Journal of Computational and Applied Mathematics 235 (2011) 4368–4388

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A multigrid method for constrained optimal control problems

M. Engel *, M. Griebel

Institute for Numerical Simulation, University of Bonn, Wegelerstr. 6, 53115 Bonn, Germany

ARTICLE INFO

Article history:
Received 17 July 2008
Received in revised form 1 April 2011

MSC:
49K20
65N55

Keywords:
Multigrid methods
Saddle point system
PDE-constrained optimization
Control constraints
Primal–dual active set methods

ABSTRACT

We consider the fast and efficient numerical solution of linear–quadratic optimal control problems with additional constraints on the control. Discretization of the first-order conditions leads to an indefinite linear system of saddle point type with additional complementarity conditions due to the control constraints. The complementarity conditions are treated by a primal–dual active set strategy that serves as outer iteration. At each iteration step, a KKT system has to be solved. Here, we develop a multigrid method for its fast solution. To this end, we use a smoother which is based on an inexact constraint preconditioner.

We present numerical results which show that the proposed multigrid method possesses convergence rates of the same order as for the underlying (elliptic) PDE problem. Furthermore, when combined with a nested iteration, the solver is of optimal complexity and achieves the solution of the optimization problem at only a small multiple of the cost for the PDE solution.

1. Introduction

The efficient numerical solution of optimal control problems is an important task in a variety of applications such as control of fluid flow or combustion processes. In this paper, we are concerned with the fast solution of linear–quadratic optimal control problems with additional constraints on the control. Discretization of the corresponding first-order conditions yields a large linear indefinite saddle point system and additional complementarity conditions due to the control constraints. Efficient methods to solve such problems include the primal–dual active set strategy [1–3] and interior-point methods [4,5]. In both methods a large indefinite linear system – the KKT system – has to be treated in each iteration step.

Solution methods for such saddle point problems can be classified into two broad categories: segregated and coupled approaches; see the overview article [6] and the references cited therein. The two most notable examples of the segregated approach are the Schur complement reduction and null space methods. In the Schur complement reduction method (which is also called the range space method in the optimization context), the saddle point system is reduced to a lower-dimensional system for the adjoint variables. This approach finds wide-spread use in computational methods for fluid dynamics. In the null space method, which is more commonly used in optimization problems, the system is reduced to a lower-dimensional system for the control unknowns alone by projecting onto the null space of the constraints.

In the full space or coupled approach, by contrast, one solves simultaneously for all unknowns. This can be done by a direct solver (which is prohibitive for PDE-constrained optimization problems due to the problem size) or by some iterative method, e.g. of Krylov type. To this end, efficient preconditioning is mandatory in order to achieve acceptable convergence rates. In many practical cases such preconditioners in turn build on a segregated approach. This can be seen as preconditioning a full space method with a reduced space method [7], or as accelerating a reduced space method by Krylov iterations on the full space.

* Corresponding author.
E-mail addresses: engel@ins.uni-bonn.de (M. Engel), griebel@ins.uni-bonn.de (M. Griebel).

0377-0427/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2011.04.002

Preconditioners which are based on the multigrid method have the potential to result in a fast and efficient solver with optimal complexity, at least for elliptic PDEs. In optimization, most applications of multigrid methods for saddle point systems have been in the context of reduced space methods up until now. Here, multigrid can be used to speed up the solution of the state and adjoint equations which are required at each iteration in the reduced space. The earliest work in this direction is [8], where a multigrid method for integral equations of the second kind has been employed to solve elliptic optimal control problems.

For the full space approach, the use of the multigrid methodology is so far very limited. In [9], a null space iteration was used as smoother within a multigrid method for unconstrained problems, in [10] a projected Gauss–Seidel smoothing was used within a Full Approximation Storage (FAS) multigrid method. Moreover, the work in [10] is the only full space method where additional control constraints were incorporated into a multigrid solution method up until now.

For completeness, we mention the algebraic multigrid approach introduced in [11]. There, a collective point Gauss–Seidel smoother is applied within an AMG method and thus enables to treat general coefficients within the diffusion equation. We remark, however, that in contrast to our approach, the AMG method is applied to a partially reduced optimality system in the sense that the control is eliminated and only the state and the adjoint remain as unknowns. Furthermore, the AMG-based approach has not been extended to include inequality constraints on the control.

In this article, we employ a primal–dual active set method (PDAS) for the numerical solution of control-constrained optimal control problems. The PDAS generates a sequence of large indefinite linear systems. We construct a multigrid solver for the fast and efficient solution of these KKT systems. Here, we pursue the full space approach and propose a smoothing method that is based on an inexact constraint preconditioner. The construction of the smoothing iteration is motivated by the favorable properties of constraint preconditioners [12]; however, at the same time we alleviate their biggest drawback, namely the high computational cost, by introducing an approximative version and a multigrid hierarchy. Altogether, we obtain a nested inner–outer iterative method which exhibits mesh-independent convergence. Furthermore, the outer iteration converges at a superlinear rate and the inner systems of each outer iteration step are solved with optimal complexity. These results are numerically demonstrated on a variety of model problems commonly found in the literature.

The difference to previous methods, like e.g. the approach in [10] is as follows: while there a projected Gauss–Seidel smoothing was used within a Full Approximation Storage, we employ a block smoother in an SQP/PDAS method. This gives at least an alternative to previous solvers which might be fruitful in certain applications involving the primal–dual active set approach.

The remainder of this paper is organized as follows: in Section 2, we formulate the optimal control problem and state the optimality system and the complementarity conditions. In Section 3, we introduce the discretization for the optimality system. In Section 4, we briefly introduce the primal–dual active set strategy and derive the system to be solved at each outer iteration. In Section 5, we develop an iterative method based on inexact constraint preconditioning. We then derive a multigrid method where we use this iteration as smoothing method. In Section 7, we present numerical results which demonstrate the optimal complexity of the solver for unconstrained problems. Furthermore, we show that control-constrained problems are solved efficiently as well. Finally, we draw some conclusions and give an outlook on future work.

2. Problem formulation

We consider an objective functional of tracking type given by

$$J(y,u) = \frac{1}{2}\|y-\bar{y}\|^2_{L^2(\Omega)} + \frac{\sigma}{2}\|u\|^2_{L^2(\Omega)}, \tag{1}$$

with a given target state $\bar{y} \in L^2(\Omega)$ and a regularization parameter $\sigma > 0$. The unknowns y denote the state variables and u denote the control unknowns. We assume the domain $\Omega \subset \mathbb{R}^d$, $d = 2$, to be a bounded Lipschitz domain. The controls u and the states y are coupled via the state equation

$$\begin{aligned} Ly &= f + u \quad \text{in } \Omega \\ y &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{2}$$

where L is an elliptic operator in divergence form given by $Ly = -\sum_{i,j=1}^{d} D_i(a_{ij}D_j y)$ with $a_{ij} \in L^\infty(\Omega)$, $a_{ij} = a_{ji}$ and ellipticity constant $c > 0$. The right-hand side f is a given function in $L^2(\Omega)$.

The optimization problem defined by (1) and (2) then reads

$$\begin{aligned} &\text{minimize } J(y,u) \\ &\text{subject to } Ly = f + u \quad \text{in } \Omega,\ y = 0 \text{ on } \partial\Omega \\ &\quad \text{and} \quad u \in U_{\text{ad}}, \end{aligned} \tag{OP}$$

where U_{ad} denotes a set of admissible controls which is a proper convex and closed subset of $U = L^2(\Omega)$. Here, we consider the box constraints given by

$$U_{\text{ad}} = \{v \in L^2(\Omega) \mid \xi_l \le v \le \xi_u \text{ a.e. in } \Omega\} \tag{3}$$

where $\xi_l, \xi_u \in L^\infty(\Omega) \cap H^1(\Omega)$ denote the lower and upper bounding functions for the controls.

It is well known that the optimization problem (OP) has a unique solution (y^*, u^*); cf. [13,14]. Furthermore, there exists the adjoint variable p^* such that the triple (y^*, u^*, p^*) satisfies the following first-order optimality conditions:

$$\begin{aligned} &Ly^* = f + u^* \quad \text{in } \Omega \\ &y^* = 0 \quad \text{on } \partial\Omega \\ &L'p^* = \bar{y} - y^* \quad \text{in } \Omega \\ &p^* = 0 \quad \text{on } \partial\Omega \\ &(\sigma u^* - p^*, v - u^*) \geq 0 \quad v \in U_{\text{ad}}. \end{aligned} \tag{OS}$$

In the case without constraints on the control, i.e. $U_{\text{ad}} = U$, the last inequality reduces to the optimality condition

$$\sigma u^* - p^* = 0. \tag{4}$$

The optimal state and associated adjoint satisfy $y^* \in H_0^1(\Omega) \cap H^2(\Omega)$ and $p^* \in H_0^1(\Omega) \cap H^2(\Omega)$, respectively. For the optimal control one obtains $u^* \in H^1(\Omega)$ for the control-constrained case and $u^* \in H^2(\Omega)$ for the unconstrained case. In our case, due to the linear equality constraints, the quadratic functional J and the convexity of U_{ad}, the optimization problem is convex and therefore the necessary first-order condition (OS) is also sufficient for a solution of the optimization problem (OP).

3. Discretization

We will now introduce the discrete optimality system. To this end, let $\mathcal{T}_h$ be a shape-regular quasi-uniform partition of the domain Ω into convex quadrilaterals T_i, $i = 1, \ldots, N$. This mesh will serve for the discretization of the controls as well as for the state and adjoint unknowns.

We discretize the control function u using piecewise constant functions. We denote the space of piecewise constant functions on the mesh $\mathcal{T}_h$ by $U_h \subset U$. Let Π_h be the local L^2-projection operator onto U_h, i.e.

$$\Pi_h u(x) = \frac{1}{|T_i|} \int_{T_i} u(y)\mathrm{d}y, \quad x \in T_i \in \mathcal{T}_h. \tag{5}$$

For Π_h and $u \in H^1(\Omega)$, the well-known estimate

$$\|u - \Pi u\|_{L^2(\Omega)} \leq Ch\|u\|_{H^1(\Omega)} \tag{6}$$

holds [15]. Clearly, in general we cannot expect a better approximation order for the controls than $\mathcal{O}(h)$ in the L^2-norm if piecewise constants are used.

By discretizing the bounding functions ξ_l and ξ_u with piecewise constants we obtain a discrete approximation to the set of admissible controls U_{ad},

$$U_{\text{ad},h} = \{v_h \in U_h \mid \Pi_h \xi_l \leq v_h \leq \Pi_h \xi_u\}. \tag{7}$$

For the approximation error of the solution u_h^* to the "semi-discrete" problem

$$\min_{u \in U_{\text{ad},h}} J(y, u) \tag{8}$$

the optimal order result $\|u^* - u_h^*\|_{L^2(\Omega)} = \mathcal{O}(h)$ has been shown in [16]. Furthermore, for triangular elements, h^2-superconvergence in the midpoints holds; cf. [17]. Recently, two approaches have been presented that achieve h^2-convergence also in a continuous norm. The first approach [18] works by a semi-discretization, where a discrete approximation to the control is obtained by projecting the adjoint. The second approach [17] uses a post-processing step to improve the convergence order from h to h^2.

