Direct and Indirect Multiple Shooting for Parabolic Optimal Control Problems

Thomas Carraro and Michael Geiger

Abstract We present two multiple shooting approaches for optimal control problems (OCP) governed by parabolic partial differential equations (PDE). In the context of ordinary differential equations, shooting techniques have become a state-of-the-art solver component, whereas their application in the PDE case is still in an early phase of development. We derive both direct (DMS) and indirect (IMS) multiple shooting for PDE optimal control from the same extended problem formulation. This approach shows that they are algebraically equivalent on an abstract function space level. However, discussing their respective algorithmic realizations, we underline differences between DMS and IMS. In the numerical examples, we cover both linear and nonlinear parabolic side conditions.

1 Introduction

Multiple shooting methods have been extensively used during the past four decades to solve both ODE boundary value problems (BVP) and optimal control problems (OCP), and for the latter problem class different shooting techniques were developed, according to the general dichotomy of indirect and direct solution methods for OCP. In the context of PDE constrained OCP, where the past 15 years have seen a rapid development of both theoretical insights and solution algorithms, shooting methods are up to now rarely used despite their success in the ODE framework. Direct multiple shooting (DMS) or related time-domain decomposition methods are treated, e.g., in [13, 29], and an indirect shooting approach was introduced in [14]. All these articles employ shooting techniques focusing on specific aspects such as adaptivity or preconditioning, while our scope in this article is to show the derivation of DMS and IMS from the same extended problem formulation to underline their peculiarities in the PDE framework.

Shooting algorithms for parabolic OCP have to overcome additional difficulties caused mainly by the spatial variables. On the continuous level, the functional

T. Carraro (✉) • M. Geiger
Institute of Applied Mathematics, Heidelberg University, Heidelberg, Germany
e-mail: thomas.carraro@iwr.uni-heidelberg.de; michael.geiger@iwr.uni-heidelberg.de

T. Carraro et al. (eds.), *Multiple Shooting and Time Domain Decomposition Methods*, Contributions in Mathematical and Computational Sciences 9,
DOI 10.1007/978-3-319-23321-5_2

analytic setting (solution spaces, weak formulations etc.) has to be developed carefully; e.g., the initial values for the subinterval solutions are L^2 functions rather than $\mathbb{R}^n$ vectors as in the ODE case. On the discrete level, the usually high-dimensional spatial discretization leads to large-scale optimization problems and strongly advises the development of suitable adaptive techniques.

The two most attractive features of multiple shooting are its intrinsic stabilizing effect and its potential for parallelization. The former enables the solution of ill-conditioned problems where other methods fail; such instabilities are mirrored by local stability estimates such as

$$\|u(t; s_1) - u(t; s_2)\| \leq ce^{L(t-t_0)}\|s_1 - s_2\|$$

that are common in the analysis of initial value problems (IVP). Here, t is the independent variable normally interpreted as time, t_0 is the initial time-point, s_1 and s_2 are two parameter values (in multiple shooting methods, they denote initial values) and $u(t; s)$ is the solution depending on the parameter s. Furthermore, L is a fixed Lipschitz constant inherent to the problem; the value of the exponential factor can be controlled by splitting the solution interval into smaller parts as is common in multiple shooting.

The parallelizability of shooting techniques has been addressed and exploited in the PDE context by the so-called parareal method (developed in [22]) which has been shown to be equivalent to multiple shooting in [11]. Despite the results mentioned so far, there are many aspects that have not yet been systematically examined, which is why shooting techniques for PDE problems still constitute an interesting and promising subject.

A detailed presentation of indirect multiple shooting (IMS) for nonlinear parabolic OCP with additional control constraints has recently been given in [9]. In the current publication, we continue this work by comparing IMS and DMS techniques for the mentioned problem class, but without considering additional control or state constraints. The latter simplification prevents us from losing track of our main objective, namely to show the equivalence of IMS and DMS on an abstract function space level and the differences of their respective algorithmic concretization. Furthermore, as in the well studied ODE case, we expect IMS and DMS to behave differently in the presence of control and/or state constraints. In fact, in the ODE context the presence of state and control constraints has led to prefer DMS to IMS. An accurate performance comparison between DMS and IMS in PDE context is left as an interesting topic for further research.

In the literature of multiple shooting methods, the distinction between direct and indirect approaches is done according to two aspects: how the method is derived and at which stage the underlying system is discretized. According to the first aspect, the distinction is done between methods that are derived by certain optimality conditions (indirect approach) and methods that are not obtained by optimality conditions (direct approach). Considering the second aspect, the same methods are classified as indirect if they use the ‘first optimize then discretize’ approach or direct if they follow the ‘first discretize, then optimize’ approach.

Formerly in the ODE context, to develop indirect methods a Hamiltonian functional was introduced and the optimality conditions were derived following the Pontryagin Maximum Principle [25] and introducing adjoint (or co-state) variables. Numerical methods for this approach lead to a boundary value problem for the state and adjoint variables. As a result, in indirect methods the control is not present in the shooting system. On the contrary, in direct methods the control variable is included in the shooting system, leading to a procedure that in the multiple shooting context is also called 'all-at-once' approach.

Following the other distinction between indirect and direct methods, namely the differentiation between the approaches that optimize first and approaches that discretize first, in indirect methods first the optimality conditions are derived at the continuous level, and then a discretization method is used to derive a finite dimensional system that can be solved numerically. On the contrary, in direct methods the state and control variables are discretized first and typically the discrete control space has low dimension. Then, a method for nonlinear programming problems is applied to the resulting discrete system [18, 28]. According to this distinction, direct methods are not derived in a function space setting. This has the advantage of avoiding the definition of adjoint variables in function spaces, especially if state constraints are included in the optimization problem.

We do not want to discuss the several arguments that indicate advantages and disadvantages of either one method or the other, we will rather discuss the derivation of the two methods starting from the same formulation at the continuum level. Therefore, we give a unique argumentation to define whether a method is 'direct' or 'indirect' allowing to distinguish the two approaches without recurring to specific discretization methods. Proceeding like this has practical consequences because keeping the derivation at the continuous level will allow for example to derive error estimation methods in PDE context for the specific discretization of choice. This opens up new directions for future research.

The remainder of this contribution is oriented along the following outline: In the next section, we recapitulate the notational framework for PDE optimal control. Section 3 presents the OCP in a slightly modified but equivalent form which provides the suitable context for shooting methods. The KKT conditions of this extended OCP formulation are the starting point for both IMS and DMS methods on a function space level, which are described in separate sections. We start Sect. 4 by introducing a second variant of DMS which is common in ODE and DAE governed optimal control (here called 'classical' DMS); the connection between these two DMS approaches constitutes the main result of the section. After some brief remarks on discretization in Sect. 5, we discuss the effects of the differences between IMS and DMS in Sect. 6 by considering two concrete numerical examples. Some concluding remarks as well as an outlook toward possible further research constitute the final section.

2 Preliminaries

The general structure of an OCP requires the minimization of an objective functional, where the minimum is sought in a set of functions u given as solutions of a differential equation which depends on a control quantity q:

$$\min_{(q,u)} J(q,u) \text{ subject to } e(q,u) = 0. \tag{1}$$

In many problems, one has to deal with additional constraints to the control and/or state variables of the form $c(q,u) \leq 0$ which make the problems more difficult to solve. We skip them in order to avoid an excessive notation and in order not to lose track of our actual objective, namely comparing IMS and DMS methods. Concerning multiple shooting for parabolic OCP with control constraints $c(q) \leq 0$, we refer to [9].

We now describe in detail the parabolic OCP which is considered throughout this article. In this context, we have to deal with the following theoretical setting. The OCP reads in detail:

$$\min_{(q,u)} J(q,u) = \kappa_1 J_1(u) + \kappa_2 J_2(u(T)) + \frac{\alpha}{2}\|q\|_Q^2, \tag{2}$$

subject to the parabolic PDE

$$\partial_t u(x,t) + \mathscr{A}(u(x,t)) + \mathscr{B}(q(x,t)) = f(x,t) \quad \text{in } \Omega \times I, \tag{3a}$$

$$u(x,0) = u_0(x) \quad \text{in } \Omega \tag{3b}$$

We discuss the constituent parts of this formulation separately. Therefore, we assume that $V \hookrightarrow H = H^* \hookrightarrow V^*$ is a Gelfand triple of Hilbert spaces of functions on Ω (where the superscript $*$ denotes duality of spaces) and R is a suchlike Banach space. In the objective functional $J(q,u)$, we eliminate either $J_1(u)$, which we always assume to be of tracking type $\int_I \|u(t) - \hat{u}(t)\|_V^2 \, dt$, or the end-time matching term $J_2(u(T)) := \|u(T) - \hat{u}_T\|_H^2$ by imposing the conditions $\kappa_i \in \{0, \frac{1}{2}\}, \kappa_1 \neq \kappa_2$. The term $\frac{\alpha}{2}\|q\|_Q^2$ serves as a regularization term, and α is the usual regularization parameter.

The computational domain of our problem is a space-time cylinder $\Omega \times I$ with a bounded convex polygonal or polyhedral spatial domain $\Omega \subset \mathbb{R}^d$ with $d \in \{1,2,3\}$ and a finite time interval $I = (0,T)$. The function spaces for state and control variables are usually Bochner spaces of the type $W(I;Y)$ where the time variable t is mapped into a Banach space Y.

In the above function space framework, the natural setting for the parabolic PDE (3a) is the following: For given $q(x,t) \in Q := L^2(I;R)$ and righthand side $f(x,t) \in L^2(I;V^*)$, find a state function $u(x,t)$ that satisfies (3) obeying additionally imposed suitable boundary conditions. Under these structural assumptions, the

solution space for $u(x,t)$,

$$X := \{v(x,t) \in L^2(I;V) \mid \partial_t v(x,t) \in L^2(I;V^*)\}, \tag{4}$$

is known to be continuously embedded into the space $C(\bar{I};H)$ of temporally continuous functions with values in H (see, e.g., [10]), which means that an initial condition $u_0(x) \in H$ is well-defined. The differential operator $\mathscr{A} : X \to L^2(I;V^*)$ may be linear or nonlinear, whereas $\mathscr{B} : L^2(I;R) \to L^2(I;V^*)$ is usually linear and often simply an injection operator, given $R \hookrightarrow V^*$.

For a weak formulation of (3) we need some preparatory definitions. If $\overline{\mathscr{A}} : V \to V^*$ and $\overline{\mathscr{B}} : R \to V^*$ are pointwise-in-time operators corresponding to $\mathscr{A}$ and $\mathscr{B}$, respectively, we assume the elliptic operator $\overline{\mathscr{A}}$ to be coercive and define the following scalar products and semilinear forms:

$$((u,\varphi))_I := \int_I (u(t),\varphi(t))_H \, dt, \qquad a_I(u)(\varphi) := \int_I \langle \overline{\mathscr{A}}(u(t)),\varphi(t)\rangle_{V^*\times V} \, dt,$$

$$b_I(q)(\varphi) := \int_I \langle \overline{\mathscr{B}}(q(t)),\varphi(t)\rangle_{V^*\times V} \, dt.$$

We normally omit the index I denoting the integration interval if it is evident from the context. In this notational framework, the weak formulation of (3) reads: Find $u \in X$, so that for all $\varphi \in X$

$$((\partial_t u,\varphi)) + a(u)(\varphi) + b(q)(\varphi) + (u(0),\varphi(0)) = ((f,\varphi)) + (u_0,\varphi(0)), \tag{6}$$

where we included the initial condition (3b) weakly.

Remark 1 Conditions which guarantee that linear parabolic equations of type (3) or (6) possess a unique solution are provided in [16]. This framework is based on the underlying elliptic case which was presented, e.g., in [32]. We emphasize that both our theoretical observations for the linear case in the current and next sections and our linear example Sect. 6.1 fulfil the mentioned conditions and therefore allow for unique solutions.

The solution of the OCP is known to be among the stationary points of the Lagrange functional

$$\mathscr{L}(q,u,z) := J(q,u) + e(q,u;z) \tag{7}$$

where $e(q,u;z)$ is an abbreviation for the weakly formulated PDE side condition (6),

$$e(q,u;z) := ((\partial_t u,z)) + a(u)(z) + b(q)(z) - ((f,z)) + (u(0)-u_0,z(0)),$$

which sometimes enables a compact presentation on a more abstract level, see the discussion in Sect. 4. The Lagrange multiplier $z \in X$ denotes the solution of

the adjoint equation $\mathscr{L}_u'(\delta u) = 0$ which naturally arises as part of the following optimality conditions:

$$\mathscr{L}_z'(\delta z) = ((\partial_t u, \delta z)) + a(u)(\delta z) + b(q)(\delta z) - ((f, \delta z)) + (u(0) - u_0, \delta z(0)) = 0, \tag{8a}$$

$$\mathscr{L}_u'(\delta u) = J_u'(q,u)(\delta u) - ((\partial_t z, \delta u)) + a_u'(u)(\delta u, z) + (z(T), \delta u(T)) = 0, \tag{8b}$$

$$\mathscr{L}_q'(\delta q) = J_q'(q,u)(\delta q) + b_q'(q)(\delta q, z) = 0. \tag{8c}$$

This so-called KKT system consists of the derivatives of (7) that form the state, adjoint and control equations. In the next section, we rewrite the above OCP in a form that is more suited to the derivation of multiple shooting algorithms.

