\mathrm{a}} &= \tilde{f}^{\vartheta_{\mathrm{a}}}\,, \\
-\frac{\partial \tilde{\vartheta}_{\mathrm{c}}}{\partial t} + v_{\mathrm{c}} \frac{\partial \tilde{\vartheta}_{\mathrm{c}}}{\partial t} + \frac{\partial v_{\mathrm{c}}}{\partial z}\, \tilde{\vartheta}_{\mathrm{c}} &= \tilde{f}^{\vartheta_{\mathrm{c}}}\,, \\
-\frac{\partial \tilde{v}_{\mathrm{a}}}{\partial z} &= \tilde{f}^{\gamma_{\mathrm{a}}\vartheta_{\mathrm{a}}}\,, \\
\frac{\partial \tilde{v}_{\mathrm{c}}}{\partial z} &= \tilde{f}^{\gamma_{\mathrm{c}}\vartheta_{\mathrm{c}}}\,, \\
-c_{\mathrm{p,s}} \frac{\partial \tilde{\vartheta}_{\mathrm{s}}}{\partial t} - \frac{1}{\mathrm{Pe}} \frac{\partial^2 \tilde{\vartheta}_{\mathrm{s}}}{\partial z^2} &= \tilde{f}^{\vartheta_{\mathrm{s}}}\,,
\end{aligned}
$$

DAEs of burner and mixer:

$$
\begin{aligned}
\tilde{\chi}_{k,\mathrm{b}} &= \tilde{g}^{\chi_{k,\mathrm{b}}}\,, \\
\tilde{\vartheta}_{\mathrm{b}} &= \tilde{g}^{\vartheta_{\mathrm{b}}}\,, \\
\tilde{\gamma}_{\mathrm{b}} &= \tilde{g}^{\gamma_{\mathrm{b}}}\,, \\
\tilde{\gamma}_{\mathrm{air}} &= \tilde{g}^{\gamma_{\mathrm{air}}}\,, \\
-\frac{\mathrm{d}\tilde{\chi}_{k,\mathrm{m}}}{\mathrm{d}t} &= \tilde{g}^{\chi_{k,\mathrm{m}}}\,, \\
-\frac{\mathrm{d}\tilde{\vartheta}_{\mathrm{m}}}{\mathrm{d}t} &= \tilde{g}^{\vartheta_{\mathrm{m}}}\,, \\
\tilde{\gamma}_{\mathrm{m}} &= \tilde{g}^{\gamma_{\mathrm{m}}}\,,
\end{aligned}
$$

currents and the potential fields:

$$
\begin{aligned}
\tilde{I}_{\mathrm{a}}(t) &= \int_0^1 \frac{\partial d^i}{\partial I_{\mathrm{a}}}\,\mathrm{d}z + \frac{\partial d^{U_{\mathrm{cell}}}}{\partial I_{\mathrm{a}}}\,\tilde{U}_{\mathrm{cell}}\,, & \tilde{i}_{\mathrm{a}}(z,t) &= \tilde{f}^{i_{\mathrm{a}}} + \tilde{I}_{\mathrm{a}}(t)\,, \\
\tilde{I}_{\mathrm{c}}(t) &= \int_0^1 \frac{\partial d^i}{\partial I_{\mathrm{c}}}\,\mathrm{d}z + \frac{\partial d^{U_{\mathrm{cell}}}}{\partial I_{\mathrm{c}}}\,\tilde{U}_{\mathrm{cell}}\,, & \tilde{i}_{\mathrm{c}}(z,t) &= \tilde{f}^{i_{\mathrm{c}}} + \tilde{I}_{\mathrm{c}}(t)\,, \\
\tilde{I}_{\mathrm{e}}(t) &= \int_0^1 \frac{\partial d^i}{\partial I_{\mathrm{e}}}\,\mathrm{d}z + \frac{\partial d^{U_{\mathrm{cell}}}}{\partial I_{\mathrm{e}}}\,\tilde{U}_{\mathrm{cell}}\,, & \tilde{i}_{\mathrm{e}}(z,t) &= \tilde{f}^{i_{\mathrm{e}}} + \tilde{I}_{\mathrm{e}}(t)\,, \\
-\frac{\mathrm{d}\tilde{U}_{\mathrm{cell}}}{\mathrm{d}t}(t) &= \int_0^1 \tilde{f}^{U_{\mathrm{cell}}}\mathrm{d}z\,, & \tilde{i}(z,t) &= \tilde{f}^{i}\,, \\
-\frac{\partial \tilde{\Phi}_a^L}{\partial t}(z,t) &= \tilde{f}^{\Phi_{\mathrm{a}}^{\mathrm{L}}}\,, & -\frac{\partial \tilde{\Phi}_c^L}{\partial t}(z,t) &= \tilde{f}^{\Phi_{\mathrm{c}}^{\mathrm{L}}}\,.
\end{aligned}
$$