It remains to discretize the state and adjoint equations in the optimality system (OS). To this end, we apply a mixed finite element discretization using the lowest-order Raviart–Thomas approximation spaces, i.e. RT_0-elements [19]. Employing midpoint and trapezoidal quadrature, one obtains a positive definite system for the scalar unknowns; see e.g. [20]. In two dimensions, we then obtain a nine-point stencil and a diagonal mass matrix. All in all, this discretization of the PDE constraints (2) yields the linear system

$$L_h y_h = M_h f_h + M_h u_h \tag{9}$$

with $L_h, M_h \in R^{N \times N}$, where N denotes the number of elements in $\mathcal{T}_h$, and $f_h = \Pi_h f$. Analogously, for the discretization of the adjoint equation in (OS) we obtain the system

$$L_h^T p_h = M_h \bar{y}_h - M_h y_h, \tag{10}$$

with $\bar{y}_h = \Pi_h \bar{y}$.

For the employed discretization scheme, optimal order L^2-convergence holds, i.e. we obtain a convergence rate of the order $\mathcal{O}(h)$ for the error. Furthermore, under the smoothness assumption $C^{3,1}$ for the scalar unknown, superconvergence of the order $\mathcal{O}(h^2)$ at the midpoints of the elements can be shown for the error measured in a discrete L^2-norm with midpoint rule integral evaluation. We again refer to [20] for details.

The L^2-inner product appearing in the third equation of (OS) is discretized again employing midpoint quadrature. Thus, the same mass matrix M_h as in (9), (10) results. We then obtain

$$(\sigma M_h u_h - M p_h)^T (v_h - u_h) \geq 0, \quad v_h \in U_{\mathrm{ad},h} \tag{11}$$

as discrete optimality condition.

Let us briefly consider the case without any constraints on the control, i.e. (OS) with (4). Then, the discretization of (OS) results in the linear system

$$K_h x_h = b_h \tag{12}$$

where K_h is the saddle point or KKT matrix

$$K_h = \begin{pmatrix} M_h & 0 & L_h^T \\ 0 & \sigma M_h & -M_h \\ L_h & -M_h & 0 \end{pmatrix}, \tag{13}$$

$x_h = (y_h, u_h, p_h)$ is the vector of unknowns and the right-hand side is given by $b_h = (M_h \bar{y}_h, 0, M_h f_h)$.

If the constraints have full row rank and the Hessian block is positive definite on the null space of the constraints then it is well known that K is regular. Both conditions are obviously satisfied in our case. Furthermore, K is indefinite and its inertia is given by $\mathrm{inertia}(K) = (2N, N, 0)$. Note that systems of this type also have to be solved at each step of the primal–dual active set strategy. This will be discussed in the following.

4. The primal–dual active set method

In this section we will briefly describe the primal–dual active set strategy that will be used as outer iteration to handle the control constraints [2,1,3]. The main advantage of treating the constraints in an outer iteration is as follows: the resulting inner subsystem and in particular the smoother of a multigrid method applied there does not need to take the constraints into account. Thus, the inner systems to be solved are strictly linear. Furthermore, compared to interior-point methods, the PDAS approach is more efficient for control-constrained problems; cf. [2].

Let us now derive the PDAS method in detail. To this end, note that for the variational inequality (11), an equivalent formulation is given by

$$\begin{aligned} &h\sigma M_h u_h - M_h p_h + \lambda = 0, \\ &\lambda = \max(\lambda + c(u_h - \Pi_h \xi_u), 0) + \min(\lambda + c(u_h - \Pi_h \xi_l), 0), \quad c > 0. \end{aligned} \tag{14}$$

Here, the unknowns λ are the Lagrange multipliers associated with the inequality constraints. They satisfy the Karush–Kuhn–Tucker conditions

$$\begin{aligned} &\lambda \leq 0 \quad \text{on } \mathcal{A}_-^* = \{x \mid u_h^* = \Pi_h \xi_l\}, \\ &\lambda \geq 0 \quad \text{on } \mathcal{A}_+^* = \{x \mid u_h^* = \Pi_h \xi_u\}, \\ &\lambda = 0 \quad \text{on } \mathcal{I}^* = \{x \mid \Pi_h \xi_l < u_h^* < \Pi_h \xi_u\}. \end{aligned} \tag{15}$$

Here, $\mathcal{A}_-^*$ and $\mathcal{A}_+^*$ are the active sets and $\mathcal{I}^*$ is the inactive set at the (discrete) optimal solution u_h^*.

The primal–dual active set strategy is an iterative algorithm that makes use of (14) to predict the active and inactive sets and treats an associated equality-constrained optimization problem at each step. This leads to the following algorithm:

1: Choose initial values y_h^0, u_h^0, p_h^0, λ^0 and set $k = 1$
2: **while** not converged **do**
3: predict $\mathcal{A}_-^k$, $\mathcal{A}_+^k$, $\mathcal{I}^k$ as follows:

$$\mathcal{A}_-^k = \left\{ i \mid u_h^{k-1} + \frac{\lambda^{k-1}}{\sigma} < \Pi_h \xi_l \text{ on } T_i \right\} \tag{16}$$

$$\mathcal{A}_+^k = \left\{ i \mid u_h^{k-1} + \frac{\lambda^{k-1}}{\sigma} > \Pi_h \xi_u \text{ on } T_i \right\} \tag{17}$$

$$\mathcal{I}^k = \{ i \mid i \notin \mathcal{A}_-^k \cup \mathcal{A}_+^k \} \tag{18}$$

4: **if** $k \geq 2$ and $\mathcal{A}_-^k = \mathcal{A}_-^{k-1}$, $\mathcal{A}_+^k = \mathcal{A}_+^{k-1}$, $\mathcal{I}^k = \mathcal{I}^{k-1}$ **then**
5: converged = true
6: **else**

7: solve the equality-constrained problem

$$\begin{aligned} &M_h y_h^k + L_h^T p_h^k = M_h \bar{y}_h \\ &\sigma M_h u_h^k - M_h^T p_h^k + \lambda^k = 0 \\ &L_h y_h^k - M_h u_h^k = M_h f_h \\ &\lambda^k = 0 \quad \text{on } \mathcal{I}^k \\ &u_h^k = \Pi_h \xi_l \quad \text{on } \mathcal{A}_-^k \\ &u_h^k = \Pi_h \xi_u \quad \text{on } \mathcal{A}_+^k \end{aligned} \tag{EQP}$$

8: **end if**

9: $k = k + 1$

10: **end while**.

This concludes the description of the PDAS method. For details and convergence properties, we refer to [1]. Note that for the case without control constraints, we have $\mathcal{A}_- = \mathcal{A}_+ = \emptyset$ and the overall algorithm reduces to just the solution of (EQP) which in turn reduces to the saddle point system (12).

The main computational effort in this algorithm has to be spent for the solution of (EQP). In many publications this is done by eliminating y_h^k, p_h^k and solving the reduced system for the control unknowns u_h^k by a conjugate gradient method. Here, we want to avoid the high cost for the full solution of the constraints in each iteration step and aim at a full space multigrid method instead. To this end, we modify the system (EQP) in such a way that it can be formulated as a KKT system (12).

We proceed as follows: first, we partition the control unknowns according to $u_h^k = [u_h^{\mathcal{I}^k}\ u_h^{\mathcal{A}_-^k}\ u_h^{\mathcal{A}_+^k}]$. The same partitioning applies to the Lagrange multipliers $\lambda^k = [\lambda^{\mathcal{I}^k}\ \lambda^{\mathcal{A}_-^k}\ \lambda^{\mathcal{A}_+^k}]$. Note that this partitioning induces corresponding 3×3 block, 3×1 column and 1×3 row block partitions of the mass matrix M_h. Then, the system given by the first three lines in (EQP) can be written as

$$\left[\begin{array}{c|ccc|c} M_h & & & & L_h^T \\ \hline & \sigma M_h^{\mathcal{I}^k,\mathcal{I}^k} & \sigma M_h^{\mathcal{I}^k,\mathcal{A}_-^k} & \sigma M_h^{\mathcal{I}^k,\mathcal{A}_+^k} & -M_h^{\mathcal{I}^k,*} \\ & \sigma M_h^{\mathcal{A}_-^k,\mathcal{I}^k} & \sigma M_h^{\mathcal{A}_-^k,\mathcal{A}_-^k} & \sigma M_h^{\mathcal{A}_-^k,\mathcal{A}_+^k} & -M_h^{\mathcal{A}_-^k,*} \\ & \sigma M_h^{\mathcal{A}_+^k,\mathcal{I}^k} & \sigma M_h^{\mathcal{A}_+^k,\mathcal{A}_-^k} & \sigma M_h^{\mathcal{A}_+^k,\mathcal{A}_+^k} & -M_h^{\mathcal{A}_+^k,*} \\ \hline L_h & -M_h^{*,\mathcal{I}^k} & -M_h^{*,\mathcal{A}_-^k} & -M_h^{*,\mathcal{A}_+^k} & \end{array}\right] \left[\begin{array}{c} y_h^k \\ \hline u_h^{\mathcal{I}^k} \\ u_h^{\mathcal{A}_-^k} \\ u_h^{\mathcal{A}_+^k} \\ \hline p_h^k \end{array}\right] = \left[\begin{array}{c} M_h \bar{y}_h \\ \hline -\lambda^{\mathcal{I}^k} \\ -\lambda^{\mathcal{A}_-^k} \\ -\lambda^{\mathcal{A}_+^k} \\ \hline M_h f_h \end{array}\right]. \tag{19}$$

Now we utilize the last three equations in (EQP) to reduce (19) to a system for y_h^k, $u_h^{\mathcal{I}_k}$, p_h^k, i.e. we eliminate $u_h^{\mathcal{A}_-^k}$, $u_h^{\mathcal{A}_+^k}$ and we consider the controls u_h^k only on the inactive set $\mathcal{I}^k$. The solution of (EQP) then proceeds in two steps: first, the saddle point system

$$K_h^{\mathcal{I}^k} x_h^{\mathcal{I}^k} = r_h^{\mathcal{I}^k} \tag{20}$$

has to be solved, where

$$K_h^{\mathcal{I}^k} = \begin{bmatrix} M_h & & L_h^T \\ & \sigma M_h^{\mathcal{I}^k,\mathcal{I}^k} & -M_h^{\mathcal{I}^k,*} \\ L_h & -M_h^{*,\mathcal{I}^k} & \end{bmatrix}, \qquad r_h^{\mathcal{I}^k} = \begin{bmatrix} M_h \bar{y}_h \\ -\sigma M_h^{\mathcal{I}^k,\mathcal{A}_-^k} \Pi_h \xi_l - \sigma M_h^{\mathcal{I}^k,\mathcal{A}_+^k} \Pi_h \xi_u \\ M_h f_h + M_h^{*,\mathcal{A}_-^k} \Pi_h \xi_l + M_h^{*,\mathcal{A}_+^k} \Pi_h \xi_u \end{bmatrix}, \tag{21}$$

and the vector of unknowns is given by $x_h^{\mathcal{I}^k} = [y_h^k\ u_h^{\mathcal{I}^k}\ p_h^k]$. Note that the KKT operator $K_h^{\mathcal{I}^k}$ and the right-hand side vector $r_h^{\mathcal{I}^k}$ depend on the index k of the outer iteration. In the second step, the Lagrange multipliers λ^k are computed by

$$\begin{aligned} \lambda^{\mathcal{A}_-^k} &= M_h^{\mathcal{A}_-^k,*} p_h^k - \sigma M_h^{\mathcal{A}_-^k,\mathcal{I}^k} u_h^{\mathcal{I}^k} - \sigma M_h^{\mathcal{A}_-^k,\mathcal{A}_-^k} \Pi_h \xi_l - \sigma M_h^{\mathcal{A}_-^k,\mathcal{A}_+^k} \Pi_h \xi_u, \\ \lambda^{\mathcal{A}_+^k} &= M_h^{\mathcal{A}_+^k,*} p_h^k - \sigma M_h^{\mathcal{A}_+^k,\mathcal{I}^k} u_h^{\mathcal{I}^k} - \sigma M_h^{\mathcal{A}_+^k,\mathcal{A}_-^k} \Pi_h \xi_l - \sigma M_h^{\mathcal{A}_+^k,\mathcal{A}_+^k} \Pi_h \xi_u, \end{aligned} \tag{22}$$

compare lines 3 and 4 in (19). On the inactive set, we just set $\lambda^{\mathcal{I}^k} = 0$. Note again that, for the case without control constraints, system (20) reduces to (12), i.e. we just have $K_h^{\mathcal{I}^k} = K_h$, $r_h^{\mathcal{I}^k} = b_h$ and $x_h^{\mathcal{I}^k} = x_h$.