Remark 2 The regularity of the Lagrange multiplier z is in general a delicate matter. Due to the structure of our objective functional, which may either be a temporally distributed L^2-term or an L^2-term at the final timepoint T, in our case the adjoint variable z lies in the same space X as the state variable u. A similar argument reveals the regularity of the adjoint variables $\lambda \in H$ in Sect. 3.1.

3 Multiple Shooting Methods for Parabolic OCP

As a suitable starting point for our observations we have introduced a general parabolic OCP which we extend, in Sect. 3.1, to a formulation tailored to the derivation of multiple shooting. The remainder of this section will be concerned with two variants of multiple shooting that are common in ODE optimal control. We will embed them into the context of parabolic OCP, thereby taking into account the additional challenges arising during the transfer from ODE to PDE (see Sect. 3.2 for IMS and Sect. 3.3 for DMS). Instead of only emphasizing the differences between these multiple shooting approaches, it is our objective to also show that they are rooted in one single common problem formulation and merely constitute different algorithmic realizations that may render either IMS or DMS preferable in concrete situations, depending on the problem at hand. We compare both shooting techniques in the concluding Sect. 3.4.

Remark 3 For simplicity, we assume the PDE side conditions of the OCP occurring in the following sections to be uniquely solvable, which has to be justified separately in every concrete problem. Furthermore, we assume also the OCP themselves to be uniquely solvable, which is guaranteed in a linear-quadratic framework on a convex domain but in general has to be verified. By these assumptions we avoid a detailed

discussion of theoretical issues; for more information, we refer to the textbooks [16, 30].

3.1 The Modified Formulation of the Optimal Control Problem

The modification of the OCP (2) subject to (6) relies on a decomposition of the closure $\bar{I}$ of the interval I,

$$\bar{I} = \{\tau_0\} \cup \bigcup_{j=0}^{M-1} I_j, \quad I_j = (\tau_j, \tau_{j+1}], \tag{9}$$

where $\tau_0 = 0$ and $\tau_M = T$, and the subsequent redefinition of the OCP in terms of local control and state functions q^j, u^j on the subintervals I_j, which lie in the spaces $Q^j := L^2(I_j; R)$ and $X^j := \{v \in L^2(I_j; V) \,|\, \partial_t v \in L^2(I_j; V^*)\}$, respectively. For a more global view on these intervalwise problems, we define the compositions $\mathbf{u} = ((u^j)_{j=0}^{M-1})$ and $\mathbf{q} = ((q^j)_{j=0}^{M-1})$ as well as the corresponding spaces

$$\mathbf{X} := \bigtimes_{j=0}^{M-1} X^j, \quad \mathbf{Q} := \bigtimes_{j=0}^{M-1} Q^j.$$

We note that $\mathbf{X} = \{v \in L^2(I; V) \,|\, v|_{I_j} \in X^j\}$ and $\mathbf{Q} = \{q \in L^2(I; R) \,|\, q|_{I_j} \in Q^j\}$, which implies $X \subsetneq \mathbf{X}$ and $Q = \mathbf{Q}$. With these notations, the modified control problem reads:

$$\min_{(\mathbf{q},\mathbf{u})} \bar{J}(\mathbf{q}, \mathbf{u}) := \sum_{j=0}^{M-1} J^j(q^j, u^j) = \kappa_1 \sum_{j=0}^{M-1} \int_{I_j} \|u^j - \hat{u}|_{I_j}\|_V^2 \, dt$$

$$+\kappa_2 \|u^{M-1}(\tau_M) - \hat{u}_T\|_H^2 + \frac{\alpha}{2} \sum_{j=0}^{M-1} \int_{I_j} \|q^j\|_Q^2 \, dt \tag{10a}$$

$$\text{s. t.} \quad ((\partial_t u^j, \varphi)) + a(u^j)(\varphi) + b(q^j)(\varphi) - ((f|_{I_j}, \varphi))$$

$$+(u^j(\tau_j) - s^j, \varphi(\tau_j)) = 0 \ \text{ for } j \in \{0, \ldots, M-1\}. \tag{10b}$$

In this formulation, Eq. (10b) represent IVP on the subintervals I_j. However, we do not know the exact values $u(\tau_j)$ and therefore have to impose artificial initial values $\mathbf{s} = (s^j)_{j=0}^M \in H^{M+1}$. This leads to jumps in the global solution $\mathbf{u}$ composed of the interval solutions (i.e. $\mathbf{u}|_{I_j} \equiv u^j$). Therefore, problem (10) cannot be equivalent to the original OCP, because $\mathbf{u} \notin C(\bar{I}; H)$, whereas the solution $u \in X$ of the global OCP has to be continuous on I due to the above mentioned embedding $X \hookrightarrow C(\bar{I}; H)$. This matches the fact that X is a proper subset of $\mathbf{X}$. We therefore have

to enforce the global continuity of the solution $\mathbf{u}$ of (10) by imposing the following additional continuity conditions:

$$(s^0 - u_0, \phi) = 0 \quad \forall\, \phi \in H, \tag{11a}$$

$$(s^{j+1} - u^j(\tau_{j+1}), \phi) = 0 \quad \forall\, \phi \in H,\, j \in \{0, \dots, M-1\}. \tag{11b}$$

Before we prove the equivalence of the original OCP and the extended problem formulation (10)–(11), we state the following preparatory lemma.

Lemma 1 *The objective functionals $J(q,u)$ and $\overline{J}(\mathbf{q},\mathbf{u})$ coincide for $\mathbf{u} = ((u^j)_{j=0}^{M-1})$ with $u^j = u|_{I_j}$, i.e. for globally continuous intervalwise defined functions $\mathbf{u}$.*

Proof Due to the additivity of integration on subintervals, we obtain

$$J(q,u) = \kappa_1 \int_I \|u(t) - \hat{u}(t)\|_V^2\, dt + \kappa_2 \|u(T) - \hat{u}_T\|_H^2 + \frac{\alpha}{2} \int_I \|q(t)\|_R^2\, dt$$

$$= \kappa_1 \sum_{j=0}^{M-1} \int_{I_j} \|u^j(t) - \hat{u}|_{I_j}(t)\|_V^2\, dt + \kappa_2 \|u^{M-1}(\tau_M) - \hat{u}_T\|_H^2 + \frac{\alpha}{2} \sum_{j=0}^{M-1} \int_{I_j} \|q^j(t)\|_R^2\, dt.$$

This corresponds to $\overline{J}(\mathbf{q},\mathbf{u}) = \sum_{j=0}^{M-1} J^j(q^j, u^j)$. □

The following theorem states the equivalence of the original and the modified OCP.

Theorem 1

(*a*) *Let $(q,u) \in Q \times X$ be a solution to the original OCP (2) subject to (6). Then $(\mathbf{q},\mathbf{u}) \in \mathbf{Q} \times \mathbf{X}$, defined by $q^j := q|_{I_j}$ and $u^j := u|_{I_j}$, is a solution to the modified OCP (10)–(11).*

(*b*) *Let, on the other hand, $(\mathbf{q},\mathbf{u}) \in \mathbf{Q} \times \mathbf{X}$ solve the modified problem (10)–(11). If we define q by $q|_{I_j} := q^j$ and u by $u|_{I_j} := u^j$, then $(q,u) \in Q \times X$ solves the original OCP (2) subject to (6).*

Proof

(a) Since $u \in X$ is globally continuous in time, we have $s^0 = u_0$ as well as $s^{j+1} = u^{j+1}(\tau_{j+1}) = u(\tau_{j+1})$ and $u^j(\tau_{j+1}) = u(\tau_{j+1})$, which means in turn $s^{j+1} = u^j(\tau_{j+1})$. Thus, the matching conditions (11) are fulfilled. Let now $(\tilde{\mathbf{q}},\tilde{\mathbf{u}}) = ((\tilde{q}^j, \tilde{u}^j)_{j=0}^{M-1}) \in \mathbf{Q} \times \mathbf{X}$ such that $\overline{J}(\tilde{\mathbf{q}},\tilde{\mathbf{u}}) < \overline{J}(\mathbf{q},\mathbf{u})$ and the continuity conditions (11) are fulfilled. The latter assumption immediately implies $\tilde{\mathbf{u}} \in X$, i.e. $(\tilde{q},\tilde{u}) := (\tilde{\mathbf{q}},\tilde{\mathbf{u}}) \in Q \times X$ due to $\mathbf{Q} = Q$. Lemma 1 now yields

$$J(\tilde{q},\tilde{u}) = \overline{J}(\tilde{\mathbf{q}},\tilde{\mathbf{u}}) < \overline{J}(\mathbf{q},\mathbf{u}) = J(q,u)$$

which is a contradiction to the assumed optimality of (q,u).

(b) Since $\mathbf{u}$ is part of a solution of the modified OCP, especially (11), we know that $s^0 = u_0$ and $s^{j+1} = u^j(\tau_{j+1})$. The initial value s^{j+1} on I_{j+1} clearly fulfils $s^{j+1} = u^{j+1}(\tau_{j+1})$. From $\mathbf{u} \in \mathbf{X}$ we know that $u^j \in C(\overline{I}_j; H)$, and together with

the global continuity we know $\mathbf{u} \in C(\bar{I}; H)$. Considering $\partial_t u^j \in L^2(I_j; V^*)$, the corresponding global property $\partial_t u \in L^2(I; V^*)$ now directly follows. This means that u, defined by $u|_{I_j} := u^j$, lies in X, and together with q (analogously defined by $q|_{I_j} := q^j$) we obtain $(q, u) \in Q \times X$. Assuming that there is $(\tilde{q}, \tilde{u}) \in Q \times X$ with $J(\tilde{q}, \tilde{u}) < J(q, u)$, we get

$$\overline{J}(\tilde{\mathbf{q}}, \tilde{\mathbf{u}}) = J(\tilde{q}, \tilde{u}) < J(q, u) = \overline{J}(\mathbf{q}, \mathbf{u})$$

by Lemma 1, which is a contradiction to the optimality of $(\mathbf{q}, \mathbf{u})$. □

Agreement *To avoid a cumbersome case-by-case analysis, we assume for the rest of this contribution that all considerations are based on a distributed objective functional corresponding to* $\kappa_1 = \frac{1}{2}$ *and* $\kappa_2 = 0$*. The necessary modifications in case of an end-time functional (where* $\kappa_1 = 0$ *and* $\kappa_2 = \frac{1}{2}$*) are straightforward and will be covered in brief remarks.*

The reformulated problem (10)–(11) is the starting point for multiple shooting algorithms. In order to state the IMS and DMS methods properly, we have to derive the first order necessary optimality conditions of the modified OCP. Therefore, we first define the corresponding Lagrange functional, which is an extended version of (7) where the equality constraints (11) are considered in addition. We have in detail:

$$\begin{aligned}\mathscr{L}((q^j, u^j, z^j)_{j=0}^{M-1}, (s^j, \lambda^j)_{j=0}^{M}) &:= \sum_{j=0}^{M-1} J^j(q^j, u^j) \\ &+ \sum_{j=0}^{M-1} [((\partial_t u^j, z^j)) + a(u^j)(z^j) + b(q^j)(z^j) - ((f|_{I_j}, z^j))] \\ &+ \sum_{j=0}^{M-1} (u^j(\tau_j) - s^j, z^j(\tau_j)) + \sum_{j=0}^{M-1} (s^{j+1} - u^j(\tau_{j+1}), \lambda^{j+1}) + (s^0 - u_0, \lambda^0)\end{aligned} \tag{13}$$

The cost functional (10a) has been rearranged in an intervalwise fashion where all addends are structured alike. We have two kinds of Lagrange multipliers, the adjoint variables $\mathbf{z} = ((z^j)_{j=0}^{M-1}) \in \mathbf{X}$ corresponding to the intervalwise PDE side condition, and, newly, the spatial functions $\boldsymbol{\lambda} = (\lambda^j)_{j=0}^{M} \in H^{M+1}$ as multipliers for the equality constraints (11). We are now able to derive the first order optimality conditions, the so-called KKT system, by differentiating the above Lagrangian w.r.t. all its arguments. This yields, for all test functions $(\delta z, \delta u, \delta q, \delta\lambda, \delta s) \in X^j \times X^j \times Q^j \times H \times H$ and for all $j \in \{0, \cdots, M-1\}$, the intervalwise equations

$$\begin{aligned}\mathscr{L}'_{z^j}(\delta z) = ((\partial_t u^j, \delta z)) + a(u^j)(\delta z) + b(q^j)(\delta z) \\ -((f|_{I_j}, \delta z)) + (u^j(\tau_j) - s^j, \delta z(\tau_j)) = 0,\end{aligned} \tag{14a}$$

$$\mathscr{L}'_{u^j}(\delta u) = J_u^{j'}(q^j, u^j)(\delta u) - ((\partial_t z^j, \delta u)) + a'_u(u^j)(\delta u, z^j)$$
$$+(z^j(\tau_{j+1}) - \lambda^{j+1}, \delta u(\tau_{j+1})) = 0, \quad (14b)$$
$$\mathscr{L}'_{q^j}(\delta q) = J_q^{j'}(q^j, u^j)(\delta q) + b'_q(q^j)(\delta q, z^j) = 0, \quad (14c)$$
$$\mathscr{L}'_{\lambda^0}(\delta\lambda) = (s^0 - u_0, \delta\lambda) = 0, \quad (14d)$$
$$\mathscr{L}'_{\lambda^j}(\delta\lambda) = ((s^{j+1} - u^j(\tau_{j+1}), \delta\lambda) = 0, \quad (14e)$$
$$\mathscr{L}'_{s^j}(\delta s) = ((\lambda^j - z^j(\tau_j), \delta s) = 0, \quad (14f)$$
$$\mathscr{L}'_{s^M}(\delta s) = (\lambda^M, \delta s) = 0. \quad (14g)$$

Remark 4 The last equation (14g) reflects a homogeneous initial condition at the final time-point $\tau_M = T$ [see also the original adjoint equation (8b)], whereas the terms $J_u^{j'}(q^j, u^j)(\delta u)$ serve as righthand sides for the intervalwise adjoints. In case of an end-time functional, (14g) comprises an extra term describing the initial condition for the adjoint equation, whereas the derivatives of the distributed functional terms w.r.t. u in (14b) vanish.