Fig. 11 Adjoint equations of the 1D MCFC model. The nonlinear source terms are abbreviated by $\tilde{f}^w := \sum_{q \in Z_v} \frac{\partial f^q}{\partial q}\,\tilde{q}$ with $w \in Z_v$ standing for the corresponding state variable. Analogously, $\tilde{g}^w := \sum_{q \in B_v} \frac{\partial g^q}{\partial q}\,\tilde{q}$. Each equation of the molar fractions $\chi_{k,\cdot}$ appears sevenfold indexed by $k \in \mathcal{I}$. There are integrals from ODE-PDE couplings induced by d^i and by $U_{\mathrm{cell}}(t)$ entering the space dependent source terms f^q.

by an adaptive solver, the resulting gradient is not the exact discrete gradient. Nevertheless, if the optimization method can handle inexact gradients and if one refines the grids of state and adjoint variables suffiently often, convergence will be maintained; see e. g. [30].

With a state solver and an adjoint solver, we can set up efficient optimization routines (e. g., Quasi-Newton methods) applied to the reduced objective. The two subsystems, state and adjoint system, are integrated successively at each function call of the objective similar

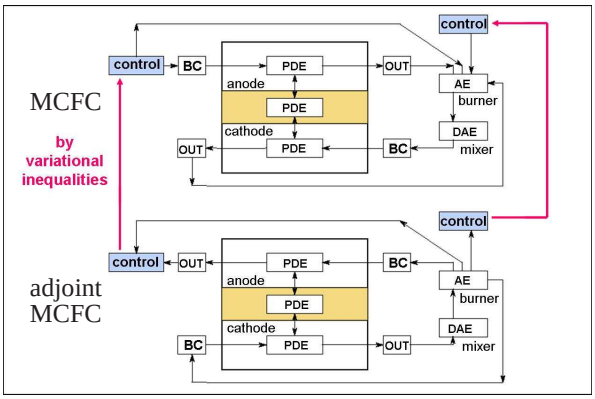

Fig. 12 Coupling structure of state and adjoint system; cf. [23]. Note that the controls enter both the anode inlet and the burner (top). Since the flux directions and the coupling of the boundary conditions (BC) are reversed in the adjoint system (bottom), the anode "outlet" as well as the burner "oulet" enter the state system (bottom) in turn twofold via complicated variational inequalities.

to the well-known backward sweep method of the early days of ODE optimal control; see Mc Reynolds and Bryson [21]. Thereafter, the gradient of the (reduced) objective is computed. This allows the application of all gradient-based optimization methods. This framework is especially efficient if many time steps are to be performed. In contrast, Newton or SQP methods (see e. g. [16]) seem to be hardly applicable due to their necessity of providing second order information (which would require a book keeping of two more such PDAE systems).