It remains to obtain the solution of (20) in a fast and efficient fashion. To this end, in the following section we will first derive a stationary iterative method that later serves as smoother in a multigrid approach.

5. A stationary iterative method for KKT systems

Now we discuss an iterative method for the solution of the KKT system (20) which arises at each iteration step of the PDAS method. A stationary iterative method for (20) can be written as a preconditioned Richardson method, i.e. we have, using a damping factor of 1,

$$x_h^{\mathcal{I},i+1} = x_h^{\mathcal{I},i} + (C_h^{\mathcal{I}})^{-1}(r_h^{\mathcal{I}} - K_h^{\mathcal{I}} x_h^{\mathcal{I},i}). \tag{23}$$

Here and in the following we omit the parameter k of the outer PDAS iteration step for reasons of simplicity. We define the preconditioner $C_h^{\mathcal{I}}$ in (23) as the block triangular matrix given by

$$C_h^{\mathcal{I}} = \begin{pmatrix} & \hat{H}_Z^{\mathcal{I}} & \hat{L}_h^T \\ \hat{L}_h & -M_h^{*,\mathcal{I}} & -M_h^{\mathcal{I},*} \end{pmatrix}. \tag{24}$$

The blocks $\hat{L}_h$ and $\hat{L}_h^T$ are suitable approximations to the discretized differential operators L_h and L_h^T in the state equation (9) and the adjoint equation (10), respectively. We stress the fact that these approximations are not constructed explicitly but are rather given implicitly by definition of the action of $\hat{L}_h^{-1}$ and $\hat{L}_h^{-T}$ on a given vector by means of a few steps of an iterative method. The precise approximations will be made clear later. The blocks $M_h^{*,\mathcal{I}}$ and $M_h^{\mathcal{I},*}$ associated to corresponding parts of the mass matrix are carried over unchanged from the KKT operator (21). The matrix $\hat{H}_Z^{\mathcal{I}}$ is given by

$$\hat{H}_Z^{\mathcal{I}} = M_h^{\mathcal{I},*}\hat{L}_h^{-T}M_h\hat{L}_h^{-1}M_h^{*,\mathcal{I}} + \sigma M_h^{\mathcal{I},\mathcal{I}}. \tag{25}$$

We point out that the matrix $\hat{H}_Z^{\mathcal{I}}$ is never formed explicitly but rather is defined through its matrix–vector product, again making use of the approximations $\hat{L}_h$ and $\hat{L}_h^T$.

Due to the block triangular form of $C_h^{\mathcal{I}}$, the computation of $w_h^{\mathcal{I}} = (C_h^{\mathcal{I}})^{-1}v_h^{\mathcal{I}}$, with $w_h^{\mathcal{I}} = (w_h^y, w_h^{u^{\mathcal{I}}}, w_h^p)$ and $v_h^{\mathcal{I}} = (v_h^y, v_h^{u^{\mathcal{I}}}, v_h^p)$ is achieved by performing the block back substitution given by the algorithm

1: $w_h^p \leftarrow \hat{L}_h^T w_h^p = v_h^y$
2: $w_h^{u^{\mathcal{I}}} \leftarrow \hat{H}_Z^{\mathcal{I}} w_h^{u^{\mathcal{I}}} = v_h^{u^{\mathcal{I}}} + M_h^{\mathcal{I},*} w_h^p$
3: $w_h^y \leftarrow \hat{L}_h w_h^y = v_h^p + M_h^{*,\mathcal{I}} w_h^{u^{\mathcal{I}}}$.

Here, in lines 1 and 3, solutions of linear systems with the approximations $\hat{L}_h$ and $\hat{L}_h^T$ are required. These solutions are computed approximately by a fixed small number of steps (in many cases, just one) of two stationary iterative methods

$$w_h^{y,i+1} = w_h^{y,i} + G_h^{-1}(v_h^p + M_h^{*,\mathcal{I}} w_h^{u^{\mathcal{I}}} - L_h w_h^{y,i}) \quad \text{and} \quad w_h^{p,i+1} = w_h^{p,i} + G_h^{-T}(v_h^y - L_h^T w_h^{p,i}), \tag{26}$$

respectively. The matrix G_h results from an appropriate splitting of the discrete differential operator L_h. For L_h being the discrete Laplace operator, we use the Gauss–Seidel iteration, i.e. $G_h = D_h - E_h$, where D_h is the diagonal and E_h the lower triangular part of L_h. For a more difficult L_h, different schemes such as incomplete LU factorization of L_h or alternating line Gauss–Seidel may be used. In general, any appropriate smoothing iteration for a robust multigrid solver for $L_h w_h^y = v_h^p + M_h^{*,\mathcal{I}} w_h^{u^{\mathcal{I}}}$ and $L_h^T w_h^p = v_h^y$ is a good candidate for (26) and its corresponding G_h here.

In line 2, formally the inversion of the operator $\hat{H}_Z^{\mathcal{I}}$ is required. As mentioned before, $\hat{H}_Z^{\mathcal{I}}$ is never formed explicitly and thus we apply a Krylov method, where only the matrix–vector product with $\hat{H}_Z^{\mathcal{I}}$ is needed. This again involves multiplications with $\hat{L}_h^{-1}$ and $\hat{L}_h^{-T}$, i.e. the application of a few steps of the iterative methods (26). It is clear that the resulting operator is symmetric. Furthermore, for exact inner solves, the operator would be positive definite.[1] For these reasons, we employ the conjugate gradient method with the matrix–vector product given by (25). Again, we do not solve to a tight absolute tolerance, but rather perform only a small fixed number of cg iterations, here in the extremal case this even might be only one.

All in all, we obtain an iterative method that we write as

$$x_h^{\mathcal{I},i+\nu} = (\mathcal{S}_{h,\alpha,\beta}^{\mathcal{I}})^{\nu}(x_h^{\mathcal{I},i}, r_h^{\mathcal{I}}), \tag{27}$$

where the operation of $\mathcal{S}_{h,\alpha,\beta}^{\mathcal{I}}$ is given by (23), ν is the number of applications of $\mathcal{S}_{h,\alpha,\beta}^{\mathcal{I}}$, α denotes the number of iterations for the inner solves with $\hat{L}_h$ and $\hat{L}_h^T$ and β denotes the number of conjugate gradient steps employed to obtain an approximate solution to the system on line 2, i.e. to approximate the action of $(\hat{H}_Z^{\mathcal{I}})^{-1}$.

At this point, let us comment on the relation of (24) to the so-called constraint preconditioner; see [6]. Here, a specific preconditioner corresponding to (24) is given by

$$B_h^{\mathcal{I}} = \begin{pmatrix} & \tilde{H}_Z^{\mathcal{I}} & L_h^T \\ L_h & -M_h^{*,\mathcal{I}} & -M_h^{\mathcal{I},*} \end{pmatrix}, \tag{28}$$

where the matrix $\tilde{H}_Z^{\mathcal{I}}$ is some approximation to the reduced Hessian restricted to the inactive set

$$H_Z^{\mathcal{I}} = M_h^{\mathcal{I},*}L_h^{-T}M_hL_h^{-1}M_h^{*,\mathcal{I}} + \sigma M_h^{\mathcal{I},\mathcal{I}}. \tag{29}$$

[1] In the case of inexact inner solves by just one or a few iterations steps, positive definiteness is not guaranteed for arbitrary values of σ and general choices of G_h. However, in our numerical experiments, it turned out that the cg method always converged.

The expression *constraint preconditioner* is apparent from the structure of B_h: the blocks associated with the constraints are adopted without modification from the KKT matrix $K_h^{\mathcal{I}}$ of (21). Now let us denote the upper left 2×2 block of the KKT matrix (21), which corresponds to the Hessian of the Lagrangian, by H, i.e.

$$H = \begin{bmatrix} M_h & \\ & \sigma M_h^{\mathcal{I},\mathcal{I}} \end{bmatrix}. \tag{30}$$

Let us recall that a reduced Hessian is computed by $H_Z^{\mathcal{I}} = Z^T H Z$ where the columns of the matrix Z span any basis for the null space of the constraints; see e.g. [21]. Such a null space basis Z is often computed by factorization or reordering methods;[2] however, if the constraints represent a discretized PDE, this approach is computationally prohibitive. A feasible alternative is given by the so-called fundamental basis [21], which is given in our context by

$$Z = [-L_h^{-1} M_h^{*,\mathcal{I}} \quad I_h]^T, \tag{31}$$

where I_h denotes the identity. Using (31) in $H_Z^{\mathcal{I}} = Z^T H Z$, we just obtain the expression (29) for $H_Z^{\mathcal{I}}$. As shown in [12], the following properties hold for the preconditioned matrix $(B_h^{\mathcal{I}})^{-1} K_h^{\mathcal{I}}$: first, an eigenvalue 1 arises with multiplicity $2N$, then N eigenvalues are defined by the generalized eigenvalue problem $H_Z^{\mathcal{I}} x = \mu \tilde{H}_Z^{\mathcal{I}} x$ and finally, the dimension of the Krylov subspace $\mathcal{K}((B_h^{\mathcal{I}})^{-1} K_h^{\mathcal{I}}, r_h^{\mathcal{I}})$ associated with the preconditioned matrix $(B_h^{\mathcal{I}})^{-1} K_h^{\mathcal{I}}$ and the right-hand side $r_h^{\mathcal{I}}$ is at most $N+2$. Thus, if $\tilde{H}_Z^{\mathcal{I}} = H_Z^{\mathcal{I}}$, i.e. if the true reduced Hessian is used in $B_h^{\mathcal{I}}$, a Krylov method preconditioned with $B_h^{\mathcal{I}}$ will converge in three iterations. The main limitation of this approach is the high computational cost due to the exact inversions of the discretized PDE operators L_h and L_h^T. Some of the cost could be avoided by replacing the Krylov method for the solution with $\tilde{H}_Z^{\mathcal{I}}$ by some quasi-Newton approximation like BFGS. On the other hand, BFGS approximations can yield dense matrices, in this case of dimension N, unless one resorts to limited memory variants with lower approximation quality. For these reasons, a first step in improving efficiency is to use only approximate solutions for the inner and outer linear systems, which may just result in the presented iteration (23). Note here that, for $\alpha, \beta \to \infty$, we have $C_h^{\mathcal{I}} \to B_h^{\mathcal{I}}$ with $\tilde{H}_Z^{\mathcal{I}} = H_Z^{\mathcal{I}}$. In this sense, the developed iteration could as well be interpreted as an inexact constraint preconditioner. To be precise, however, we stress the point that in contrast to (28), the operator (24) is not a constraint preconditioner, since for small α the constraints are not exactly eliminated.