This system of equations can be split into two parts. The first one, Eqs. (14a)–(14c), correspond to the KKT system of the original problem (2) subject to (6), but restricted to a subinterval I_j [compare these equations to (8)]. The corresponding unknowns u^j, z^j and q^j are functions depending on spatial variables and time. The second part consists of Eqs. (14d)–(14g) and appears only in our problem reformulation. The unknowns s^j and λ^j are spatial functions in the isolated time-points τ_j.

Stationary points of the Lagrangian, i.e. solutions of (14), are solution candidates for the modified OCP. The KKT system constitutes a root-finding problem, which can, e.g., be handled by Newton's method. For this purpose, we need the second derivatives of the extended Lagrange functional (13), which is the Jacobian of the optimality conditions (14). We recall that Newton's method for solving a nonlinear but continuously differentiable problem $f(x) = 0$ consists in the iteration

$$x_{k+1} = x_k - J_f(x_k)^{-1} f(x_k)$$

initialized by a suitable starting point x_0. To avoid inverting the Jacobian J_f, this is usually rewritten in the two-step form

$$J_f(x_k) \cdot \delta x = -f(x_k), \quad (15a)$$
$$x_{k+1} = x_k + \delta x. \quad (15b)$$

The linear system displayed in the following is merely a formal representation of (15a) transferred to our context. We therefore rearranged Eq. (14) in a way that facilitates the illustration of IMS and DMS concepts in the subsequent sections but

is not performed in practice:

$$\begin{pmatrix} 0 & \mathscr{L}''_{uz} & \mathscr{L}''_{qz} & \mathscr{L}''_{sz} & 0 \\ \mathscr{L}''_{zu} & \mathscr{L}''_{uu} & 0 & 0 & \mathscr{L}''_{\lambda u} \\ \mathscr{L}''_{zq} & 0 & \mathscr{L}''_{qq} & 0 & 0 \\ \mathscr{L}''_{zs} & 0 & 0 & 0 & \mathscr{L}''_{\lambda s} \\ 0 & \mathscr{L}''_{u\lambda} & 0 & \mathscr{L}''_{s\lambda} & 0 \end{pmatrix} \begin{pmatrix} \delta z \\ \delta u \\ \delta q \\ \delta s \\ \delta\lambda \end{pmatrix} = - \begin{pmatrix} \mathscr{L}'_z \\ \mathscr{L}'_u \\ \mathscr{L}'_q \\ \mathscr{L}'_s \\ \mathscr{L}'_\lambda \end{pmatrix}. \tag{16}$$

The righthand side of (16) consists of block vectors, i.e., the components of $\mathscr{L}'_z$, e.g., are the subinterval state equations, i.e. $\mathscr{L}'_z = (\mathscr{L}'_{z^0}, \cdots, \mathscr{L}'_{z^{M-1}})^\top$. Analogously, each of the solution variables is a block vector consisting of subinterval update values (e.g., $\delta q = (\delta q^{(0)}, \cdots, \delta q^{(M-1)})^\top$). The blocks of the matrix are either zero submatrices in case the equation to be differentiated does not depend on the variable w.r.t. which we differentiate, or they are sparse (often diagonal, or, after discretization, block diagonal) matrices due to the decoupling of the component equations of (14) between different subintervals. We underline that in the context of large scale parabolic OCP this system is never assembled explicitly due to its huge size. The multiple shooting techniques derived in the following sections rely on different splittings of system (16) which reduce its size significantly. Nevertheless, we will still not assemble the corresponding smaller matrices, but employ Krylov-Newton methods that allow to solve the respective Newton equations in a matrix-free manner.

Remark 5 For a discussion of the size of system (16), we exemplarily describe the matrix in more detail. The upper left 3×3 block consists of nine quadratic $M \times M$ blocks, whereas the lower right 2×2 block comprises four $(M+1) \times (M+1)$ blocks. The remaining submatrices are rectangular matrices of appropriate dimension. Summarizing, the Newton matrix is of size $(5M + 2) \times (5M + 2)$. Assuming the number M of subintervals I_j to be of moderate size (from $M = 1$ in the case of simple shooting up to $M \approx 10-100$), the system appears to be small. We emphasize, however, that so far we are still situated in a function space environment, i.e. up to now we considered neither time nor space discretization. We will see later in Sect. 5 that especially the discretization of the spatial variables leads to a huge enlargement of the systems that have to be solved numerically (cf. also Remark 12).

3.2 Indirect Multiple Shooting

We start with indirect shooting and describe first the overall structure of the method. As stated above, we seek ways of splitting the solution process of the linear system (16) (respectively, of its discrete counterpart). One such splitting leads to IMS, which is structured like a two-step (fixed-point) iteration. Furthermore, we discuss some algorithmic details that can also be found in [9]; therefore, we keep the presentation rather short.

3.2.1 Structure

In the Newton system (16), all variables u^j, z^j, q^j, s^j and λ^j are independent. We regroup them according to the following scheme,

$$\left(\begin{array}{ccc|cc} 0 & \mathscr{L}''_{uz} & \mathscr{L}''_{qz} & \mathscr{L}''_{sz} & 0 \\ \mathscr{L}''_{zu} & \mathscr{L}''_{uu} & 0 & 0 & \mathscr{L}''_{\lambda u} \\ \mathscr{L}''_{zq} & 0 & \mathscr{L}''_{qq} & 0 & 0 \\ \hline \mathscr{L}''_{zs} & 0 & 0 & 0 & \mathscr{L}''_{\lambda s} \\ 0 & \mathscr{L}''_{u\lambda} & 0 & \mathscr{L}''_{s\lambda} & 0 \end{array}\right) \left(\begin{array}{c} \delta z \\ \delta u \\ \delta q \\ \hline \delta s \\ \delta\lambda \end{array}\right) = - \left(\begin{array}{c} \mathscr{L}'_z \\ \mathscr{L}'_u \\ \mathscr{L}'_q \\ \hline \mathscr{L}'_s \\ \mathscr{L}'_\lambda \end{array}\right),$$

thus creating two subsystems and introducing inherent dependencies between the variables. In a **first solution step**, we fix $\mathbf{s} = (s^j)_{j=0}^M$ and $\boldsymbol{\lambda} = (\lambda^j)_{j=0}^M$ and solve the intervalwise boundary value problems

$$\begin{aligned} (\!(\partial_t u^j, \delta z)\!) + a(u^j)(\delta z) + b(q^j)(\delta z) - (\!(f|_{I_j}, \delta z)\!) \\ + (u^j(\tau_j) - s^j, \delta z(\tau_j)) = 0, \end{aligned} \tag{17a}$$

$$\begin{aligned} J_u^{j'}(q^j, u^j)(\delta u) - (\!(\partial_t z^j, \delta u)\!) + a'_u(u^j)(\delta u, z^j) \\ + (z^j(\tau_{j+1}) - \lambda^{j+1}, \delta u(\tau_{j+1})) = 0, \end{aligned} \tag{17b}$$

$$J_q^{j'}(q^j, u^j)(\delta q) + b'_q(q^j)(\delta q, z^j) = 0. \tag{17c}$$

These equations correspond to $\mathscr{L}'_{z^j} = 0, \mathscr{L}'_{u^j} = 0$ and $\mathscr{L}'_{q^j} = 0$ in (14). The variables u^j, z^j and q^j do now depend on s^j and λ^{j+1}. The BVP character of the intervalwise problems results from the forward-backward structure of the state and adjoint equations: s^j is the initial value for u^j at τ_j, and λ^{j+1} is the initial value for z^j at the subinterval endpoint τ_{j+1}.

Remark 6 It is an interesting problem in its own right how to choose s^j and λ^{j+1}, as the quality of the initial choice certainly influences the convergence of Newton's method. For ODE problems, there have been several suggestions; e.g., additional information on the solution, if available, could improve the initial guesses (for an example, see [8]). Alternatively, one could employ homotopy methods as in [23].

The states $u^j(s^j, \lambda^{j+1})$ and $z^j(s^j, \lambda^{j+1})$ are coupled via the control equation, and together the three equations yield the same structure on each subinterval as the global KKT system (8). However, the intervalwise solutions do not fit together at the subinterval endpoints, thus the solution is globally discontinuous due to the artificially chosen initial values s^j and λ^{j+1}. This contradicts the embedding $X \hookrightarrow C(\bar{I}; H)$. Therefore, we use the local solutions $u^j(s^j, \lambda^{j+1}), z^j(s^j, \lambda^{j+1})$ and $q^j(s^j, \lambda^{j+1})$ in order to update s^j and λ^{j+1} in the **second solution step**. This consists in solving the following system (corresponding to $\mathscr{L}'_{\lambda^j} = 0, \mathscr{L}'_{s^j} = 0$):

Algorithm 1 Indirect multiple shooting for PDE governed OCP

Require: Decomposition $I = \{\tau_0\} \cup \bigcup_{j=0}^{M-1} (\tau_j, \tau_{j+1}]$, initial values $\{(s_0^j, \lambda_0^{j+1})_{j=0}^{M-1}\}$.
1: Set $k = 1$.
2: **while** Shooting conditions (18) not fulfilled **do**
3: **for** $j = 0$ to $M - 1$ **do**
4: Solve intervalwise boundary value problems (17).
5: **end for**
6: Solve (19), compute update $\{(s_k^j, \lambda_k^{j+1})_{j=0}^{M-1}\}$ of initial values, set $k \leftarrow k + 1$.
7: **end while**

$$(s^0 - u_0, \delta\lambda) = 0, \tag{18a}$$

$$(\lambda^j - z^j(\tau_j; s^j, \lambda^{j+1}), \delta s) = 0, \quad (j = 0, \ldots, M-1) \tag{18b}$$

$$(s^j - u^{j-1}(\tau_j; s^{j-1}, \lambda^j), \delta\lambda) = 0, \quad (j = 1, \ldots, M) \tag{18c}$$

$$(\lambda^M, \delta s) = 0. \tag{18d}$$

These equations constitute the shooting system, which is the part of (14) we actually solve by Newton's method. Abbreviating the above shooting equations by $F(\mathbf{s}, \boldsymbol{\lambda}) = 0$, we thus have to solve

$$\nabla_{(\mathbf{s},\boldsymbol{\lambda})} F(\mathbf{s}, \boldsymbol{\lambda}) \cdot \begin{pmatrix} \delta\mathbf{s} \\ \delta\boldsymbol{\lambda} \end{pmatrix} = -F(\mathbf{s}, \boldsymbol{\lambda}). \tag{19}$$

This leads to improved initial values $\mathbf{s}^{\text{new}} = \mathbf{s} + \delta\mathbf{s}$, $\boldsymbol{\lambda}^{\text{new}} = \boldsymbol{\lambda} + \delta\boldsymbol{\lambda}$, with which we restart from step one described above. We see now the asserted structure of a two-step fixed-point iteration where we alternate between computing (u^j, z^j, q^j) and updating (s^j, λ^j). The whole process is resumed in the following Algorithm 1.

3.2.2 Algorithms for the Subproblems

We now focus on the two essential Steps 4 and 6 of Algorithm 1, namely the solution of the intervalwise BVP (17) (the first part of our two stage problem) and the solution of the shooting system (19) (the second part, correspondingly).