With this FOTD-method we will now optimize the transition of the fuel cell for load change in I_{cell} from 0.4 to 0.5 at $t = 0.1$ by controlling exemplarily solely the temperature at the air inlet $\vartheta_{\mathrm{air}}(t)$. We choose the semi-discretization of Fig. 5 with $N = 41$ lines. Since `fmincon` needs a fixed number of optimization variables, we construct a fixed pseudo-adaptive control grid as the junction of the two adaptive time grids from state and adjoint solver evaluated at the initial guess for the control. Since the solvers scale linearly with the number of time steps, we can use stricter tolerances, resulting in a control grid of 767 time steps. The time stepping of the state and adjoint solver is done adaptively with the control grid included to account for events in the right hand side., see [23]. For the optimization, a Quasi-Newton method with BFGS updates is employed which does not rely on second order information. We compare two cases differing in the cost factor λ of the controls ($\lambda = 5 \cdot 10^{-5}$ vs. $\lambda = 5 \cdot 10^{-2}$). Figure 13 depicts the resulting optimal cell coltage $U_{\mathrm{cell,opt}}$ for both cases compared to the desired cell voltage U_{ref} and the uncontrolled case $U_{\mathrm{cell,uc}}$. We see, that solely controlling $\vartheta_{\mathrm{air}}(t)$ with small control costs already allows a fast transition to the new stationary state: it is reached at $t \approx 80$ compared to $t \approx 1000$ for the uncontrolled case, which is also a significant improvement over the results from the FDTO method in Fig. 9 (where the control costs were even slightly smaller). The second study with larger control costs $\lambda = 5 \cdot 10^{-2}$ shows better convergence properties in the optimization process, but in the end does not allow a safe load change, since more attention is spent on saving control costs than approximating the desired

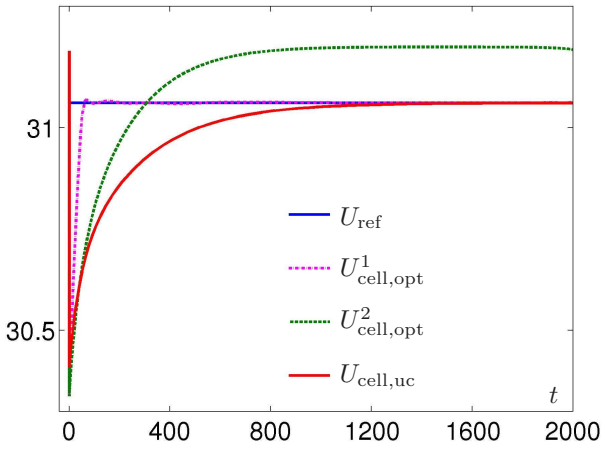

Fig. 13 Results from FOTD: desired state $U_{\mathrm{ref}}(t)$ and optimal cell voltages $U^1_{\mathrm{cell,opt}}(t)$ for $\lambda = 5{\cdot}10^{-5}$, respectively $U^2_{\mathrm{cell,opt}}(t)$ for $\lambda = 5 \cdot 10^{-2}$ and the uncontrolled behavior $U_{\mathrm{cell,uc}}(t)$ of the fuel cell at a load change from 0.4 to 0.5 at $t = 0.1$; cf. [23].

state. This trade-off has to be regarded when chosing λ in application problems. The convergence behaviour of the BFGS-method is depicted in Fig. 14 for the practical interesting case $\lambda = 5 \cdot 10^{-5}$. With fast progress in the beginning, the method converges after 50 iterations with a total number of 129 evaluations of the reduced objective, which involves one state and one adjoint system solve each. Due to the time stepping methods, the amount of one state solve can be reduced to some seconds even when using fine space and particularly rather fine time discretizations. The whole optimization run takes 27 minutes on an Intel64 3.3 GHz dual core with 8 GB RAM using MATLAB R2010B.

Figure 15 shows the optimal solutions for the anode, cathode, and solid temperature for $\lambda = 5 \cdot 10^{-5}$. Starting from a stationary state, the sudden load change at $t = 0.1$ induces rapid changes in the temperatures. Due to optimal control, a new stationary state is reached again fastly. For more results concerning the FOTD approach, we refer to [23].

In principle this approach is also applicable to the 2D models.

4 Conclusion

The race between *first discretize then optimize* vs. *first optimize then discretize* in the numerical solution of optimal control problems involving ordinary differential equations is decided in favour of methods for solving finite dimensional nonlinear programming to which the infinitely dimensional optimal control problem is transcribed. The fewer requirements on the knowledge of the user of such software, the generally large domain of convergence in cases of highly nonlinear problems do pay off against the possible deeper insight into the structure of the optimal solution which can be obtained by methods for the solution of multi-point boundary value problems to evaluate the necessary optimality conditions of the maximum principle.