Since it is well known that the Richardson method may require strong damping in order to enforce convergence and may in general only converge slowly, we advocate here a further acceleration by using a multigrid method in which (23) is merely employed as a smoother. This will be discussed in the following section.

6. A multigrid method for KKT systems

When applying an iterative method such as (27) as a smoothing iteration within a multigrid context, its smoothing property is of primary interest. Therefore, in the following we will briefly comment on the smoothing behavior of (27) and compute the so-called smoothing factor based on the local Fourier analysis for a special case of (27). For the sake of readability, we omit the technical details and refer to the Appendix and [22]; for a general introduction to the local Fourier analysis, see [23–25].

The smoothing factor obtained by the local Fourier analysis is a quantitative measure for the largest possible amplification factor of an iteration with respect to high frequency components. In the scalar case, the smoothing factor for an operator S_{L_h} is defined as

$$\mu_{\mathrm{LFA}}(S_{L_h}) = \sup \left\{ |\tilde{S}_{L_h}(\theta)| : \theta \in [-\pi, \pi)^2 \setminus \left[-\frac{\pi}{2}, \frac{\pi}{2} \right)^2 \right\}, \tag{32}$$

where θ are the frequencies considered on an infinite grid and $\tilde{S}_{L_h}(\theta)$ is the symbol or generalized eigenvalue of S_{L_h}; c.f. (58) in the Appendix. For systems, we consider a vector of frequency functions and the symbol corresponding to (24) is the matrix composed of the respective scalar symbols.

The smoothing factor will be derived for the iteration in the unconstrained case, i.e. $\mathcal{A} = \emptyset$ and thus we will omit the indices $\mathcal{I}$ on matrices and vectors. In order to compute the smoothing factor, the iteration (27) has to allow a simple splitting. To this end, we set $\alpha = 1$, $\beta = 0$, and $\hat{H}_Z = \sigma M_h$. Furthermore, we consider the linear–quadratic model problem with Laplace's equation as constraint, i.e. L_h is given by the usual five-point stencil, and as constraint smoother, we employ the damped Jacobi iteration with the optimal damping factor $\omega = 4/5$. With $h_j, j = 0, \ldots, J$ we denote the mesh sizes of the different grid levels in the multigrid hierarchy where the smoothing iteration is applied. In the following, this particular case of iteration (27) will be denoted by $\mathcal{S}_{j,1}$.

[2] This is closely related to the family of null space methods; cf. [21].

Proposition 1. *Under the above assumptions, the smoothing factor for (27) is obtained as*

$$\mu_{\mathrm{LFA}}(S_{j,1}) = \frac{3}{5} + \frac{h_j^4}{50\sigma} + \sqrt{\frac{h_j^4}{50\sigma}\left(\frac{6}{5} + \frac{h_j^4}{50\sigma}\right)}. \tag{33}$$

Note that $\mu_{\mathrm{LFA}}(S_{j,1})$, in contrast to μ_{LFA} in the scalar case, explicitly depends on the mesh size h_j. This is usually the case when deriving smoothing factors for systems, due to differential operators of different order appearing in K_j.

A smoothing factor $\mu_{\mathrm{LFA}}(S_{j,1}) > 1$ is an indication for degradation of convergence or even divergence of the multigrid process. For (33) it holds that $\mu_{\mathrm{LFA}}(S_{j,1}) = 1$ for $\sigma = h_j^4/4$ and thus we obtain the condition

$$\mu_{\mathrm{LFA}}(S_{j,1}) < 1 \quad \text{if } \sigma > \frac{h_j^4}{4}. \tag{34}$$

In Section 7 we will further elaborate on (34) with numerical experiments and a discussion of the consequences of (34) for the robustness of our multigrid solver.

Now we develop a multigrid method for the KKT system (12) and for its particular form (20). In general, any multigrid approach makes use of discretizations on successively coarser levels. Its further components are suitable interpolation operators between different grid levels, a smoothing iteration which is employed on each level, and a solution method for the equation on the coarsest grid level. For details on the multigrid approach, we refer the reader to [24–26].

In the construction of a multigrid method for the KKT system (20), several challenges arise due to the block and saddle point structures of the system. The first task is the design of a suitable smoothing iteration. To this end, we apply the preconditioned Richardson iteration (23) derived in the previous section. In most of our numerical experiments, it was sufficient to use $\nu = \alpha = \beta = 1$. Furthermore, we use the direct discretization approach, that is, we apply the cell-centered RT_0 and piecewise constant discretizations of Section 3 for the state and adjoint components and the control unknowns of the system, respectively, on a sequence of successively coarser grids. The different grid levels are here denoted by the parameter j, $j = 0, \ldots, J$ where $j = 0$ stands for the coarsest mesh. The mesh size for each grid level is given by $h_j = 2^{-j}h_c$ with h_c denoting the coarsest mesh size. For a given level j and associated mesh size h_j, we denote the associated mesh with $\mathcal{T}_j$. We recall that for the cell-centered discretization and $d = 2$, a coarse grid cell $T_i \in \mathcal{T}_{j-1}$ is the union of four fine grid cells which we sometimes denote with $T_i^\nu \in \mathcal{T}_j$, $\nu = 1, 2, 3, 4$.

From the two-step solution of (EQP) it follows that the Lagrange multipliers λ as well as the bounding functions ξ_l and ξ_u need to be discretized on the finest grid level only. However, for the discretization of $x_{h_j}^{\mathcal{I}}$, $r_{h_j}^{\mathcal{I}}$ and the operator $K_{h_j}^{\mathcal{I}}$ in (20) on a grid level $j < J$, it is evident that we have to approximate the inactive set $\mathcal{I}$ on that grid level. This has to be done in each PDAS iteration, after the inactive and active sets on the finest level J have been determined by the algorithm and before the solution of system (20). Here, the third step of the PDAS algorithm yields index sets $\mathcal{I}$, $\mathcal{A}_-$ and $\mathcal{A}_+$ on the finest level J and we denote the set of grid cells corresponding to $\mathcal{I}$ with $\mathcal{T}_{\mathcal{I}_J}$. Now, for given $\mathcal{T}_{\mathcal{I}_j}$ we define $\mathcal{T}_{\mathcal{I}_{j-1}}$ as the set of all coarser grid cells for which at least one fine grid subcell is contained in that of the next finer inactive set, i.e.

$$\mathcal{T}_{\mathcal{I}_{j-1}} = \{T_i \in \mathcal{T}_{j-1} \mid T_i^\nu \in \mathcal{T}_{\mathcal{I}_j} \text{ for } \nu \in \{1, 2, 3, 4\}\}, \quad j = J, \ldots, 1. \tag{35}$$

Note that no representation of the active sets $\mathcal{A}_-$ and $\mathcal{A}_+$ is needed on coarser levels. From the sequence of meshes given by (35) we then obtain a sequence of operators $K_j^{\mathcal{I}}$ by direct discretization, where we now use the grid level j as index in (20). In the same way, we will use the index j instead of the subscript h_j for other operators, vectors and variables.

The intergrid transfer operators are defined in a blockwise manner, e.g. restriction and prolongation are given by

$$\mathcal{R}_j^{j-1} = \begin{pmatrix} R_j^{j-1} & & \\ & R_j^{j-1,\mathcal{I}} & \\ & & R_j^{j-1} \end{pmatrix}, \qquad \mathcal{P}_{j-1}^{j} = \begin{pmatrix} P_{j-1}^{j} & & \\ & P_{j-1}^{j,\mathcal{I}} & \\ & & P_{j-1}^{j} \end{pmatrix}, \tag{36}$$

respectively. Note that the three unknown components y_j, $u_j^{\mathcal{I}}$ and p_j resulting from the discretization are all located at the cell-centers. Therefore, basically the same scalar restriction and prolongation operator can be used for all three components. To be precise, we choose here the four-point average restriction for R_j^{j-1} and the bilinear interpolation for P_{j-1}^{j}.[3] In the usual stencil notation [26,25], the four-point average restriction and the bilinear interpolation are given by

$$R_{j,FPA}^{j-1} = \frac{1}{4}\begin{bmatrix} 1 & & 1 \\ & & \\ 1 & & 1 \end{bmatrix}, \qquad P_{j-1,BL}^{j} = \frac{1}{16}\begin{bmatrix} 1 & 3 & 3 & 1 \\ 3 & 9 & 9 & 3 \\ 3 & 9 & 9 & 3 \\ 1 & 3 & 3 & 1 \end{bmatrix}. \tag{37}$$

[3] These transfer operators are consistent with the general rule $m_p + m_r > 2m$, where m_p and m_r are the order of prolongation and restriction, respectively, and $2m$ is the order of the differential operator; see [25].

Furthermore, in (36), the symbol $R_j^{j-1,\mathcal{I}}$ represents the four-point average operator giving values only for grid cells $T_i \in \mathcal{T}_{\mathcal{I}_{j-1}}$. When applying $R_j^{j-1,\mathcal{I}}$ to obtain a coarse grid value of the $u_{j-1}^{\mathcal{I}}$-component, fine grid values on the active set $\mathcal{T}_j \setminus \mathcal{T}_{\mathcal{I}_j}$ could be needed. However, on $\mathcal{A}_-$ and $\mathcal{A}_+$ the solution is fixed to the constraints u^α, u^β, respectively and the corresponding residuals vanish. Thus, active nodes should provide no contribution here and consistently the corresponding stencil entries are set to zero. Analogous considerations apply to the prolongation $P_{j-1}^{j,\mathcal{I}}$. Here, coarse grid values on the active set $\mathcal{T}_{j-1} \setminus \mathcal{T}_{\mathcal{I}_{j-1}}$ could enter the prolongation stencil when computing the correction for a fine grid value of $u_j^{\mathcal{I}}$. However, again the corrections due to active nodes should be zero and the corresponding stencil entries are set to zero.

It remains to define the solution method for the coarsest mesh. There, the operator $K_0^{\mathcal{I}}$ is assembled explicitly and the associated equation is solved with a direct method.