There are several possibilities how to solve the intervalwise BVP. In Algorithm 2, we present a so-called reduced approach which has also been implemented for our numerical examples. The important feature is the reduction of the set of independent variables from (q^j, u^j) to the control q^j alone, meaning that the interval state $u^j = u^j(q^j)$ is interpreted in terms of the interval control. To clarify the notation of Algorithm 2, we define the reduced cost functional (on subinterval I_j)

$$j(q^j) := J^j(q^j, u^j(q^j)). \tag{20}$$

Algorithm 2 Solution of the intervalwise BVP (reduced approach)

Require: Set $\nu = 0$, prescribe tolerance TOL_1 and initial control q_0^j.

while $\|\nabla j(q_\nu^j)\| > TOL_1$ **do**
  Solve state equation (17a).
  Solve adjoint equation (17b).
  Compute gradient $\nabla j(q_\nu^j)$ of reduced cost functional.
  Set $i = 0$, prescribe tolerance TOL_2 and $\delta q_{\nu,0}^j$.
  while $\|\delta q_{\nu,i+1}^j - \delta q_{\nu,i}^j\| > TOL_2$ **do**
    Compute matrix-vector product $\nabla^2 j(q_\nu^j)\delta q_{\nu,i}^j$.
    Solve system $\nabla^2 j(q_\nu^j)\delta q_{\nu,i}^j = -\nabla j(q_\nu^j)$ by a Newton-CG type method (this requires the solution of two additional equations per iteration; these so-called *tangent* and *additional adjoint equations* are obtained by linearization of (17a) and (17b)).
  end while
  Set $\nu \leftarrow \nu + 1$ and $q_{\nu+1}^j = q_\nu^j + \delta q_{\nu,end}^j$.
end while

Algorithm 3 Solution of the IMS shooting system (matrix-free approach)

Require: Shooting variables $(\mathbf{s}_k, \boldsymbol{\lambda}_k)$, intervalwise OCP solutions u^j, z^j

Build up residual $-F(\mathbf{s}_k, \boldsymbol{\lambda}_k)$.
Set $i = 0$, prescribe tolerance TOL and choose $(\delta\mathbf{s}_k^{(0)}, \delta\boldsymbol{\lambda}_k^{(0)})$.
while $\|\nabla F(\mathbf{s}_k, \boldsymbol{\lambda}_k)(\delta\mathbf{s}_k^{(i)}, \delta\boldsymbol{\lambda}_k^{(i)}) + F(\mathbf{s}_k, \boldsymbol{\lambda}_k)\| > TOL$ **do**
  Compute matrix-vector product $\nabla F(\mathbf{s}_k, \boldsymbol{\lambda}_k)(\delta\mathbf{s}_k^{(i)}, \delta\boldsymbol{\lambda}_k^{(i)})$.
  Solve system $\nabla F(\mathbf{s}_k, \boldsymbol{\lambda}_k)(\delta\mathbf{s}_k^{(i)}, \delta\boldsymbol{\lambda}_k^{(i)}) = -F(\mathbf{s}_k, \boldsymbol{\lambda}_k)$ by a Newton-GMRES type method (this requires the solution of two additional BVP, the linearizations of (17) w.r.t. $\mathbf{s}$ resp. $\boldsymbol{\lambda}$, in each iteration).
end while
Set $k \leftarrow k + 1$ and $\mathbf{s}_{k+1} = \mathbf{s}_k + \delta\mathbf{s}_k^{end}$, $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \delta\boldsymbol{\lambda}_k^{end}$.

For a more detailed presentation we refer to [9],where IMS for parabolic OCP has already been described thoroughly. We will meet the concept of reduced control problems again in Sect. 4 in the context of different DMS techniques.

The solution of (19) by Newton's method involves the Jacobian matrix $\nabla_{(\mathbf{s},\boldsymbol{\lambda})}F(\mathbf{s}, \boldsymbol{\lambda})$ of the shooting conditions (18). Despite having substantially reduced the size of the Newton system [in this regard, (19) must be compared to (16)], the effort for explicitly assembling this Jacobian is still not manageable. Therefore, we choose a matrix-free method (in our case a Newton-GMRES approach) to solve (19). Algorithm 3 comprises the essential steps. Details of the Krylov-Newton method addressed in Step 5 can again be found in [9]. The proceeding in the case of DMS is quite similar in many respects and will be discussed in more detail in the next section.

3.3 Direct Multiple Shooting

We have already seen in the introduction that almost all literature on multiple shooting for PDE governed OCP concentrates on DMS methods (see [13, 29, 31]). Furthermore, the 'classical' method mostly used for ODE optimal control introduced in [3–5] is a direct method. Many algorithmic and implementational details can be found, e.g., in [20]. Further developments of this method, especially considering specific condensing techniques, can be found for example in [6, 17–19, 27]. The application of the 'classical' DMS method to large scale systems can be found for example in [21, 26]. The DMS method presented here is substantially equivalent to this 'classical' approach, the difference consists in the reduced and unreduced strategies to solve the problem as will be shown in Sect. 4.

We will now introduce direct shooting focusing on algorithmic details that we have skipped in the IMS context.

3.3.1 Structure

We derive DMS along the very same lines as IMS by splitting the solution process of system (16) into two parts. However, DMS relies on a different regrouping of the variables which is illustrated by the following scheme:

$$\left(\begin{array}{cc|ccc} 0 & \mathscr{L}''_{uz} & \mathscr{L}''_{qz} & \mathscr{L}''_{sz} & 0 \\ \mathscr{L}''_{zu} & \mathscr{L}''_{uu} & 0 & 0 & \mathscr{L}''_{\lambda u} \\ \hline \mathscr{L}''_{zq} & 0 & \mathscr{L}''_{qq} & 0 & 0 \\ \mathscr{L}''_{zs} & 0 & 0 & 0 & \mathscr{L}''_{\lambda s} \\ 0 & \mathscr{L}''_{u\lambda} & 0 & \mathscr{L}''_{s\lambda} & 0 \end{array}\right) \left(\begin{array}{c} \delta z \\ \delta u \\ \hline \delta q \\ \delta s \\ \delta\lambda \end{array}\right) = - \left(\begin{array}{c} \mathscr{L}'_z \\ \mathscr{L}'_u \\ \hline \mathscr{L}'_q \\ \mathscr{L}'_s \\ \mathscr{L}'_\lambda \end{array}\right).$$

Thus, we now fix s^j, λ^j, and here also the controls q^j, and compute in the **first solution step** only the state and adjoint variables $u^j = u^j(q^j, s^j)$ and $z^j(q^j, s^j, \lambda^{j+1})$, which have now become dependent variables. This leads to the following initial value problems:

$$\begin{aligned} (\!(\partial_t u^j, \delta z)\!) + a(u^j)(\delta z) + b(q^j)(\delta z) - (\!(f|_{I_j}, \delta z)\!) \\ + (u^j(\tau_j) - s^j, \delta z(\tau_j)) = 0, \qquad (21a) \\ J''_u(q^j, u^j)(\delta u) - (\!(\partial_t z^j, \delta u)\!) + a'_u(u^j)(\delta u, z^j) \\ + (z^j(\tau_{j+1}) - \lambda^{j+1}, \delta u(\tau_{j+1})) = 0. \qquad (21b) \end{aligned}$$

Contrarily to the IMS case, these equations bear no BVP structure, because they are not fully coupled. We may in a first step compute the state solutions $u^j(q^j; s^j)$ and then use the result to compute the adjoint solutions $z^j(u^j(q^j; s^j); \lambda^{j+1})$, albeit

backward in time. The local IVP (21) correspond to the first two equation blocks of (14), $\mathscr{L}'_{z^j} = 0$ and $\mathscr{L}'_{u^j} = 0$.

In the current situation, we still have to solve the matching conditions $\mathscr{L}'_{s^j} = 0, \mathscr{L}'_{\lambda^j} = 0$ and the control equation $\mathscr{L}'_{q^j} = 0$, together constituting the **second solution step**. The resulting system that has to be solved by Newton's method reads

$$(s^0 - u_0, \delta\lambda) = 0, \tag{22a}$$

$$J_q^{0\prime}(q^0, u^0)(\delta q) + b'_q(q^0)(\delta q, z^0(\tau_j; q^0, s^0, \lambda^1)) = 0, \tag{22b}$$

$$(\lambda^j - z^j(\tau_j; q^j, s^j, \lambda^{j+1}), \delta s) = 0, \quad (j = 0, \ldots, M-1) \tag{22c}$$

$$(s^j - u^{j-1}(\tau_j; q^{j-1}, s^{j-1}), \delta\lambda) = 0, \quad (j = 1, \ldots, M) \tag{22d}$$

$$J_q^{j\prime}(q^j, u^j)(\delta q) + b'_q(q^j)(\delta q, z^j(\tau_j; q^j, s^j, \lambda^{j+1})) = 0, \quad (j = 0, \ldots, M-1) \tag{22e}$$

$$(\lambda^M, \delta s) = 0. \tag{22f}$$

Thus we have again reduced the size of the original Newton system (16), but here the resulting system is larger than in the IMS framework. Therefore, the first step in DMS consists of only solving the IVP (21) in contrast to the far more complicated BVP (17). Abbreviating (22) by $F(\mathbf{q}, \mathbf{s}, \boldsymbol{\lambda}) = 0$, we end up with Newton's equation

$$\nabla_{(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})} F(\mathbf{q}, \mathbf{s}, \boldsymbol{\lambda}) \cdot \begin{pmatrix} \delta\mathbf{q} \\ \delta\mathbf{s} \\ \delta\boldsymbol{\lambda} \end{pmatrix} = -F(\mathbf{q}, \mathbf{s}, \boldsymbol{\lambda}). \tag{23}$$

Altogether, the structure of DMS is again a two-step fixed-point iteration, where we first keep the controls and initial values fixed and compute (u^j, z^j) before updating (q^j, s^j, λ^j). We resume the solution process in the following Algorithm 4.

Algorithm 4 Direct multiple shooting for PDE governed OCP

Require: Decomp. $I = \{\tau_0\} \cup \bigcup_{j=0}^{M-1} (\tau_j, \tau_{j+1}]$, initial values and controls $\{(q_0^j, s_0^j, \lambda_0^{j+1})_{j=0}^{M-1}\}$.
1: Set $k = 1$.
2: **while** Shooting conditions (22) not fulfilled **do**
3: **for** $j = 0$ to $M - 1$ **do**
4: Solve intervalwise initial value problems (21).
5: **end for**
6: Solve (23), comp. update $\{(q_k^j, s_k^j, \lambda_k^{j+1})_{j=0}^{M-1}\}$ of initial values and controls, set $k \leftarrow k + 1$.
7: **end while**

3.3.2 Algorithm for Newton's Method

In contrast to the IMS case above, where we presented the details of the solution of both the intervalwise BVP (17) and the system (19) of shooting conditions, only Step 6 of Algorithm 4, i.e. the realization of Newton's method, is worth being discussed more thoroughly. The solution of the IVP in Step 4 is straightforward; the only important feature is that, on each subinterval, we have to first solve the state equation, because u^j is needed for solving the adjoint equation. With this restriction in mind, we turn our attention to Newton's system (23), the solution of which is different from that of the corresponding system in IMS and has not been presented elsewhere; therefore, we elaborate the following presentation in detail.

In Sect. 3.2, we stated that application of a matrix-free Krylov-Newton method is desirable due to the size of problem (19). We did not go into the details and referred to [9] instead. As the system (23) is even larger than (19) (due to the presence of the controls, see also Remark 12 below), a direct solver is even less advisable here. We will now discuss a Newton-GMRES method that has not been addressed in the DMS context before.