Iter	F-count	f(x)	First-order optimality	Norm of step
0	1	3.137e+01	5.2e-02	
5	6	7.001e+00	1.2e-02	3.6e-01
10	13	6.343e+00	1.1e-02	1.2e-01
15	20	5.861e+00	5.8e-03	1.0e-01
20	27	5.687e+00	2.3e-03	2.9e-02
25	34	5.653e+00	1.2e-03	1.8e-02
30	42	5.488e+00	2.6e-03	1.4e-01
35	47	5.428e+00	1.2e-03	7.3e-03
40	56	5.383e+00	1.5e-03	2.4e-02
45	77	5.374e+00	3.9e-04	4.1e-03
50	129	5.372e+00	3.2e-04	1.8e-10

Fig. 14 Results from FOTD: Output of the Quasi-Newton method `fmincon`(MATLAB) for $\lambda = 5 \cdot 10^{-5}$; cf. [23].

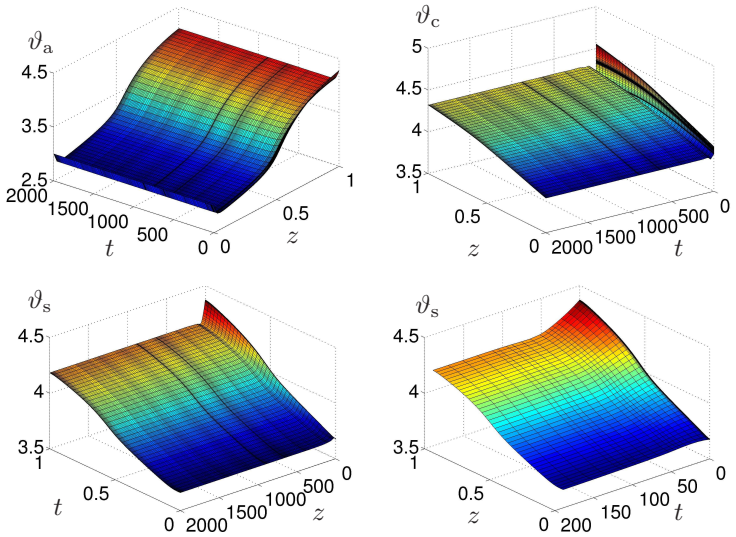

Fig. 15 Results from FOTD: Optimal solutions for the anode, cathode, and solid temperature (with zoom) for a load change started at $t = 0.1$ for $\lambda = 5 \cdot 10^{-5}$ and control ϑ_{air}.

Moreover, these conditions can generally be verified approximately out of the discrete optimal solution using also the discrete adjoints. Only if an extreme accuracy is required, for example in certain aerospace applications, a postprocessing step with a multi-point boundary value problem such as multiple shooting must be carried through.

However, the competition is still open if optimal control problems subject to partial differential equations are considered. The classification between FOTD and FDTO surely needs a more thorough subdivision particularly if automatic differentiation is involved. In the following we rate the methods investigated in this paper according to the way the gradient (and possibly the Hessian) is provided.

Numerical differentiation provides the fastest way to transform a simulation code into an optimization code. However, methods based on numerical differentiation face the problem, that the gradient evaluation needs as much PDAE solves as there are optimization variables. To reduce this number, we generally work on the reduced objective and shorten the time horizon by interval decompositions or receding horizon methods. Then we get acceptable results. Nevertheless, methods based on numerical differentiation are limited to a coarse discretization of the control. By separation of the control and state discretization in method 1, we were able to resolve the PDAE model much more accurate with an adaptive time-stepping method, while keeping the number of control points low.

The second variant is to provide the gradient by AD. We have shown how to apply a black-box use of AD to the PDAE model in method 2 yielding accurate gradient and Hessian information. Therefore, we have chosen a discretization on fixed grids. Inside the AMPL/IPOPT framework we were restricted to coarse discretizations, since the complexity of the model causes bad convergence properties. However, only this method allows to include inequality constraints on the state variables directly. Especially for simpler models one gets a reliable method for the direct treatment of such state constraints, cf. [29]. Unfortunately, not all codes can be treated by black-box AD, particularly adaptive codes can not be differentiated at present time.