All in all, we obtain the following algorithm for one iteration of the multigrid method on a level j:

$x_j^{\mathcal{I},m+1} \leftarrow MG^\gamma(j, b_j^{\mathcal{I}}, x_j^{\mathcal{I},m})$

1: **if** $j = 0$ **then**
2: Solve $K_0^{\mathcal{I}} x_0^{\mathcal{I}} = b_0^{\mathcal{I}}$
3: **else**
4: Presmooth $\tilde{x}_j^{\mathcal{I}} = (\mathcal{S}_{h,\alpha,\beta}^{\mathcal{I}})^{\nu_1}(x_j^{\mathcal{I},m}, b_j^{\mathcal{I}})$
5: Residual restriction $b_{j-1}^{\mathcal{I}} = \mathcal{R}_j^{j-1}(b_j^{\mathcal{I}} - K_j^{\mathcal{I}} \tilde{x}_j^{\mathcal{I}})$
6: Grid recursion $v_{j-1}^{\mathcal{I}} = MG^\gamma(j-1, b_{j-1}^{\mathcal{I}}, 0)$
7: Correction $\tilde{\tilde{x}}_j^{\mathcal{I}} = \tilde{x}_j^{\mathcal{I}} + \mathcal{P}_{j-1}^j v_{j-1}^{\mathcal{I}}$
8: Postsmooth $x_j^{\mathcal{I},m+1} = (\mathcal{S}_{h,\alpha,\beta}^{\mathcal{I}})^{\nu_2}(\tilde{\tilde{x}}_j^{\mathcal{I}}, b_j^{\mathcal{I}})$
9: **end if**

For $\gamma = 1$ we obtain the well-known V-cycle, $\gamma = 2$ yields the so-called W-cycle. The F-cycle can be defined recursively by an F-cycle on level j, followed by a V-cycle on the same level. Each of these cycles has the cost $\mathcal{O}(N)$ per iteration. Finally, the Full Multigrid algorithm (FMG) combines nested iteration with some multigrid cycle. This way, an approximation to the discrete solution may be obtained in $\mathcal{O}(N)$ operations with an error that is below the discretization error. For further details, we again refer to [25,24].

Note that in order to apply FMG to system (20), certain modifications are necessary. In the case of unconstrained problems, these modifications are straightforward. For control-constrained problems, there is some additional work required. Note that now the box-constraint functions ξ_l, ξ_u have to be explicitly discretized on each level for two reasons: first, the right-hand side vector $r_j^{\mathcal{I}}$ needs to be constructed on all discretization levels $j < J$, and second, in FMG an approximation to the solution instead of the correction is to be prolongated to the next finer level. Thus, the associated prolongation $\tilde{P}_{j-1}^{j,\mathcal{I}}$ has to take the current values of u_{j-1} on the active set into account instead of the zero contributions of the usual prolongation $P_{j-1}^{j,\mathcal{I}}$ in the multigrid cycle. For these reasons, additionally $\mathcal{A}_-$ and $\mathcal{A}_+$ need to be represented on each level.[4] To this end, instead of proceeding as in (35), we restrict $\mathcal{A}_-$ and $\mathcal{A}_+$ according to

$$\mathcal{T}_{\mathcal{A}_{\pm,j-1}} = \{T_i \in \mathcal{T}_{j-1} \mid \cup T_i^s \in \mathcal{T}_{\mathcal{A}_{\pm,j}} \text{ for } s \in \{1,2,3,4\}\}, \quad j = J, \ldots, 1, \tag{38}$$

where $\mathcal{A}_\pm$ stands for $\mathcal{A}_-$ or $\mathcal{A}_+$. After the restriction step (38) we set

$$\mathcal{A}_{j-1} = \mathcal{A}_{-,j-1} \cup \mathcal{A}_{+,j-1} \quad \text{and} \quad \mathcal{T}_{\mathcal{I}_{j-1}} = \mathcal{T}_{j-1} \setminus \mathcal{T}_{\mathcal{A}_{j-1}}. \tag{39}$$

7. Numerical results

In this section, we present numerical results which have been obtained with our multigrid method and our multigrid PDAS method. First, in order to assess the properties of the multigrid method, we consider a model problem with $U_{\text{ad}} = L^2(\Omega)$, i.e. we have no additional constraints on the control. Afterward, we present results for control-constrained problems solved with the PDAS multigrid method.

We introduce the discrete L^2-norm of the error with respect to the control unknown u_h as

$$e_{u_h}^m = \|u_h^m - u^*\|_{L^2,h} = \left(\sum_{T_i \in \mathcal{T}_h} |T_i| |u_{h,i}^m - u^*(x_i)|^2 \right)^{1/2}, \tag{40}$$

with $|T_i|$ and x_i denoting the area and the center of the quadrilateral T_i, respectively and m denoting the iteration index of the multigrid cycle. Analogously we define the errors $e_{y_h}^m$ and $e_{p_h}^m$. The total discrete L^2-error for the mth multigrid iterate is defined as

$$e_h^m = ((e_{y_h}^m)^2 + (e_{u_h}^m)^2 + (e_{p_h}^m)^2)^{1/2}. \tag{41}$$

[4] Note that, in contrast to this, only the sequence $\mathcal{T}_{\mathcal{I}_j}$, $j = J, \ldots, 0$ was needed in the case of the conventional multigrid cycles.

Table 1
Reduction factors ϱ^{20} and $\varrho^{20}_{\text{avg}}$ with respect to the discrete L^2-error e^m_h and different types of multigrid cycles.

		J					
		5	6	7	8	9	10
$V_{1,1}$	ϱ^{20}	1.13_{-1}	1.14_{-1}	1.14_{-1}	1.14_{-1}	1.15_{-1}	1.15_{-1}
	$\varrho^{20}_{\text{avg}}$	1.05_{-1}	1.06_{-1}	1.08_{-1}	1.08_{-1}	1.09_{-1}	1.09_{-1}
$V_{2,1}$	ϱ^{20}	7.07_{-2}	7.21_{-2}	7.30_{-2}	7.31_{-2}	7.31_{-2}	7.31_{-2}
	$\varrho^{20}_{\text{avg}}$	6.57_{-2}	6.79_{-2}	6.84_{-2}	6.89_{-2}	6.93_{-1}	6.93_{-1}
$V_{2,2}$	ϱ^{20}	5.15_{-2}	5.26_{-2}	5.34_{-2}	5.35_{-2}	5.35_{-2}	5.35_{-2}
	$\varrho^{20}_{\text{avg}}$	4.70_{-2}	4.92_{-2}	4.99_{-2}	5.04_{-2}	5.07_{-2}	5.07_{-2}
$F_{1,1}$	ϱ^{20}	8.18_{-2}	8.20_{-2}	8.20_{-2}	8.20_{-2}	8.21_{-2}	8.21_{-2}
	$\varrho^{20}_{\text{avg}}$	7.81_{-2}	7.82_{-2}	7.88_{-2}	7.86_{-2}	7.87_{-2}	7.87_{-2}
$W_{1,1}$	ϱ^{20}	8.18_{-2}	8.20_{-2}	8.20_{-2}	8.20_{-2}	8.21_{-2}	8.21_{-2}
	$\varrho^{20}_{\text{avg}}$	7.80_{-2}	7.82_{-2}	7.88_{-2}	7.86_{-2}	7.87_{-2}	7.87_{-2}

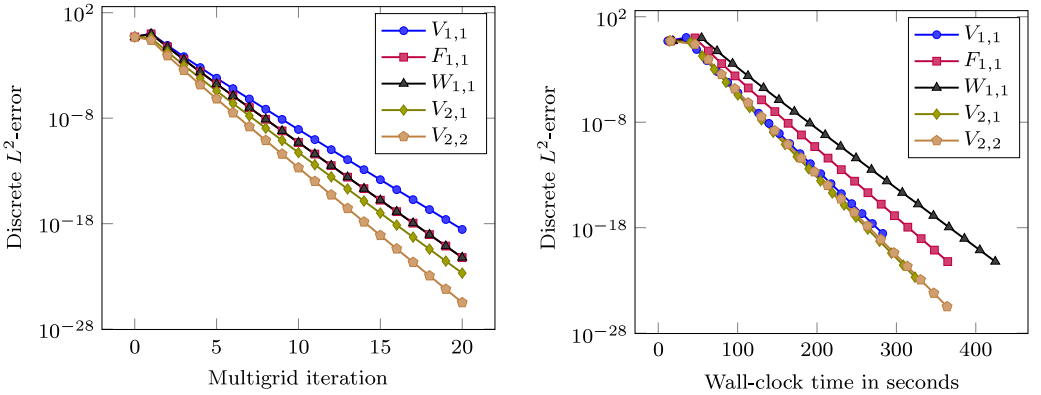

Fig. 1. Reduction of discrete L^2-error e^m_J vs. number of multigrid iterations m (left) and wall-clock time (right). The fine grid mesh size is $h_J = 2^{-10}$.

Furthermore, the residual norm in the mth multigrid iteration is given by

$$\|\text{res}^m_h\|_2 = \|r^{\mathbb{1}}_h - K^{\mathbb{1}}_h x^{\mathbb{1},m}_h\|_2 \tag{42}$$

with $K^{\mathbb{1}}_h$ and $r^{\mathbb{1}}_h$ of (21). For control-constrained problems, we define the errors $\tilde{e}^k_{u_h}$ and $\tilde{e}^k_h$ analogously to (40) and (41), respectively, with k denoting the iteration index of the outer PDAS loop.

7.1. A model problem

We minimize the tracking type functional (1) subject to the constraints

$$-\Delta y - u = 0 \quad \text{in } \Omega, y = 0 \text{ on } \partial\Omega \tag{43}$$

with $\Omega = [0, 1]^2 \in \mathbb{R}^2$ and $U_{\text{ad}} = L^2(\Omega)$. In this case, we have $\mathcal{A}_- = \mathcal{A}+ = \emptyset$ and the PDAS algorithm with system (20) reduces to the solution of a KKT system (12). The weighting parameter in (1) is chosen as $\sigma = 1.0\text{e}{-2}$. The target state $\bar{y}$ as well as the right-hand side f are chosen identically zero. The unique solution of the optimization problem is then zero, and the current iterate is equal to the error. The initial guess is a normalized random vector. The smoothing iteration (27) is employed with $\alpha = \beta = 1$ and as iterative method for the approximate solution of the state and adjoint equation, i.e. in (26), we apply the symmetric point Gauss–Seidel iteration.

In Table 1, we present the asymptotic reduction rate for the error e_h defined in (41) for different types of multigrid cycles. The size of the coarsest mesh is $h_c = 1/4$, the mesh size of the finest grid is then $2^{-(J+2)}$. From Table 1 we clearly observe that the reduction rates are independent of the resolution on the finest mesh. Furthermore, the reduction rates are of the same order as the reduction rates which one obtains when solving the scalar Poisson model problem for cell-centered discretizations with a multigrid method. Stronger smoothing obviously results in better reduction rates. Fig. 1 shows the iteration history for the different tested multigrid cycles on the fixed level $J = 7$. The reduction rates are constant. As can be seen in Fig. 1(right), the better reduction per iteration does not always pay off in terms of wall-clock time due to a higher cost per iteration. In particular, the relatively small gain in convergence speed does not justify the higher cost for a W-cycle. Here, the most efficient cycle is the $V_{2,2}$-cycle. The performance for all tested V-cycles is roughly the same, i.e. a doubled amount of smoothing results in about half as much iterations at a doubled cost per iteration. These findings are also in agreement with results for the scalar Poisson model problem.

Table 2
Discrete L^2-error e_h and wall-clock time in seconds for one FMG cycle. On each level, one multigrid iteration with a $V_{1,1}$-cycle is used.