We see that the Jacobian $\nabla_{(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})}F(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})$ of (22) involves derivatives u_s^j, u_q^j of u^j w.r.t. s^j and q^j as well as derivatives $z_s^j, z_\lambda^j, z_q^j$ of z^j w.r.t. s^j, λ^{j+1} and q^j. These derivatives, the so-called sensitivities, are obtained by solving five additional (linearized) IVP, the sensitivity equations (also known as variational equations), for $j \in \{0,\dots,M-1\}$. First, we differentiate (21a) where $u^j = u^j(s^j,q^j)$ w.r.t. s^j in direction δs and w.r.t. q^j in direction δq to obtain the equations

$$((\partial_t u_s^j,\varphi)) + a_u'(u^j)(u_s^j,\varphi) + (u_s^j(\tau_j) - \delta s^j, \varphi(\tau_j)) = 0, \tag{24a}$$

$$((\partial_t u_q^j,\varphi)) + a_u'(u^j)(u_q^j,\varphi) + b_q'(q^j)(\delta q^j,\varphi) + (u_q^j(\tau_j),\varphi(\tau_j)) = 0, \tag{24b}$$

which have to hold for all $\varphi \in X^j$. Having solved these problems (for given initial data $\delta s^{j,0}$ and $\delta q^{j,0}$), we end up with u_s^j, u_q^j which can now be inserted into the following three IVP that are obtained by differentiation of the adjoint equation (21b) w.r.t. all its arguments in corresponding directions and must hold for all $\psi \in X^j$:

$$\begin{aligned} J_{uu}^{j\prime\prime}(q^j,u^j)(u_s^j,\psi) - ((\partial_t z_s^j,\psi)) + a_{uu}''(u^j)(u_s^j,\psi,z^j) \\ + a_u'(u^j)(\psi,z_s^j) + (z_s^j(\tau_{j+1}),\psi(\tau_{j+1})) = 0, \end{aligned} \tag{25a}$$

$$\begin{aligned} J_{uu}^{j\prime\prime}(q^j,u^j)(u_q^j,\psi) - ((\partial_t z_q^j,\psi)) + a_{uu}''(u^j)(u_q^j,\psi,z^j) \\ + a_u'(u^j)(\psi,z_q^j) + (z_q^j(\tau_{j+1}),\psi(\tau_{j+1})) = 0, \end{aligned} \tag{25b}$$

$$-((\partial_t z_\lambda^j,\psi)) + a_u'(u^j)(\psi,z_\lambda^j) + (z_\lambda^j(\tau_{j+1}) - \delta\lambda^{j+1},\psi(\tau_{j+1})) = 0. \tag{25c}$$

Solving these problems with initial data $\delta\lambda^{j+1,0}$ leaves us with a complete set of sensitivities, but only w.r.t. the chosen initial values $(\delta q^{j,0}, \delta s^{j,0}, \delta\lambda^{j+1,0})$. In order to assemble $\nabla_{(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})}F$ explicitly, we have to solve the sensitivity equations for a

whole basis of $\bigcup_{j=0}^{M-1}[Q^j \times H \times H]$, which is numerically expensive for fine temporal or spatial discretizations. Therefore, we choose an adjoint approach (matrix-free) where we handle Newton's system (23) with an iterative solver, for which we choose in our case, due to the asymmetric structure of the matrix, a GMRES method. We then have to solve the sensitivity equations only once per GMRES iteration. The adjoint approach thus avoids assembling the Jacobian and operates on the matrix-vector product $\nabla_{(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})}F(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})\cdot(\delta\mathbf{q},\delta\mathbf{s},\delta\boldsymbol{\lambda})^\top$ instead.

This matrix-vector product, the left-hand side of (23), has the concrete form

$$\nabla_{(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})}F(\mathbf{q},\mathbf{s},\boldsymbol{\lambda})\cdot\begin{pmatrix}\delta\mathbf{q}\\ \delta\mathbf{s}\\ \delta\boldsymbol{\lambda}\end{pmatrix}=\left(\begin{array}{c}\delta s^0\\ J_{qq}^{0\prime\prime}(q^0,u^0)(\delta q^0)+b_{qq}^{\prime\prime}(q^0)(\delta q^0,z^0)\\ \qquad +b_q^{\prime}(q^0)[z_s^0(\delta s^0)+z_q^0(\delta q^0)+z_\lambda^0(\delta\lambda^0)]\\ \hline \delta\lambda^j-z_s^j(\tau_j;\delta s^j)-z_\lambda^j(\tau_j;\delta\lambda^{j+1})\\ \delta s^{j+1}-u_s^j(\tau_{j+1};\delta s^j)-u_\lambda^j(\tau_{j+1};\delta\lambda^{j+1})\\ J_{qq}^{j\prime\prime}(q^j,u^j)(\delta q^j)+b_{qq}^{\prime\prime}(q^j)(\delta q^j,z^j)\\ \qquad +b_q^{\prime}(q^j)[z_s^j(\delta s^j)+z_q^j(\delta q^j)+z_\lambda^j(\delta\lambda^j)]\\ \hline \delta\lambda^M\end{array}\right)$$

where the middle part has to be interpreted for $j=0,\ldots,M-1$; note the index shift we performed in the second component of the middle part to keep the presentation consistent. We can now formulate the following Algorithm 5 which yields the details of Step 6 of the above DMS algorithm:

Algorithm 5 Solution of the DMS shooting system (matrix-free approach)

Require: Shooting variables $(\mathbf{s}_k,\boldsymbol{\lambda}_k)$ and controls $\mathbf{q}_k$, intervalwise OCP solutions u^j, z^j

1: Build up residual $-F(\mathbf{q}_k,\mathbf{s}_k,\boldsymbol{\lambda}_k)$.
2: Set $i=0$, prescribe tolerance TOL and choose $(\delta\mathbf{q}_k^{(0)},\delta\mathbf{s}_k^{(0)},\delta\boldsymbol{\lambda}_k^{(0)})$.
3: **while** $\|\nabla F(\mathbf{q}_k,\mathbf{s}_k,\boldsymbol{\lambda}_k)(\delta\mathbf{q}_k^{(i)},\delta\mathbf{s}_k^{(i)},\delta\boldsymbol{\lambda}_k^{(i)})+F(\mathbf{q}_k,\mathbf{s}_k,\boldsymbol{\lambda}_k)\|>TOL$ **do**
4: Compute matrix-vector product $\nabla F(\mathbf{q}_k,\mathbf{s}_k,\boldsymbol{\lambda}_k)(\delta\mathbf{q}_k^{(i)},\delta\mathbf{s}_k^{(i)},\delta\boldsymbol{\lambda}_k^{(i)})$ by solving the state and adjoint sensitivity equations (24) and (25).
5: Solve system $\nabla F(\mathbf{q}_k,\mathbf{s}_k,\boldsymbol{\lambda}_k)(\delta\mathbf{q}_k^{(i)},\delta\mathbf{s}_k^{(i)},\delta\boldsymbol{\lambda}_k^{(i)})=-F(\mathbf{q}_k,\mathbf{s}_k,\boldsymbol{\lambda}_k)$ by a Newton-GMRES type method (this requires the renewed solution of (24) and (25) in each iteration).
6: **end while**
7: Set $k\leftarrow k+1$ and $\mathbf{q}_{k+1}=\mathbf{q}_k+\delta\mathbf{q}_k^{end},\mathbf{s}_{k+1}=\mathbf{s}_k+\delta\mathbf{s}_k^{end},\boldsymbol{\lambda}_{k+1}=\boldsymbol{\lambda}_k+\delta\boldsymbol{\lambda}_k^{end}$.

3.4 Comparison of Both Approaches

From the last two Sects. 3.2 and 3.3 we see that IMS and DMS on a purely algebraic level, i.e. looking merely at the equations to be solved [which are in both cases the extended KKT conditions (14)], are in fact equivalent. The differences are due to the respective algorithmic realizations that result from the different splittings of system (16) leading to different internal dependencies of the arguments of the common starting point, the extended formulation of the optimal control problem (10)–(11).

Remark 7 Note that there are further possibilities how to split the set of arguments of the extended Lagrangian (13), although they might not work as well as the ones discussed so far. This might give reason to further research.

In the algorithmic description, we concentrated on the solution of the shooting systems (19) resp. (23) by Newton's method, which may lead to the supposition that DMS is more expensive (in this context, see also Remark 12 below). However, in the DMS context we only have to solve a (nonlinear) IVP (21) on each subinterval, and also the additional sensitivity equations are only intervalwise (linear) IVP, which can be solved in a rather straightforward manner. Despite the IMS shooting system (19) being much smaller than the DMS one, the solution of the (nonlinear) subinterval problems (17) and the corresponding (linearized) problems for the Newton-GMRES method necessitates large effort, because they constitute smaller versions of the original OCP and its linearization. It is thus not clear a priori which of the two methods (IMS or DMS) is to be preferred.

Both IMS and DMS still comprise a variety of algorithms, depending on how we solve the subinterval problems (reduced approach as above vs. all-at-once approach), on how we solve Newton's system (iterative matrix-free solver as above vs. direct solver, inclusion of globalization techniques or of an SQP-like inexact Newton method), on how we solve the sensitivity equations (simply by a fixed-point method or by more sophisticated approaches) etc.

We will now turn our attention to some of these differences which are responsible for the seemingly large differences between 'classical' DMS (introduced in Sect. 4.1 for parabolic OCP) and the DMS approach given in Algorithm 4.

4 Variants of DMS

The following considerations were initiated by the observation that the DMS method derived in Sect. 3.3, despite being called 'direct', seems to be rather an indirect method judged by the standard classification (see Introduction), because it is based on the optimality conditions (14), i.e., the optimization has already been done when shooting comes into play. According to our classification discussed in the Introduction, the DMS approach shown here is a direct approach since the control

is included in the shooting system. Therefore, it benefits from the advantages of direct methods in terms of convergence as shown in our nonlinear example Sect. 6.2. Further study to compare IMS and DMS in the PDE context is needed, but is left for future work, especially considering state and control constraints.

Typical implementations of direct methods [1, 28], that use the 'first discretize, then optimize' approach, avoid the use of an adjoint equation either by using a sensitivity approach or by applying automatic differentiation techniques. We therefore present in Sect. 4.1 the 'classical' DMS approach but tailored to the parabolic situation, and show in Sect. 4.2 that this approach, relying on a reduced formulation of the extended optimal control problem (10)–(11), is equivalent to our variant of DMS. In the classical framework, the reduced problem is discretized, and the resulting nonlinear programming problem (NLP) is usually solved by an all-at-once approach, e.g. a sequential quadratic programming (SQP) method. In contrast, our DMS variant relies on a continuous unreduced formulation. The dichotomy of reduced versus unreduced approaches is discussed in [16] for the global OCP (2)–(3), and we use the notational framework established there. Note that the whole discussion takes place on the abstract function space level.

4.1 DMS Based on a Reduced Form of the Extended OCP Formulation

We will now embed a DMS method that was developed (mainly by Bock and his co-workers) in the 1980s (see, e.g. [3–5]) into the context of parabolic OCP.

DMS methods for problem (10)–(11) are usually based on a reformulation of this extended OCP formulation completely in terms of the primal shooting variables s^j and the intervalwise controls q^j, i.e. $u^j = u^j(q^j, s^j)$. Pursuing this strategy, we end up with the minimization problem

$$\min_{(\mathbf{q},\mathbf{s})} \overline{J}(\mathbf{q},\mathbf{s}) := \sum_{j=0}^{M-1} J^j(q^j, u^j(q^j, s^j)) \tag{26a}$$

$$\text{s. t.} \qquad s^0 - u_0 = 0, \tag{26b}$$

$$s^{j+1} - u^j(\tau_{j+1}; q^j, s^j) = 0, \tag{26c}$$

where (26c) comprises the continuity conditions for $j = 0, \ldots, M-1$. We call (26) a reduced formulation of the extended OCP formulation (10)–(11). It is formulated in terms of the independent variables q^j and s^j and relies upon the solution of the IVP

$$e^j(q^j, s^j, u^j(q^j, s^j)) = \begin{pmatrix} \partial_t u^j(q^j, s^j) + \mathscr{A}(u^j(q^j, s^j)) + \mathscr{B}(q^j) - f|_{I_j} \\ u^j(\tau_j; q^j, s^j) - s^j \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{27}$$

that has been solved on all subintervals I_j for $j = 0, \dots, M-1$. We assume unique solvability of the subinterval problems which implies the existence of a solution operator mapping $Q^j \times H$ to X^j. In (27), $e^j(q^j, s^j, u^j(q^j, s^j))$ is an intervalwise counterpart of the abstract side condition in (1) which is, in contrast to the preceding sections, again strongly formulated. This abstract notation helps us to keep the proof of the equivalence result in the next section short.

Remark 8 In the reformulation (26)–(27) of the extended OCP formulation, the local control variable $q^j(x,t)$ is a function of both spatial variables x and time t. In DMS methods for ODE control problems which depend only on t, the control is usually parameterized as a piecewise polynomial of order $p \leq 3$ on the subintervals I_j, i.e. $q^j \equiv q^j(q_0^j, \cdots, q_p^j)$ (see, e.g., [20]). This parameterization of the control saves a large amount of computing time and storage (up to four control parameters per shooting interval I_j as opposed to a number of control values on each I_j determined by the control discretization, which is usually much finer than the mentioned parameterization). Furthermore, so-called condensing techniques (reducing the shooting system to the control variables) are frequently employed; they are not efficient if q is discretized on a similarly fine level as the state u. However, reducing the control to a much smaller space by parameterization leads to only suboptimal solutions of the given control problems. In the PDE case, a parameterization of the control $q(x,t)$ may lead to a loss of structural information in the spatial variables. Some ideas in this regard are briefly discussed in Sect. 5 (see Remark 11 below).