In a third variant, the gradient is provided by solving adjoint equations. Setting up these equations is time-consuming and needs much more insight in the theory of optimal control. Nevertheless, a good adjoint solver yields the gradient information with highest accuracy and efficiency compared to the other two variants, since the effort of a gradient evaluation and a state solve is nearly equal. In addition, adaptivity in space or time can be incorporated efficiently. The question is, whether the good results amortize the human ressources that are necessary to develop an adjoint solver. To simplify the derivation and coding of the adjoint solver, we suggested to separate the differential operators and the source terms and to apply symbolic or automatic differentiation to the source terms. With the adjoint solver, we set up a Quasi-Newton method that always solves state and adjoint equations separately making use of time-stepping schemes. This allows a faster and more accurate solution of the optimal control problem compared to the other methods.

However, the considerable effort for the implementation of the optimality system remains an issue which cannot be neglected. Therefore, there remains a need for further development of automatic generation of parts of the gradient or parts of the optimality systems by AD.

References

[1] C. H. Bischof, H. M. Bücker, B. Lang, A. Rasch, and A. Vehreschild, *Combining Source Transformation and Operator Overloading Techniques to Compute Derivatives for MATLAB Programs*, Conference Proceeding, Proceedings of the Second IEEE International Workshop on Source Code Analysis and Manipulation (SCAM 2002), October 1, 2002, Montreal Canada (IEEE Computer Society, 2002); URL: `http://www.sc.rwth-aachen.de/adimat/`.

[2] Maïtine Bergounioux, Kazufumi Ito, and Karl Kunisch. *Primal-dual strategy for constrained optimal control problems*, SIAM J. on Control and Optimization **37**, 1176–1194 (1999).
[3] A. Borzi and V. Schulz, *Computational Optimization of Systems Governed by Partial Differential Equations* (SIAM, Philadelphia, USA, 2011).
[4] C. Büskens, *Echtzeitoptimierung und Echtzeitoptimalsteuerung parametergestörter Probleme* Habilitationsschrift (Faculty of Mathematics and Physics, University of Bayreuth, Bayreuth, Germany, 2002).
[5] K. Chudej, H. J. Pesch, J. Rang, *Index Analysis of Models*, in: *Molten Carbonate Fuel Cells — Modeling, Analysis, Simulation, and Control*, edited by K. Sundmacher, A. Kienle, H. J. Pesch, J. Berndt, and G. Huppmann, (Wiley-VCH, Weinheim, Germany, 2007), pp. 63–74.
[6] K. Chudej, H. J. Pesch, K. Sternberg, *Optimal Control of Load Changes for Molten Carbonate Fuel Cell Systems: A Challenge in PDE Constrained Optimization*, SIAM J. on Applied Mathematics **70**(2), 621–639 (2009).
[7] R. Fourer, D. M. Gay, and B. W. Kernighan, *AMPL, A Modeling Language for Mathematical Programming*, (Curt Hinrichs, Thomson — Brooks/Cole, Pacific Grove, USA, 2003); URL: `http://www.ampl.com/` (2012).
[8] Eduardo Casas. *Control of an elliptic problem with pointwise state constraints*, SIAM J. on Control and Optimization **4**, 1309–1322 (1986).
[9] P. Gill, W. Murray, and M. Saunders, *SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization*, SIAM Review **47**(1), 99–131 (2005).
[10] P. Heidebrecht, *Modelling, Analysis and Optimisation of Molten Carbonate Fuel Cell with Direct Internal Reforming*, PhD thesis (Faculty of Chemical and Systems Engineering, University of Magdeburg, Germany, 2004).
[11] R. Herzog and K. Kunisch, *Algorithms for PDE-constrained optimization*, GAMM-Mitteilungen **33**(2), 163–176 (2010).
[12] Michael Hintermüller, Kazufumi Ito, and Karl Kunisch. *The primal-dual active set method as a semi-smooth Newton method*, SIAM J. on Optimization **13**(3), 865–888 (2003).
[13] Michael Hintermüller and Karl Kunisch. *PDE-constrained optimization subject to pointwise constraints on the control, the state, and its derivative*, SIAM J. on Optimization **20**(3), 1133–1156 (2009).
[14] Michael Hintermüller and Karl Kunisch. *Stationary optimal control problems with pointwise state constraints*, In Numerical PDE Constrained Optimization, volume 72 of Lecture Notes in Computational Science and Engineering (Springer, Berlin Heidelberg, 2009).
[15] Michael Hinze and Karl Kunisch, *On Suboptimal Control Strategies for the Navier-Stokes Equations*. ESAIM: Proceedings **4** (EDP Sciences, Les Ulis Cedex A, France, 1998), pp. 181–198.
[16] Michael Hinze and Karl Kunisch, *Second Order Methods for Optimal Control of Time-Dependent Fluid Flow*, SIAM J. on Control and Optimization **40**(3), 925–946 (2001).
[17] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich, *Optimization with PDE constraints*, Mathematical Modelling: Theory and Applications **23** (Springer, Berlin, Germany, 2008).
[18] J. Kerler, *Optimale Steuerung einer Schmelzkarbonatbrennstoffzelle mit direkten Verfahren*, Master Thesis (Chair of Mathematics in Engineering Sciences, University of Bayreuth, Germany, 2012).
[19] J. L. Lions, *Optimal Control of Systems Governed by Partial Differential Equations*, Grundlehren der Mathematischen Wissenschaften **170** (Springer, Berlin, Germany, 1971).
[20] Christian Meyer, Arnd Rösch, and Fredi Tröltzsch. *Optimal control problems of PDEs with regularized pointwise state constraints*, Computational Optimization and Applications **33**(2-3), 209–228 (2006).
[21] S. R. Mc Reynolds and A. E. Bryson, *A Successive Sweep Method for Solving Optimal Control Problems* 6th Joint Automatic Control Conference, Troy, N. Y. (IEEE Control System Society, 1965), pp. 551–555.
[22] T. Richter, A. Springer, and B. Vexler, *Efficient Numerical Realization of Discontinuous Galerkin Methods for Temporal Discretization of Parabolic Problems*, Numerische Mathematik, accepted (2012).