J	h_J	n	Time (s)	Ratio	e_h	Ratio
5	1/128	49,152	0.1680	–	6.06389_{-5}	–
6	1/256	196,608	0.7578	4.511	1.59350_{-5}	0.263
7	1/512	786,432	3.6797	4.856	4.06443_{-6}	0.255
8	1/1024	3145,728	16.4102	4.459	1.02422_{-6}	0.252
9	1/2048	12,582,912	68.3516	4.165	2.56857_{-7}	0.251
10	1/4096	50,331,648	276.5470	4.045	6.42940_{-8}	0.250

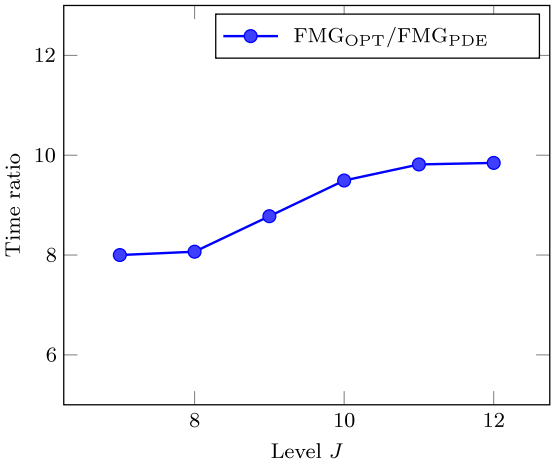

Fig. 2. Ratio of wall-clock time in seconds for FMG/Opt and FMG/PDE.

We now solve the same problem using the full multigrid approach. To this end, on each level in FMG a $V_{1,1}$-cycle is employed. Table 2 shows the wall-clock time and the discrete L^2-error e_h after one cycle of the FMG iteration. The discretization parameter h_J is the meshwidth on the finest level J, the total number of fine grid unknowns n of the optimality system thus is given by $n = 3h_J^{-2}$. Since the number of unknowns quadruples from one level to the next, we expect a corresponding fourfold increase in computational time. This is clearly observed from the presented data, thus showing that the total cost for one FMG iteration is indeed $\mathcal{O}(n)$. The employed discretization is superconvergent with second order in the discrete L^2-norm. Therefore, we should expect a decrease of the error with a factor of 4, if the mesh resolution of the finest level is doubled. In the last two columns of Table 2 we indeed see that this is the case, i.e. after one FMG iteration, the error is of the order of the discretization error on the finest mesh. All in all, we conclude that the FMG solves the discrete optimal control problem up to discretization error accuracy with optimal complexity $\mathcal{O}(n)$.

In order to give a rough estimate of the computational cost needed to solve the optimality system compared to the cost needed to solve the constraint PDE only, let us now consider the cost for one application of our smoothing method in more detail. To this end, recall that $N = h_J^{-2}$ is the number of each of the three unknowns y_h, u_h, p_h appearing in the discretized optimal control problem. The cost C_N for the relaxation of the state and adjoint equation by (26) is $\mathcal{O}(N)$. Note that this is also the cost order for the smoothing iteration when solving the constraint PDE with a multigrid method. A rough (lower bound) estimate of the overall solution cost of the optimality system is as follows: from the definition of the smoother (27) it follows that a cost of at least $2C_N$ operations incurs by the matrix–vector product with K_h^I. Here, we have neglected all operations not involving the discretized PDE operator L_h, such as multiplications with the mass matrices. The application of the preconditioner C_h again contributes a cost of $2C_N$ for the constraint blocks given by L_h and L_h^T. Finally, the conjugate gradient iteration with B_Z adds a cost of $2C_N$ for the first iteration and again a cost of $2C_N$ for the initialization of the cg method. Thus we obtain a cost count of $8C_N$ as a lower bound estimate for the total cost for one application of the smoother, counting only the operations which involve the constraint blocks L_h and L_h^T. Fig. 2 shows the ratio of the wall-clock times needed for the FMG solution of the optimal control problem versus the solution time of the underlying PDE problem, again using FMG. Here we see values between 8 and 10. Bearing in mind that the cost was estimated neglecting several operations such as multiplications with mass matrices, we conclude that these values which were achieved with our implementation are more than reasonable. In summary, we have seen that the discretized optimal control problem can be solved with optimal complexity at a small multiple of the cost which is required for the solution of the underlying PDE alone.

Let us finally remark on the regularization parameter σ in the objective functional (1). Depending on σ, there is a limitation on the size of the coarsest possible grid which results from the specific form of the reduced Hessian (and not only its approximation!). For our linear model problem, we obtained the condition (34), which requires that $\sigma \geq ch_1^4$ with

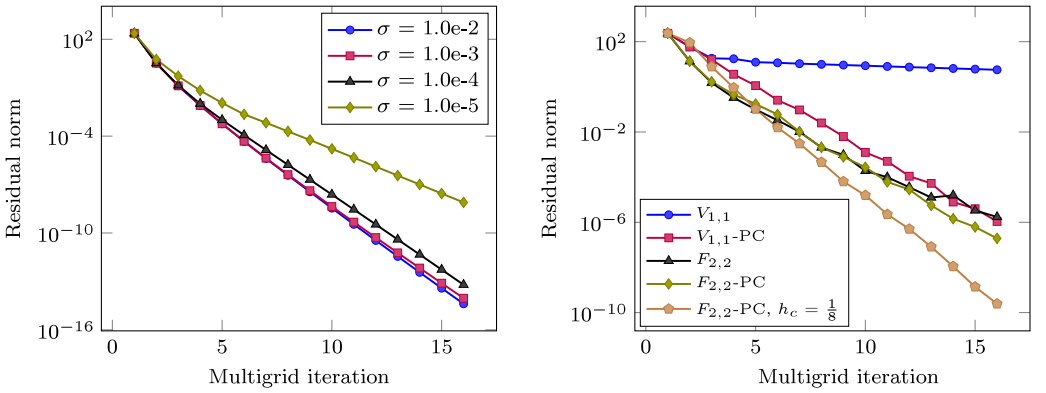

Fig. 3. Convergence for varying regularization parameter σ.

a small constant $c \sim \frac{1}{4}$. As long as this condition is met, the convergence rates do not deteriorate with decreasing value of σ but remain robust. This is demonstrated in Fig. 3. In the left plot, we see the iteration history of the residual norm for different values of σ on level $J = 10$. The mesh size of the coarsest level is given as $h_c = 1/4$. Thus $h_1^4/4 \approx 6.104\text{–}5$. We clearly see for values $\sigma \geq 1.0\text{e}{-4}$ that the convergence rates do not deteriorate. Only for smaller values of σ a degradation of convergence is observed. Moreover, a further decrease of the regularization parameter will lead to divergence. This can be seen from the right part of Fig. 3. Here the black curve shows the stalled convergence for $\sigma = 1.0\text{e}{-6}$ and the same coarse mesh. Using the multigrid method as a preconditioner and possibly more robust cycles such as the $F(2, 2)$-cycle, acceptable rates could be established again. We also clearly see that a reduction of the coarse grid size regains perfect linear convergence. Altogether, we have demonstrated that the reduction rates are independent of σ and that they are already very good for the cheap $V_{1,1}$-cycle as long as the condition $\sigma \geq ch_1^4$ on the next to coarsest mesh is met. Thus, fairly small values of σ can be computed for reasonable coarse meshes. Note that reduced space methods have no such possibility of alleviating the ill-conditioning of the reduced Hessian for small regularization parameters. Thus, for such methods the corresponding iteration numbers of outer solves by e.g. the cg method will inevitable depend on the regularization parameter even if the inner solves are done with multigrid.

7.2. General diffusion constraints

In this section we consider the general diffusion equation

$$-\nabla \cdot D\nabla y = u, \tag{44}$$

with homogeneous Dirichlet boundary conditions as constraints. We minimize again the tracking type functional (1), all other parameters stay the same as before unless noted. First, we consider the diffusion tensor

$$D = \begin{bmatrix} 11 & 9 \\ 9 & 13 \end{bmatrix}. \tag{45}$$

Note that this choice of D introduces a strong diagonal component of the flux. When solving the diffusion equation with multigrid methods where only pointwise smoothing is employed, degradation of convergence occurs. Therefore, smoothers with stronger coupling of unknowns have to be employed. Here, we use the incomplete LU factorization with zero fill (ILU) as the iteration in (26). Again, we use the inexpensive $V_{1,1}$-cycle. Fig. 4(left) shows the discrete L^2-error $e_{u_h}^m$ defined in (40) and the reduction of the residual norm (right). Note here that the error $e_{u_h}^m$ is measured against a non-vanishing exact solution u^*. Thus, the final error $e_{u_h}^m$, $m \geq 5$, reflects the discretization error. Analogous results are obtained for the errors e_{y_h} and e_{p_h} and will therefore not be given here. The resolution of the finest grid is given by $h = 2^{-(J+3)}$, i.e. $h_c = 2^{-3}$. The average residual reduction rate for the computation with finest level $J = 7$ is 0.0987, which is in excellent agreement with the multigrid solution for the underlying PDE problem.

Another example which requires sophisticated smoothing in the inner iterations is given by discretizing the Laplace equation on a non-uniform grid. Fig. 5 shows a deformation of the unit square. The deviation from the unit square boundary on each side is given by the parameter δ, which is 0.25 for the depicted mesh. The discretization of the Laplace operator on this grid yields a full diffusion tensor and strong anisotropies in the discrete operator. This is well known to prevent the successful use of pointwise smoothing methods. Again, stronger coupling of the unknowns is required within the smoothing method. Here, we employed an alternating line Gauss–Seidel method (ALGS) as smoothing iteration (26). Fig. 6 shows the residual reduction for two different values of the contraction δ, namely 0.05 (left) and 0.25. For comparison, we show the iteration history for the two cases of a $V_{1,1}$- and a $V_{2,2}$-cycle applied to the discrete constraint PDE alone. For the smoothing of the KKT system we have used the $V_{1,1}$-cycle, i.e. $\nu = 1$, and $\alpha = \beta = 2$. We again obtain convergence rates that closely

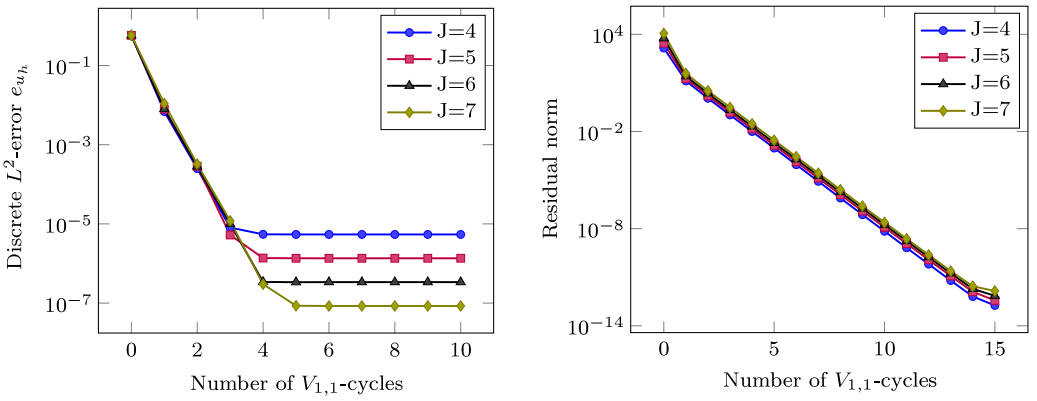

Fig. 4. Convergence for diffusion equation with diffusion tensor D.