Starting from (26), we derive the corresponding Lagrange functional (introducing a Lagrange multiplier $\boldsymbol{\lambda} = (\lambda^j)_{j=0}^M \in H^{m+1}$):

$$\mathscr{L}(\mathbf{q}, \mathbf{s}, \boldsymbol{\lambda}) = \overline{J}(\mathbf{q}, \mathbf{s}) + (s^0 - u_0, \lambda^0) + \sum_{j=0}^{M-1} (s^{j+1} - u^j(\tau_{j+1}; q^j, s^j), \lambda^{j+1}). \tag{28}$$

We obtain the (reduced) optimality system as usual by differentiation w.r.t. the arguments $(\mathbf{q}, \mathbf{s}, \boldsymbol{\lambda})$. This yields the system [where $j \in \{0, \dots, M-1\}$ in Eqs. (29b), (29c) and (29e)]

$$\mathscr{L}'_{\lambda^0}(\delta\lambda) = (s^0 - u_0, \delta\lambda), \tag{29a}$$

$$\mathscr{L}'_{\lambda^j}(\delta\lambda) = (s^{j+1} - u^j(\tau_{j+1}; q^j, s^j), \delta\lambda), \tag{29b}$$

$$\mathscr{L}'_{s^j}(\delta s) = \langle J_u^{j\prime}, u_s^{j\prime}(\delta s)\rangle_{X^{j*} \times X^j} + (\lambda^j, \delta s) - (\lambda^{j+1}, u_s^{j\prime}[\tau_{j+1}](\delta s)), \tag{29c}$$

$$\mathscr{L}'_{s^M}(\delta s) = (\lambda^M, \delta s), \tag{29d}$$

$$\mathscr{L}'_{q^j}(\delta q) = \langle J_q^{j\prime}, \delta q\rangle_{Q^{j*} \times Q^j} + \langle J_u^{j\prime}, u_q^{j\prime}(\delta q)\rangle_{X^{j*} \times X^j} - (\lambda^{j+1}, u_q^{j\prime}[\tau_{j+1}](\delta q)). \tag{29e}$$

Here, $u_q^{j'} : Q^j \to X^j$ and $u_s^{j'} : H \to X^j$ are operators mapping the controls and initial values to the respective (variational) states $u_{q/s}^{j'}$ that are denoted by the same symbols. Furthermore, the notation $u_{q/s}^{j'}[\tau_{j+1}]$ means that the respective variational state obtained by application of the operator $u_{q/s}^{j'}$ is evaluated at time-point τ_{j+1}. The classical DMS method consists in the solution of system (29). In the framework of ODE optimal control, [20] gives a detailed description of SQP methods that solve (29) without employing an adjoint equation. Therefore, either the Jacobian of (29) has to be assembled, or additional sophisticated algorithms are needed to circumvent this matrix assembly. An alternative matrix-free SQP approach has been proposed by Ulbrich [31]. This procedure corresponds to the sensitivity approach for generating derivative information that is needed during the solution process; this sensitivity approach is usually too expensive in PDE optimal control, because one has to solve an additional linearized problem for each basis vector δq of the (discrete) control space (see [16] or [24]).

4.2 Equivalence of the Two DMS Approaches

Comparing the two DMS variants presented in Sects. 3.3 and 4.1 yields, besides several minor differences, a central distinction that influences the structure of the solution process strongly. It consists of the absence of any adjoint problem in the DMS method from Sect. 4.1, whereas the variant discussed in Sect. 3.3 is based on the same full optimality system (14) as the IMS approach of Sect. 3.2, including the adjoint part.

Remark 9 Generally, modern implementations of DMS for ODE optimal control problems, which are capable of handling parabolic OCP by transforming the PDE side condition into a huge ODE system via the method of lines (MOL) approach, make use of adjoint methods for sensitivity generation, which constitute a suitable alternative to the above described approach. They often compute the adjoint equations (which may be nasty to derive by hand in case of large ODE systems with complicated nonlinear terms) by automatic differentiation (see [1]).

The following theorem, which is the main result of this section, shows the equivalence of the two DMS approaches. Moreover, the proof reveals that the seemingly so different DMS variant of Sect. 3.3 is merely a reformulation of 'classical' DMS by means of an adjoint approach for sensitivity generation. It is performed in an abstract function space setting, meaning that the argumentation is not affected by discretization.

Theorem 2 *The solution of the reduced formulation* (26) *of the modified OCP* (10)–(11) *by means of an adjoint approach leads to the DMS method described in Sect. 3.3.*

The following outline prepares the proof of Theorem 2. Classical DMS for problem (26) relies upon the solution of system (27) and necessitates the solution of (29). Analogously, the DMS approach from Sect. 3.3 for problem (10)–(11) relies upon (21) having been solved and necessitates the solution of (22). Comparing the two settings, the following correspondences are evident: (27) is the strong formulation of (21a), and (29a), (29b) and (29d) are identical to (22a), (22d) and (22f), respectively. It is thus our goal to derive the adjoint equation (21b), the continuity conditions (22c) and the control equations (22b) and (22e) from (29c) and (29e). To achieve this, we extend the ideas and techniques of Sect. 1.6 from [16] to the more complex multiple shooting situation. The following proof has already been outlined in [12].

Proof For the following discussion, we introduce adjoint operators $u_q^{j\prime *} : X^{j*} \to Q^{j*}$ and $u_s^{j\prime *} : X^{j*} \to H^* \equiv H$ (as well as their restrictions to the final time-point, $u_q^{j\prime *}[\tau_{j+1}] : H \equiv H^* \to Q^{j*}$ and $u_s^{j\prime *}[\tau_{j+1}] : H \equiv H^* \to H^* \equiv H$) corresponding to the differential operators $u_q^{j\prime} : Q^j \to X^j$ and $u_s^{j\prime} : H \to X^j$ and rewrite Eqs. (29c) and (29e) in an abstract adjoint form:

$$\mathscr{L}'_{s^j}(\delta s) = (u_s^{j\prime *}(J_u^{j\prime}), \delta s) + (\lambda^j, \delta s) - (u_s^{j\prime *}[\tau_{j+1}](\lambda^{j+1}), \delta s), \tag{29c*}$$

$$\begin{aligned}\mathscr{L}'_{q^j}(\delta q) = {} & \langle J_q^{j\prime}, \delta q\rangle_{Q^{j*}\times Q^j} + \langle u_q^{j\prime *}(J_u^{j\prime}), \delta q\rangle_{Q^{j*}\times Q^j} \\ & - \langle u_q^{j\prime *}[\tau_{j+1}](\lambda^{j+1}), \delta q\rangle_{Q^{j*}\times Q^j}.\end{aligned} \tag{29e*}$$

We discuss the adjoint operators first on an abstract level which enables a clear presentation of the formal framework. By differentiating the interval state equations (27) w.r.t. q^j in direction δq and w.r.t. s^j in direction δs, we obtain (the arguments $(q^j, s^j, u^j(q^j, s^j))$ are omitted for brevity)

$$e_u^{j\prime}(\delta u_q) = -e_q^{j\prime}(\delta q), \quad e_u^{j\prime}(\delta u_s) = -e_s^{j\prime}(\delta s). \tag{30}$$

Here, δu_q and δu_s are abbreviations for $u_q^j(\delta q)$ and $u_s^j(\delta s)$, respectively. Assuming $e_u^{j\prime}$ to have a bounded inverse, we use the implicit function theorem to obtain

$$u_q^{j\prime} = -(e_u^{j\prime})^{-1} \circ e_q^{j\prime}, \quad u_s^{j\prime} = -(e_u^{j\prime})^{-1} \circ e_s^{j\prime}.$$

Now, the definition of adjoint operators gives us the following representation:

$$u_q^{j\prime *} = -e_q^{j\prime *} \circ (e_u^{j\prime})^{-*}, \quad u_s^{j\prime *} = -e_s^{j\prime *} \circ (e_u^{j\prime})^{-*}.$$

Inserting these expressions for $u_s^{j\prime *}$ and $u_q^{j\prime *}$ into the corresponding terms of (29c*) and (29e*), we get

$$(u_s^{j\prime *}(J_u^{j\prime}), \delta s) = -(e_s^{j\prime *}((e_u^{j\prime})^{-*}(J_u^{j\prime})), \delta s), \tag{31a}$$

$$(u_s^{j'^*}[\tau_{j+1}](\lambda^{j+1}), \delta s) = -(e_s^{j'^*}((e_u^{j'})^{-*}[\tau_{j+1}](\lambda^{j+1})), \delta s), \tag{31b}$$

$$\langle u_q^{j'^*}(J_u^{j'}), \delta q\rangle_{Q^{j*}\times Q^j} = -\langle e_q^{j'^*}((e_u^{j'})^{-*}(J_u^{j'})), \delta q\rangle_{Q^{j*}\times Q^j}, \tag{31c}$$

$$\langle u_q^{j'^*}[\tau_{j+1}](\lambda^{j+1}), \delta q\rangle_{Q^{j*}\times Q^j} = -\langle e_q^{j'^*}((e_u^{j'})^{-*}[\tau_{j+1}](\lambda^{j+1})), \delta q\rangle_{Q^{j*}\times Q^j}. \tag{31d}$$

We notice that, in (31a) and (31c), the same argument $(e_u^{j'})^{-*}(J_u^{j'})$ is inserted into both operators $e_s^{j'^*}$ and $e_q^{j'^*}$. The same holds for $(e_u^{j'})^{-*}[\tau_{j+1}](\lambda^{j+1})$ in (31b) and (31d). We define the variables $z_J^j := -(e_u^{j'})^{-*}(J_u^{j'})$ and $z_\lambda^j := (e_u^{j'})^{-*}[\tau_{j+1}](\lambda^{j+1})$, which then fulfil the following equations, respectively:

$$e_u^{j'^*}(z_J^j) = -J_u^{j'}, \qquad e_u^{j'^*}[\tau_{j+1}](z_\lambda^j) = \lambda^{j+1}. \tag{32}$$

These are the (formal) adjoint equations; we will see below that they can be merged into one equation, due to the linearity of the operator $e_u^{j'^*}$ and a superposition principle, which interprets z_J^j as a solution belonging to a homogeneous initial value and non-homogeneous right-hand side $-J_u^{j'}$, and z_λ^j as a solution belonging to a homogeneous right-hand side and non-homogeneous initial value λ^{j+1}.

In our concrete situation, we start from the weak formulation of (27) given by

$$((\partial_t u^j, \varphi)) + a(u^j)(\varphi) + b(q^j)(\varphi) - ((f|_{I_j}, \varphi)) + (u^j(\tau_j) - s^j, \varphi(\tau_j)) = 0. \tag{33}$$

The differential operator $u_q^{j'} : Q^j \to X^j$ mentioned above is the solution operator of the following linearized equation (which is the derivative of (33) w.r.t. q^j in direction δq):

$$((\partial_t \delta u_q, \varphi)) + a_u'(u^j)(\delta u_q, \varphi) + (\delta u_q(\tau_j), \varphi(\tau_j)) = -b_q'(q^j)(\delta q, \varphi). \tag{34}$$

Analogously, $u_s^{j'} : H \to X^j$ is the solution operator of the derivative of (33) w.r.t. s^j in direction δs:

$$((\partial_t \delta u_s, \varphi)) + a_u'(u^j)(\delta u_s, \varphi) + (\delta u_s(\tau_j), \varphi(\tau_j)) = (\delta s, \varphi(\tau_j)). \tag{35}$$

Here, δu_q and δu_s denote the respective solution variables. We call (34) and (35) the sensitivity equations belonging to (27). They correspond to the formal equations (30).

Now we are able to substantiate the adjoint equations (32). We have seen above that the application of the adjoint operator $u_q^{j'^*}$ resp. $u_s^{j'^*}$ to a functional $J_u^{j'} \in X^{j^*}$ (or of $u_q^{j'^*}[\tau_{j+1}]$ resp. $u_s^{j'^*}[\tau_{j+1}]$ to λ^{j+1}) corresponds to carrying out the following steps:

1. Solve the adjoint equation $e_u^{j'^*}(z_J^j) = -J_u^{j'}$ (or $e_u^{j'^*}[\tau_{j+1}](z_\lambda^j) = \lambda^{j+1}$).
2. Apply the adjoint operators $e_q^{j'^*}$ resp. $e_s^{j'^*}$ to the solution z_J^j (or z_λ^j).

Discussing the first step will lead us to the adjoint equation, while the second step yields the continuity conditions for the adjoint equation and the control equations. The general adjoint equation corresponding to both (34) and (35) is given by

$$-((\partial_t \delta u^*_{q/s}, \psi)) + a'_u(u^j)(\delta u^*_{q/s}, \psi) + (\delta u^*_{q/s}(\tau_{j+1}), \psi(\tau_{j+1})) = \mathrm{rhs}(\psi). \tag{36}$$

Here, the term $\mathrm{rhs}(\psi)$ which is fanning the dynamics represents either a distributed source term or an end-time initial condition. Our situation comprises the following two adjoint equations (where the abstract adjoint variable $\delta u^*_{q/s}$ has been suitably replaced):

$$-((\partial_t z^j_J, \psi)) + a'_u(u^j)(z^j_J, \psi) + (z^j_J(\tau_{j+1}), \psi(\tau_{j+1})) = -J^{j\prime}_u(q^j, u^j)(\psi), \tag{37a}$$

$$-((\partial_t z^j_\lambda, \psi)) + a'_u(u^j)(z^j_\lambda, \psi) + (z^j_\lambda(\tau_{j+1}), \psi(\tau_{j+1})) = (\lambda^{j+1}, \psi(\tau_{j+1})). \tag{37b}$$

Evidently, Eq. (37) are both fully linear, as the possibly nonlinear operators $a(\cdot)(\cdot)$ and $J^j(\cdot)$ enter only in linearized form. Therefore, we may write (37a) and (37b) as one single equation by defining $z^j := z^j_J - z^j_\lambda$. The resulting adjoint equation reads

$$\begin{aligned} J^{j\prime}_u(q^j, u^j)(\psi) - ((\partial_t z^j, \psi)) + a'_u(u^j)(z^j, \psi) \\ + (z^j(\tau_{j+1}) - \lambda^{j+1}, \psi(\tau_{j+1})) = 0. \end{aligned} \tag{38}$$

A comparison of (21b) and (38) shows that, substituting the test function ψ by δu, our first objective, namely the introduction of the adjoint equation into the reduced DMS method by means of an adjoint approach to sensitivity generation, has been achieved.