[23] A. Rund, *Beiträge zur Optimalen Steuerung partiell-differential algebraischer Gleichungen*, PhD thesis (Faculty of Mathematics, Physics, and Computer Science, University of Bayreuth, Germany, 2012).
[24] A. Rund and K. Chudej. *Optimal control for a simplified 1D fuel cell model*, Mathematical and Computer Modelling of Dynamical Systems **18**(4), 379–396 (2012).
[25] K. Sternberg, *Simulation, Optimale Steuerung und Sensitivitätsanalyse einer Schmelzkarbonat-Brennstoffzelle mithilfe eines partiell differential-algebraischen Gleichungssystems*, PhD thesis (Faculty of Mathematics, Physics, and Computer Science, University of Bayreuth, Bayreuth, Germany, 2007).
[26] K. Sundmacher, A. Kienle, H. J. Pesch, J. Berndt, and G. Huppmann (eds.), *Molten Carbonate Fuel Cells — Modeling, Analysis, Simulation, and Control*. (Wiley-VCH, Weinheim, Germany, 2007).
[27] F. Tröltzsch, *Optimal Control of Partial Differential Equations: Theory, Methods and Applications* Graduate Studies in Mathematics **112** (American Mathematical Society, Providence, USA, 2010).
[28] A. Wächter and L. T. Biegler, *On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming* Mathematical Programming **106**(1), 25–57 (2006); see URL: `https://projects.coin-or.org/Ipopt/` (2012).
[29] Stefan Wendl, Hans Josef Pesch and Armin Rund. *On a state-constrained PDE optimal control problem arising from ODE-PDE optimal control*, In Moritz Diehl, Francois Glineur, Elias Jarlebring, and Wim Michiels, editors, Recent Advances in Optimization and its Applications in Engineering, 429–438 (Springer, Berlin, 2010).
[30] J. Carsten Ziems and Stefan Ulbrich. *Adaptive multilevel inexact SQP methods for PDE-constrained optimization*, SIAM J. on Optimization **21**(1), 1–40 (2011).