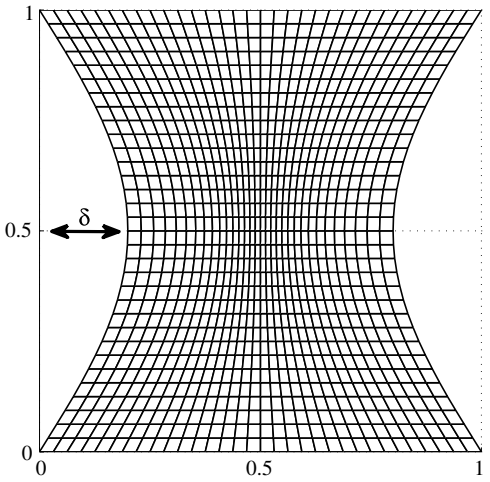

Fig. 5. Non-uniform mesh with indentation for $\delta = 0.25$.

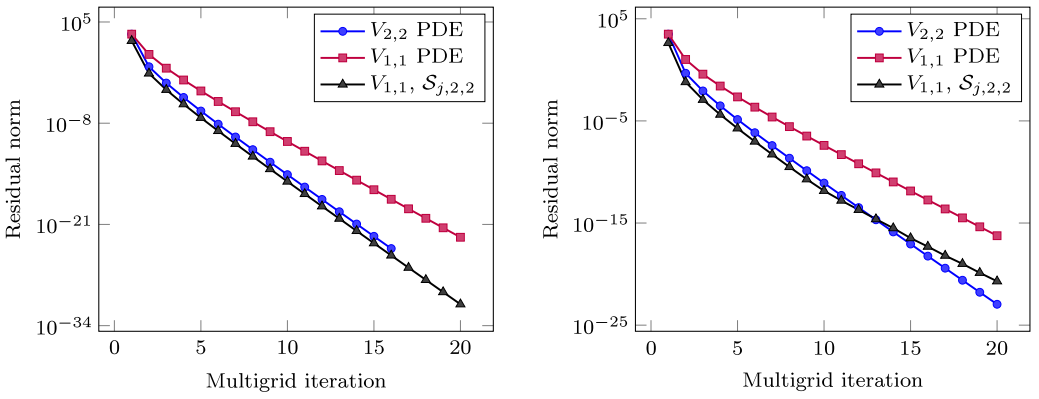

Fig. 6. Residual norm for the optimal control problem on the non-uniform mesh, $\delta = 0.05$ (left) and $\delta = 0.25$.

match those of the multigrid method for the PDE problem. This can be inferred from the data in Table 3. There, convergence rates for the three different cases are given for different levels J of the discretization.

The results of this section show that our overall method easily allows us to exploit the knowledge of sophisticated multigrid solution methods for the constraint PDE by adapting the smoothing iteration for the inner linear systems in a suitable way.

Table 3
Convergence rates for PDE only and KKT system.

J	4	5	6	7	8
$V_{2,2}$ PDE	0.0390	0.0502	0.0571	0.0622	0.0654
$V_{1,1}$ PDE	0.0997	0.1155	0.1217	0.1301	0.1313
$V_{1,1}$ KKT	0.1332	0.1352	0.1363	0.1350	0.1316

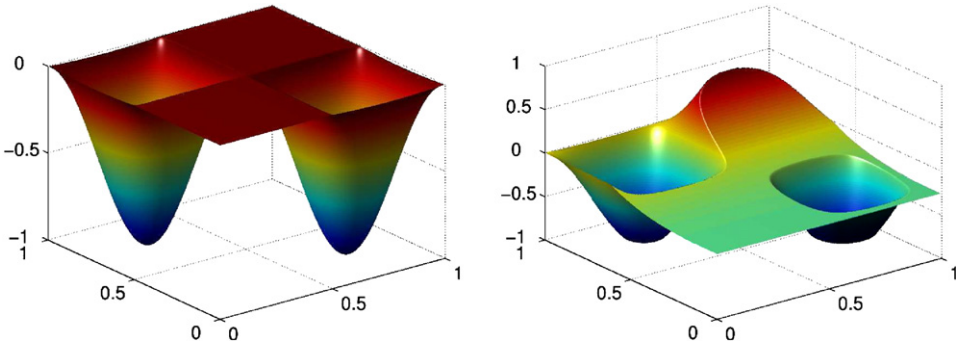

Fig. 7. Computed constrained optimal controls for the two test cases.

7.3. Control-constrained problems

Now we present numerical results for the solution of optimal control problems with additional constraints on the control. We consider two test cases. For both problems, the linear PDE constraint is given by the Poisson equation. As a first example we consider the unilateral constraint given by an upper bound

$$u \leq \xi_u = 0. \tag{46}$$

The second test case is given by the bilateral constraints

$$\xi_l = \begin{cases} -0.75 & \text{for } y \leq 0.5 \\ -0.9 & \text{for } y > 0.5 \end{cases} \quad \text{and} \quad \xi_u = y^3 - 0.5. \tag{47}$$

In Fig. 7, we depict the corresponding computed controls. Fig. 8 serves to illustrate the restriction of the inactive set $\mathcal{I}$ onto different coarse grids as given by (35). Here, the inactive set $\mathcal{I}$ for a unilaterally constrained problem with the upper bound ξ_u from the second test case is shown. The shaded region represents the active set $\mathcal{A}_+$ and is just given by $\mathcal{T}_g \setminus \mathcal{T}_{\mathcal{I}_g}$. We here show the inactive set as computed in the last PDAS iteration, i.e. at the discrete solution u_h^*.

In Fig. 9(left), we give the error $\tilde{e}_{u_h}^k$ for each step k of the outer PDAS iteration. Clearly, superlinear convergence is observed as predicted by the theory. On the right, the error with respect to the upper bound, given by

$$e_{\text{bnd}}^u = \max_{\mathcal{T}_h}(u_h - \Pi_h \xi_u), \tag{48}$$

is shown. Again, superlinear convergence is obtained. Note that e_{bnd}^u vanishes for the discrete solution and therefore e_{bnd}^u is not plotted in the fourth iteration.

Furthermore, Fig. 10 shows the residual reduction for the multigrid solution of system (20) in each PDAS step. In the first iteration of the PDAS method we have $\mathcal{A}_+ = \emptyset$ and thus the unconstrained problem is solved. We see that the reduction rates are the same for each following PDAS step and thus are independent of the structure of the active and inactive sets at each PDAS step. In Table 4 we present the discrete L^2-error e_J^u obtained in the final PDAS step. In the second and third columns, we give the errors and associated ratios which have been obtained by solving (20) with the standard multigrid solver, i.e. iterations with the $V_{1,1}$-cycle have been performed until the stopping criterion applied. Clearly convergence of second order is observed. In the fourth and fifth columns, we give the same data which has been computed by using just one iteration of the full multigrid for each PDAS step. Within the FMG, the same $V_{1,1}$-cycle has been used. The absolute error is roughly one order of magnitude larger than that for the fully converged multigrid solution, but it still reduces at the same rate. In the last two columns, we again present the analogous data, however this time the FMG has been followed by one additional $V_{1,1}$-cycle. This reduces the error to the same order of magnitude as that of the conventional multigrid solver. These results resemble the situation of scalar elliptic problems. Under the assumption that the convergence rate of the employed multigrid cycle is smaller than 1/6, one FMG iteration yields an approximate solution with an error of $(5/2)ch_J^2$ and one additional multigrid cycle reduces that error below $(1/2)ch_J^2$. Here, c is the constant from the error estimate $\|u_J^* - u^*\| \leq ch_J^2$. In Table 5, we see the growth of the active set $\mathcal{A}_+$, given by the increase in terms of the corresponding cells T_i, for each outer PDAS iterations (for the first iteration, the actual size of $\mathcal{A}_+$ is given). The results in the first two rows

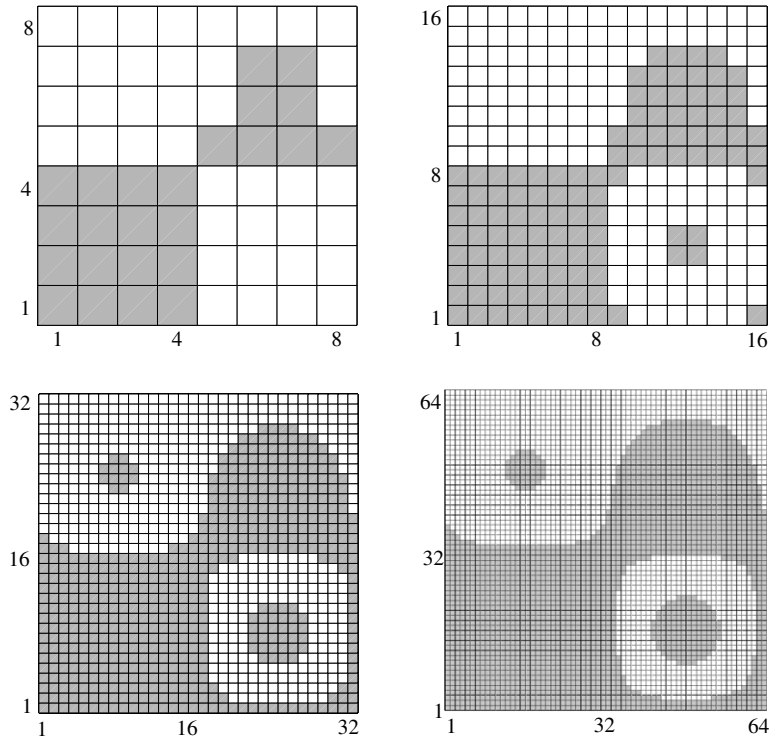

Fig. 8. Inactive set $\mathcal{I}^*$ and active set $\mathcal{A}^*$ (shaded) generated by coarsening (35) on levels $J = 0, 1, 2, 3$ for the second test case.

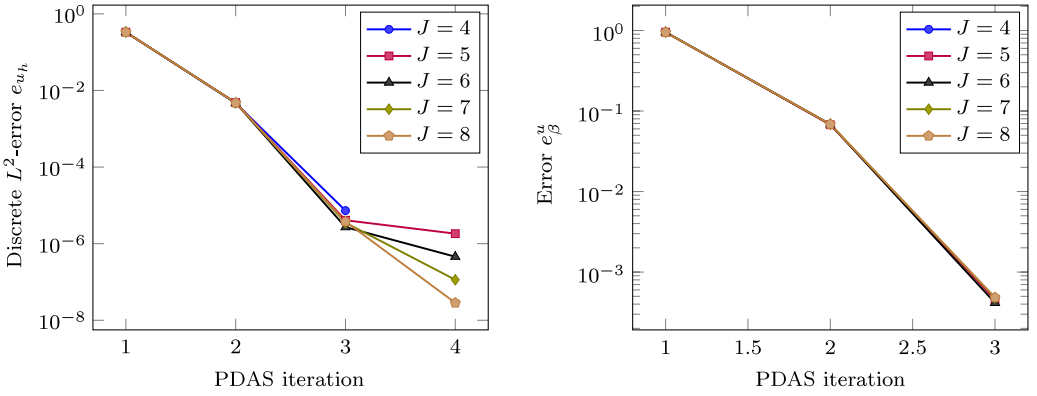

Fig. 9. Convergence of the outer PDAS iteration, L^2-error $\tilde{e}_{u_h}$ (left) and e_{bnd} (right).

correspond to the case where only one or two multigrid iterations per outer iteration are performed. The third row shows the results if a stopping criterion for the residual norm of $\varepsilon \leq 10_{-10}$ is used for the solution of the inner EQP system. The total number of necessary multigrid iterations is higher if the inner systems are solved to an a priori specified accuracy. On the other hand, if only a fixed number of multigrid cycles is employed for the solution of the inner systems, the mesh-independence of the outer iterations is lost. The most efficient method is obtained if the inner systems are solved using one cycle of the FMG solver. Therefore, we have employed the FMG as solver for the inner systems in the second test case. The corresponding results are given in Fig. 11. As before, we present the L^2-error $\tilde{e}_{u_h}$ (left) and the error e^l_{bnd} (right). Again, superlinear convergence of the outer iteration is clearly observed. The final error obtained for $k = 4$ is convergent of second order with respect to h_J. The corresponding data is given in Table 6, where we also list the computing time in seconds. The numbers confirm the optimal complexity $\mathcal{O}(3h_J^{-2})$ of the FMG solver.