We finally explain the second step of the above proceeding in detail. By means of the described superposition $z^j := z^j_J - z^j_\lambda$, system (31) diminishes to

$$(u^{j\prime *}_s(J^{j\prime}_u) - u^{j\prime *}_s[\tau_{j+1}](\lambda^{j+1}), \delta s) = -(e^{j\prime *}_s(z^j), \delta s), \tag{39a}$$

$$\langle u^{j\prime *}_q(J^{j\prime}_u) - u^{j\prime *}_q[\tau_{j+1}](\lambda^{j+1}), \delta q\rangle_{Q^{j*}\times Q^j} = -\langle e^{j\prime *}_q(z^j), \delta q\rangle_{Q^{j*}\times Q^j}, \tag{39b}$$

where the adjoint solution has been inserted into the right-hand side terms. Since the right-hand sides of the second equation of (30) and of (35) coincide, we get the following equalities (making use of the weak form $e^{j\prime}_s(\delta s)(\varphi) := (e^{j\prime}_s(\delta s), \varphi)$):

$$(\delta s, e^{j\prime *}_s(z^j)) = (e^{j\prime}_s(\delta s), z^j) = e^{j\prime}_s(\delta s)(z^j) = (\delta s, z^j(\tau_j)).$$

Thus, replacing (for all $j \in \{0, \ldots, M-1\}$) the adjoint terms in (29c*) by the corresponding term in (39a) and using the last equality, we end up with

$$\mathscr{L}'_{sj}(\delta s) = (\lambda^j, \delta s) - (z^j(\tau_j), \delta s), \tag{40}$$

which is exactly the adjoint continuity condition (22c). Analogously, we can exploit (39b). The right-hand sides of the first equation of (30) and of (34) coincide, which leaves us with

$$\langle e_q^{j\prime *}(z^j), \delta q\rangle_{Q^{j*}\times Q^j} = \langle e_q^{j\prime}(\delta q), z^j\rangle_{X^{j*}\times X^j} = e_q^{j\prime}(\delta q)(z^j) = -b_q'(q^j)(\delta q, z^j).$$

Here, we have made use of the definition $\langle e_q^{j\prime}(\delta q), \varphi\rangle_{X^{j*}\times X^j} := e_q^{j\prime}(\delta q)(\varphi)$. We can now replace the adjoint terms in (29e*) by the corresponding term in (39b), use the last equality and obtain for all $j \in \{0, \dots, M-1\}$:

$$\mathscr{L}_{q^j}'(\delta q) = \langle J_q^{j\prime}, \delta q\rangle_{Q^{j*}\times Q^j} + b_q'(q^j)(\delta q, z^j). \tag{41}$$

Since the last relationship is identical to (22e) [and, in the case $j = 0$, to (22b)], this completes the proof. □

5 Discretization

For the sake of completeness, we briefly present the discretization schemes that will be used for computing the examples in Sect. 6 below. We treat the discretization of the time variable and the spatial variables separately, but choose the same discretization schemes for IMS and DMS.

5.1 Time Semi-Discretization

The decomposition (9) of the solution interval I into subintervals $I_j = (\tau_j, \tau_{j+1}]$ is not to be seen as a discretization, but rather as a reformulation of the problem. As we have seen, it is compensated by introducing the additional equality constraints (11). Here, we describe a further decomposition of each shooting interval I_j into smaller time intervals $I_j^n = (t_j^n, t_j^{n+1}]$ of length $k_j^n := t_j^{n+1} - t_j^n$ with timepoints

$$\tau_j = t_j^0 < t_j^1 < \cdots < t_j^{N_j} = \tau_{j+1},$$

which is now an actual discretization of the time variable. On each shooting interval, we use the discontinuous Galerkin method of order r (dG(r) method). All semi-discrete variables are indexed with a symbol k denoting a piecewise constant function $k|_{I_j^n} := k_j^n$. We denote space-time scalar products and semilinear forms on I_j^n by $((\cdot,\cdot))_n$, $a_n(\cdot)(\cdot)$ and $b_n(\cdot)(\cdot)$, suppressing the shooting interval index j. The same holds for the functional $J_n^j(\cdot)$.

Starting point for the dG(r) method is the space of semi-discrete functions

$$X_k^r(I_j) := \{v_k \in L^2(I_j; V) \mid v_k(t_j^0) \in H, v_k|_{I_j^n} \in P_r(I_j^n; V), n = 1, \ldots, N_j\}$$

where the space $P_r(I_j^n)$ contains all polynomials on I_j^n up to degree r. The time-discrete functions on I_j may be discontinuous at the timepoints t_n, hence we introduce the following notation for describing the jumps:

$$v_{k,n}^{j,+} := \lim_{t \searrow 0} v_k(t_j^n + t), \quad v_{k,n}^{j,-} := \lim_{t \nearrow 0} v_k(t_j^n + t), \quad [v_k^j]_n := v_{k,n}^{j,+} - v_{k,n}^{j,-}.$$

Now we are prepared to formulate the time semi-discrete state and adjoint equations: Find $u_k^j, z_k^j \in X_k^r(I_j)$, such that for all $\delta z_k, \delta u_k \in X_k^r(I_j)$, the following equations hold:

$$\sum_{n=0}^{N_j-1} \left[((\partial_t u_k^j, \delta z_k))_n + a_n(u_k^j)(\delta z_k) + b_n(q^j)(\delta z_k) - ((f|_{I_j^n}, \delta z_k))_n \right] + \sum_{n=1}^{N_j-1} ([u_k^j]_n, \delta z_{k,n}^+) + (u_k^j(\tau_j) - s^j, \delta z_k(\tau_j)) = 0, \quad (42a)$$

$$\sum_{n=0}^{N_j-1} \left[\overline{J}_{n,u}^{j'}(q^j, u_k^j)(\delta u_k) - ((\partial_t z_k^j, \delta u_k))_n + a'_{n,u}(u_k^j)(\delta u_k, z_k^j) \right] - \sum_{n=1}^{N_j-1} ([z_k^j]_n, \delta u_{k,n}^-) + (z_k^j(\tau_{j+1}) - \lambda^{j+1}, \delta u_k(\tau_{j+1})) = 0. \quad (42b)$$

Additional linearized equations needed to compute sensitivities (e.g., the tangent and additional adjoint equations in the IMS case) are discretized analogously. In the context of parabolic OCP without shooting, this is described in detail in [2]. Note that the control has not been discretized.

Remark 10 We will only consider the case $r = 0$, where the Galerkin method is equivalent to the backward Euler time-stepping scheme up to a quadrature error induced by the box rule. To keep the presentation short, we omit the discussion of alternatives like the continuous Galerkin method of order r (cG(r) method); details can be found, e.g., in [24].

5.2 Space-Time Discretization

For discretizing the spatial variables, we use a conforming finite element method on a shape-regular mesh $\mathcal{T}_h$ decomposing the domain $\overline{\Omega}$ into closed cells K (for definitions and details we refer to, e.g., the textbook [7]). Spatially discrete

quantities are indexed by h, where $h|_K := h_K$ is the (cellwise constant) diameter of K. On the mesh $\mathscr{T}_h$, we define the space $V_h^s \subset V$ of finite element functions of polynomial degree s by

$$V_h^s := \left\{ v_h \,\middle|\, v_h|_K \in Q^s(K), K \in \mathscr{T}_h \right\}.$$

We denote by $Q^s(K)$ the space of functions that result from isoparametric transformations of polynomials defined on a reference unit cell $\hat{K}$. As we only consider the case $s = 1$, we exclusively deal with bilinear transformations. To formulate the fully discretized problem, we need the function space

$$X_{k,h}^{r,s}(I_j) := \{ v_{kh} \in L^2(I_j; V_h^s) \mid v_{kh}(t_j^0) \in V_h^s, v_{kh}|_{I_j^n} \in P_r(I_j^n; V_h^s), n = 1, \ldots, N_j \}$$

which consists of all piecewise polynomials of degree r on the time intervals with values in the finite element space V_h^s. The space-time discrete problem now consists in finding $u_{kh}, z_{kh} \in X_{k,h}^{r,s}(I_j)$, such that for all $\delta u_{kh}, \delta z_{kh} \in X_{k,h}^{r,s}(I_j)$ the following equations hold:

$$\sum_{n=0}^{N_j-1} \left[((\partial_t u_{kh}^j, \delta z_{kh}))_n + a_n(u_{kh}^j)(\delta z_{kh}) + b_n(q_{kh}^j)(\delta z_{kh}) - ((f|_{I_j^n}, \delta z_{kh}))_n \right]$$
$$+ \sum_{n=1}^{N_j-1} ([u_{kh}^j]_{n-1}, \delta z_{kh,n-1}^+) + (u_{kh}^j(\tau_j) - s^j, \delta z_{kh}(\tau_j)) = 0, \quad (43a)$$

$$\sum_{n=0}^{N_j-1} \left[\overline{J}_{n,u}^{j'}(q_{kh}^j, u_{kh}^j)(\delta u_{kh}) - ((\partial_t z_{kh}^j, \delta u_{kh}))_n + a'_{n,u}(u_{kh}^j)(\delta u_{kh}, z_{kh}^j) \right]$$
$$- \sum_{n=1}^{N_j-1} ([z_{kh}^j]_n, \delta u_{kh,n}^-) + (z_{kh}^j(\tau_{j+1}) - \lambda^{j+1}, \delta u_{kh}(\tau_{j+1})) = 0. \quad (43b)$$

This system is, apart from the additional index h and the different function spaces, evidently equal to (42).

Remark 11 Note that for the fully discrete formulation, we discretized the control q. In doing so, we followed the approach suggested in [15] and let the control discretization be induced by the corresponding one for the states u and z, thus we did not describe it in detail. Alternatively, one might discretize q explicitly; this allows for a coarser resolution of the control in both space and time, either by choosing a coarser mesh (here, one could use hierarchically structured meshes for the state and control variables) or by employing time-stepping schemes or finite element methods of lower order than those chosen for state discretization.

Remark 12 We are now in a position to compare the dimension of systems (19) for IMS and (23) for DMS. Assume I is decomposed into 10 shooting intervals

each of which is discretized by 50 time steps, and the spatial domain Ω is the unit square divided into 256 identical elements (for other configurations, see the examples in Sect. 6). In the IMS case, the solution vector $(\delta\mathbf{s}, \delta\boldsymbol{\lambda})$ of (19) comprises 10 initial values for the state and 10 for the adjoint solution, each of the size of the spatial discretization. This amounts to a system dimension of 5120. With DMS, the control enters into the system. As we resolve the control completely in the IMS case, we do not use any condensing techniques here, either, in order to keep the results comparable and the comparison fair. Thus the control is distributed in space and time and comprises a total of $10 \cdot 50 \cdot 256 = 128{,}000$ degrees of freedom. Therefore, the solution vector $(\delta\mathbf{q}, \delta\mathbf{s}, \delta\boldsymbol{\lambda})$ of (23) has dimension 133,120.

6 Numerical Results

In this section, we discuss the practical realization of the theoretical results from Sect. 3 by regarding two examples. In Sect. 6.1 we consider the case of a linear-quadratic optimal control problem. Afterwards, we introduce a nonlinear reaction term into the PDE side condition and discuss the semilinear case in Sect. 6.2. All computations have been done using the finite element software `deal.ii` and rely upon the discretization routines presented in the last section.

6.1 Linear Example

The following linear-quadratic OCP is considered on the space-time domain $\Omega \times \bar{I} = ([-1;3] \times [-1,1]) \times [0,1]$ and aims at matching a given state profile $\hat{u}_T$ at the time interval endpoint $T = 1$:

$$\min_{(q,u)} \left[\frac{1}{2} \|u(x,1) - \hat{u}_T(x)\|^2_{L^2(\Omega)} + \frac{\alpha}{2} \int_0^1 \|q(x,t)\|^2_{L^2(\Omega)}\, dt \right],$$

subject to the parameterized nonstationary Helmholtz equation

$$\begin{aligned} \partial_t u(x,t) - \Delta u(x,t) - \omega u(x,t) &= q(x,t) && \text{in } \Omega \times (0,1], \\ u(x,t) &= 0 && \text{on } \partial\Omega \times [0,1], \\ u(x,0) &= \max\left\{0, \cos\left(\frac{\pi}{2}x_1\right)\cos\left(\frac{\pi}{2}x_2\right)\right\} && \text{on } \Omega \end{aligned}$$

The prescribed profile is chosen as $\hat{u}_T(x_1,x_2) = \min\left\{0, \cos\left(\frac{\pi}{2}x_1\right)\cos\left(\frac{\pi}{2}x_2\right)\right\}$. Thus, we expect the state solution to be a cosine bump moving from the left to the right half of the spatial domain over time, thereby changing its sign.

We compute solutions of this problem for different values of the parameter ω and $\alpha = 0.01$ by means of IMS and DMS (in the variant of Sect. 3.2) We use a four times globally refined spatial mesh (512 cells) and five equidistant shooting intervals discretized each by 100 time steps. The results are shown in Table 1: from left to right, we see that only one Newton step is needed, furthermore the number of GMRES iterations, the functional value, the residual of the respective shooting system (which also yields the stopping criterion) and the computing time measured in seconds.

Both methods have been implemented as described in Sect. 3, without any additional tuning (like condensing, reduction of control spaces etc.). Since we use the same implementation for both linear and nonlinear problems, we solve the shooting system by a Newton-type method, which requires only one iteration in the current example, as can be expected for a linear problem. For increasing ω, the number of inner GMRES iterations also increases in both cases, which reflects the worsening conditioning of the respective problems; indeed, there is a value of ω where five shooting intervals are not sufficient to solve the problem. With DMS, altogether more GMRES steps are needed than with IMS, which is due to the much larger linear system. The functional values $J(q, u)$ coincide for both methods, and also the shooting residual $\|F\|$ is of comparable size. However, the DMS algorithm takes longer (by a factor of 1.5 up to 2) than IMS to solve the problem with this same accuracy. Finally, in Fig. 1 we see that after convergence of the shooting methods (here: IMS) the expected wandering and inversion of the cosine bump is reproduced.

Table 1 Comparison of IMS (left) and DMS (right) for varying ω (required: $\|F\| < 5.0e^{-5}$) in a linear framework

ω	$\#_{New}$	$\#_{GMRES}$	$J(q,u)$	$\|F\|$	$t(s)$	$\#_{New}$	$\#_{GMRES}$	$J(q,u)$	$\|F\|$	$t(s)$
0	1	52	0.0446	$1.6e^{-11}$	1497	1	110	0.0446	$1.9e^{-10}$	2507
1	1	64	0.0367	$1.8e^{-11}$	1825	1	128	0.0367	$2.2e^{-10}$	2909
2	1	76	0.0290	$2.1e^{-11}$	2149	1	156	0.0290	$2.3e^{-10}$	3531
3	1	83	0.0218	$2.4e^{-11}$	2347	1	192	0.0218	$2.3e^{-10}$	4360
4	1	130	0.0163	$2.6e^{-11}$	3601	1	248	0.0163	$2.6e^{-10}$	5586
5	1	165	0.0148	$2.9e^{-11}$	4571	1	416	0.0148	$2.8e^{-10}$	9423

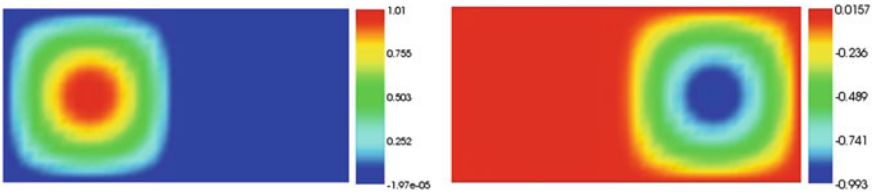

Fig. 1 Contour plot of the IMS solution on 5 shooting intervals after convergence: initial time $T = 0$ (*left*), final time $T = 1$ (*right*)

Table 2 Comparison of IMS (left) and DMS (right) for varying ω (required: $\|F\| < 1.0e^{-3}$) in a nonlinear framework

ω	$\#_{New}$	$\#_{GMRES}$	$J(q,u)$	$\|F\|$	$t(s)$	$\#_{New}$	$\#_{GMRES}$	$J(q,u)$	$\|F\|$	$t(s)$
0	4	24/51	0.1639	$3.1e^{-6}$	2530	4	28/53	0.1639	$3.1e^{-5}$	2088
1	4	26/52	0.1420	$6.4e^{-6}$	2795	4	38/62	0.1420	$1.5e^{-5}$	2427
2	4	28/56	0.1187	$2.5e^{-6}$	3118	4	43/74	0.1187	$9.0e^{-5}$	2926
3	4	28/75	0.0948	$3.9e^{-6}$	4201	4	51/84	0.0948	$1.4e^{-4}$	3280
4	4	28/79	0.0735	$5.6e^{-6}$	4713	4	68/108	0.0735	$2.1e^{-4}$	4201
5	4	28/94	0.0645	$1.2e^{-5}$	5658	4	80/139	0.0645	$2.6e^{-4}$	5376

6.2 Nonlinear Example

The second problem is a slight modification of the first one. We now choose the regularization parameter as $\alpha = 0.05$. Furthermore, we add a polynomial nonlinearity to the PDE side condition and consider the problem

$$\partial_t u(x,t) - \Delta u(x,t) - \omega u(x,t) + u(x,t)^3 = q(x,t) \quad \text{in } \Omega \times (0,1],$$

whereas the initial condition, the boundary values, the objective functional and the computational domain are chosen identical to the configuration in Sect. 6.1.

We need several Newton iterations before convergence, but at the beginning, the shooting variables are still far away from their true values. It is thus not necessary to carry out the first Newton steps on a highly refined spatial mesh. On the contrary, the shooting process becomes far more efficient if the first iterations are carried out on a coarse mesh until a good approximation has been obtained and if the spatial mesh is only then refined. In the above example, we therefore start on a mesh of only 8 cells and alternate between computing a Newton update for the shooting variables and refining the spatial mesh. This is repeated until we reach the finest mesh with 512 cells. Table 2 shows the results of this approach with global mesh refinement; in the second column of the respective method, we have given the minimum and maximum number of GMRES iterations needed within one Newton step. We emphasize that this process carries the potential of including adaptive mesh refinement into the shooting process, which is a starting point for further research. Furthermore, the IMS and DMS methods are comparable w.r.t. computing time, which is contrary to the linear example above. An extended comparison of IMS and DMS has to include additional control or state constraints, which is currently being examined.

7 Conclusion and Outlook

To underline the algebraic equivalence of DMS and IMS on an abstract function space level, we have rigorously derived the two approaches starting from a common formulation. Furthermore, we have given a detailed description of their algorithmic

realization. We have shown that our DMS approach is equivalent to the 'classical' approach known in ODE context, which leads, in contrast to our approach, to a reduced formulation. The advantage of keeping the derivation of a DMS method at the continuous level, in contrast to the 'first discretize then optimize' classical approach, is the possibility to derive an a posteriori error estimation for the space and time discretization, including the continuity conditions in the shooting system. This important aspect is part of our current research.

References

1. Albersmeyer, J.: Adjoint-based algorithms and numerical methods for sensitivity generation and optimization of large scale dynamic systems. Ph.D. thesis, Heidelberg University (2010)
2. Becker, R., Meidner, D., Vexler, B.: Efficient numerical solution of parabolic optimization problems by finite element methods. Optim. Methods Softw. **22**, 813–833 (2007)
3. Bock, H.G.: Numerical treatment of inverse problems in chemical reaction kinetics. In: Modelling of Chemical Reaction Systems. Proceedings of an International Workshop in Heidelberg, pp. 102–125. Springer, Berlin (1981)
4. Bock, H.G.: Recent advances in parameter identification problems for ODE. In: Numerical Treatment of Inverse Problems in Differential and Integral Equations, pp. 95–121. Birkhäuser, Boston (1983)
5. Bock, H.G., Plitt, K.: A multiple shooting algorithm for direct solution of optimal control problems. In: Proceedings of the 9th IFAC World Congress Budapest. Pergamon Press (1984)
6. Bock, H.G., Kostina, E., Schlöder, J.P.: Direct multiple shooting and generalized Gauss-Newton method for parameter estimation problems in ODE models. In: Carraro, T., Geiger, M., Körkel, S., Rannacher, R. (eds.) Multiple Shooting and Time Domain Decomposition Methods. Contributions in Mathematical and Computational Sciences. Springer, Berlin/Heidelberg (2015)
7. Braess, D.: Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics, 3rd edn. Cambridge University Press, Cambridge (2007)
8. Bulirsch, R., Stoer, J.: Introduction to Numerical Analysis. Texts in Applied Mathematics, vol. 12, 3rd edn. Springer, New York (2002)
9. Carraro, T., Geiger, M., Rannacher, R.: Indirect multiple shooting for nonlinear parabolic optimal control problems with control constraints. SIAM J. Sci. Comput. **36**, A452–A481 (2014)
10. Dautray, R., Lions, J.-L.: Mathematical Analysis and Numerical Methods for Science and Technology. Evolution Problems I, vol. 5. Springer, Berlin (1992)
11. Gander, M., Vandewalle, S.: Analysis of the parareal time-parallel time-integration method. SIAM J. Sci. Comput. **29**, 556–578 (2007)
12. Geiger, M.E.: Adaptive multiple shooting for boundary value problems and constrained parabolic optimization problems. Ph.D. thesis, Heidelberg University (2015)
13. Heinkenschloss, M.: A time-domain decomposition iterative method for the solution of distributed linear quadratic optimal control problems. J. Comput. Appl. Math. **173**, 169–198 (2005)
14. Hesse, H.K., Kanschat, G.: Mesh adaptive multiple shooting for partial differential equations. Part I: linear quadratic optimal control problems. J. Numer. Math. **17**, 195–217 (2009)
15. Hinze, M.: A variational discretization concept in control constrained optimization: the linear-quadratic case. Comput. Optim. Appl. **30**, 45–63 (2005)
16. Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE Constraints. Mathematical Modelling, vol. 23. Springer, New York (2009)

17. Kirches, C., Bock, H.G., Schlöder, J.P., Sager, S.: Block-structured quadratic programming for the direct multiple shooting method for optimal control. Optim. Methods Softw. **26**(2), 239–257 (2011)
18. Kirches, C., Wirsching, L., Bock, H.G., Schlöder, J.P.: Efficient direct multiple shooting for nonlinear model predictive control on long horizons. J. Process Control **22**(3), 540–550 (2012)
19. Kirches, C., Bock, H.G., Schlöder, J.P., Sager, S.: Complementary condensing for the direct multiple shooting method. In: Modeling, Simulation and Optimization of Complex Processes. Springer, Berlin (2012)
20. Leineweber, D.B.: The theory of MUSCOD in a nutshell. Diploma thesis, Heidelberg University (1995)
21. Leineweber, D.B., Bauer, I., Bock, H.G., Schlöder, J.P.: An efficient multiple shooting based reduced {SQP} strategy for large-scale dynamic process optimization. Part 1: theoretical aspects. Comput. Chem. Eng. **27**(2), 157–166 (2003)
22. Lions, J.-L., Maday, Y., Turinici, G.: Resolution d'edp par un schema en temps 'parareel'. C. R. Acad. Sci. Paris Ser. I **332**, 661–668 (2001)
23. Lory, P.: Enlarging the domain of convergence for multiple shooting by the homotopy method. Numer. Math. **35**, 231–240 (1980)
24. Meidner, D.: Adaptive space-time finite element methods for optimization problems governed by nonlinear parabolic systems. Ph.D. thesis, Heidelberg University (2008)
25. Pontryagin, L.S., Boltyanski, V.G., Gamkrelidze, R.V., Miscenko, E.F.: The Mathematical Theory of Optimal Processes. Wiley, Chichester (1962)
26. Potschka, A.: A direct method for the numerical solution of optimization problems with time-periodic PDE constraints. Ph.D. thesis, Heidelberg University (2012)
27. Potschka, A.: Direct multiple shooting for parabolic PDE constrained optimization. In: Carraro, T., Geiger, M., Körkel, S., Rannacher, R. (eds.) Multiple Shooting and Time Domain Decomposition Methods. Contributions in Mathematical and Computational Sciences. Springer, Berlin/Heidelberg (2015)
28. Schäfer, A., Kuhl, P., Diehl, M., Schlöder, J., Bock, H.G.: Fast reduced multiple shooting methods for nonlinear model predictive control. Chem. Eng. Process. **46**(11), 1200 – 1214 (2007)
29. Serban, R., Li, S., Petzold, L.: Adaptive algorithms for optimal control of time-dependent partial differential-algebraic equation systems. Int. J. Numer. Methods Eng. **57**, 1457–1469 (2003)
30. Tröltzsch, F.: Optimal Control of Partial Differential Equations: Theory, Methods and Applications. American Mathematical Society, Providence (2010)
31. Ulbrich, S.: Generalized SQP methods with 'parareal' time-domain decomposition for time-dependent PDE-constrained optimization. In: Real-time PDE-Constrained Optimization. Computational Science and Engineering, vol. 3, pp. 145–168. SIAM, Philadelphia (2007)
32. Wloka, J.: Partial Differential Equations. Cambridge University Press, Cambridge (1987)