Let us briefly comment on the robustness of the outer PDAS iteration with respect to the regularization parameter σ in the cost functional (1). To this end, we consider the example problem (46) with different values of σ. In Fig. 12, we give the discrete L^2-error of the control u_h (left) and the error e^u_{bnd} (right) for the finest level $J = 8$. The superlinear convergence is obviously obtained in all cases. However, there is an, albeit very mild, dependence on the actual value of σ. For decreasing

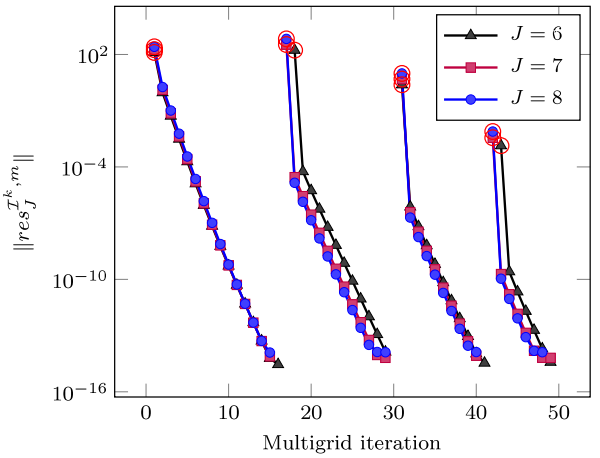

Fig. 10. Residual reduction of multigrid for the EQP in each PDAS step. The red circle indicates the respective initial residual. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Discrete L^2-error of the control u_J^* for different levels J obtained in the final step of the PDAS iteration with fully converged $V_{1,1}$-cycles, FMG and FMG plus one additional $V_{1,1}$-cycle.

J	$V_{1,1}$		FMG		FMG + $V_{1,1}$	
	$e^4_{u_J}$	Ratio	$e^4_{u_J}$	Ratio	$e^4_{u_J}$	Ratio
3	2.9225_{-5}	–	3.1906_{-4}	–	4.8750_{-5}	–
4	7.3051_{-6}	2.49_{-1}	8.3699_{-5}	2.62_{-1}	1.2529_{-5}	2.57_{-1}
5	1.8262_{-6}	2.50_{-1}	2.1282_{-5}	2.54_{-1}	3.1626_{-6}	2.52_{-1}
6	4.5655_{-7}	2.50_{-1}	5.3510_{-6}	2.51_{-1}	7.9325_{-7}	2.50_{-1}
7	1.1414_{-7}	2.50_{-1}	1.3403_{-6}	2.50_{-1}	1.9854_{-7}	2.50_{-1}
8	2.8534_{-8}	2.50_{-1}	3.3527_{-7}	2.50_{-1}	4.9656_{-8}	2.50_{-1}

Table 5
Increments of active set $\mathcal{A}_+$ for different accuracies of inner system solutions.

	1	2	3	4	5	6	7	Total
1	2044,753	+703069	+104690	+12471	+877	+80	+2	
	1	1	1	1	1	1	1	7
2	2007,484	+185614	+3921	+27				
	2	2	2	2				8
$\varepsilon = 10_{-10}$	1998,192	+98254	+706					
	13	10	9					32

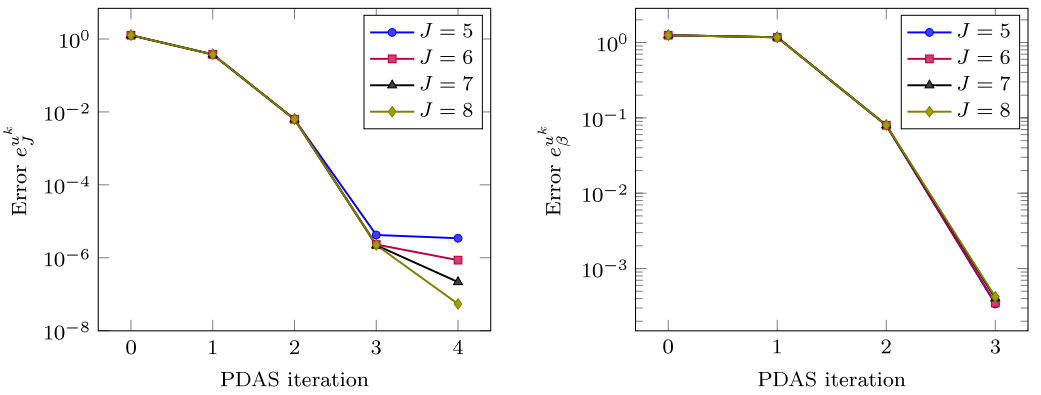

Fig. 11. Convergence of the outer PDAS iterations. L^2-error $\tilde{e}_{u_h}$ (left) and e_{bnd} (right).

σ, the number of outer PDAS iterations increases slightly. This observation can be made independent of the solver used for the solution of the inner systems and was also reported in [2].

Table 6
Discrete L^2-error of the control u_J^* for the second test case and wall-clock time for the solution with FMG plus one additional $V_{1,1}$-cycle.

J	$e^4_{u_J}$	Ratio	Time (s)	Ratio
5	3.4246_{-6}	–	1.2723_{+1}	–
6	8.5909_{-7}	2.51_{-1}	5.9187_{+1}	4.65
7	2.1570_{-7}	2.51_{-1}	2.6605_{+2}	4.49
8	5.4015_{-8}	2.50_{-1}	1.1127_{+3}	4.18
9	1.3530_{-8}	2.50_{-1}	4.5630_{+3}	4.10

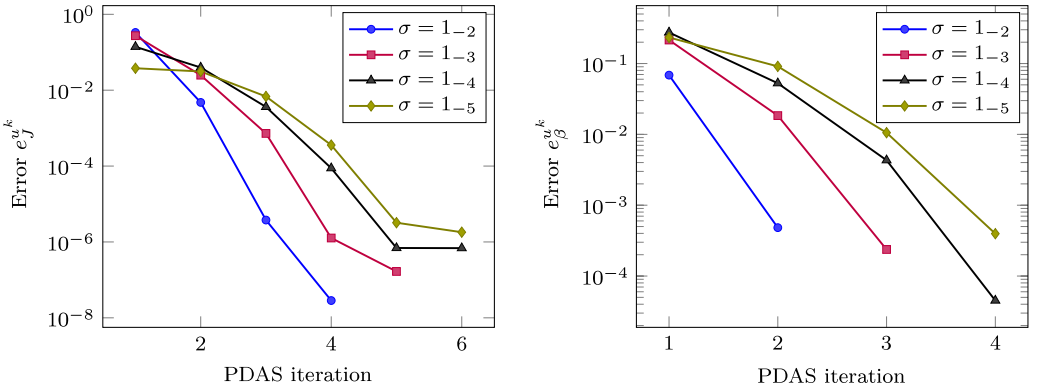

Fig. 12. Convergence of the outer PDAS iterations for different values of the regularization parameter σ. L^2-error $\tilde{e}_{u_h}$ (left) and e^u_{bnd} (right).

Table 7
Size of the active set $\mathcal{A}_+$ and error e_{bnd} for different levels $J = 6, 7, 8$ and for regularization parameters $\sigma = 1.0\text{e}{-2}$ and $\sigma = 1.0\text{e}{-5}$.

σ	J	1	2	3	4	5
1.0_{-2}	6	124,952	+6096	+24	–	–
		6.843_{-2}	4.149_{-4}	0.0	–	–
	7	499,670	+24474	+144	–	–
		6.848_{-2}	4.800_{-4}	0.0	–	–
	8	1998,192	+98254	+706	–	–
		6.856_{-2}	4.820_{-4}	0.0	–	–
1.0_{-5}	6	101,450	+20528	+7744	+1330	+20
		2.338_{-1}	9.117_{-2}	8.090_{-3}	3.379_{-4}	0.0
	7	405,590	+82330	+30916	+5342	+110
		2.326_{-1}	8.836_{-2}	9.050_{-3}	3.848_{-4}	0.0
	8	1622,360	+329360	+123534	+21352	+546
		2.331_{-1}	9.109_{-2}	1.055_{-2}	3.960_{-4}	0.0

Finally, in Table 7 we give the size of the active set and the error e_{bnd} for $\sigma = 1.0\text{e}{-2}$ and $1.0\text{e}{-5}$ and values of $J = 6, 7, 8$. From the given data we conclude that for a fixed value of σ, the mesh-independence of the outer iteration is not affected.

8. Conclusions and outlook

We presented a multigrid method for optimality systems which arise from the discretization of constrained optimal control problems of tracking type. The method was applied to test problems commonly found in the literature. Certainly some additional work will be required in order to extend the present approach to real-life applications, however the results are very promising. We observed convergence rates for the optimization problem that closely match those obtained for the underlying elliptic PDE problem which serves as constraint. The FMG method provides a solver with optimal complexity. The total cost (measured in wall-clock time) for the solution of optimal control problems is just a small multiple of the cost for the solution of the constraint equations only. For problems with additional constraints on the control, the multigrid method was used to solve the equality-constrained subproblems which arise at each step of a primal–dual active set strategy. It was shown in numerical experiments that superlinear convergence of the outer iteration is obtained provided that the EQP is solved accurately enough. Furthermore, it was demonstrated that a highly efficient overall solver is obtained if the full multigrid approach is employed for the solution of the inner systems.

Acknowledgments

The authors gratefully acknowledge the support from the Deutsche Forschungsgemeinschaft DFG through the Sonderforschungsbereich 611 "Singular Phenomena and Scaling in Mathematical Models".

Appendix. Local Fourier analysis of the smoothing iteration

The aim of the local Fourier analysis, or local mode analysis, as it is termed in [23], is to give *quantitative estimates* of the smoothing and convergence factors for a practical multigrid method. In this respect it differs from classical convergence proofs which mostly provide qualitative results. The local Fourier analysis is seen as an essential tool for the design of an efficient multigrid method for general problems. Here we focus on the smoothing analysis based on the local Fourier analysis, i.e. the aim of this appendix is to determine (an estimate of) the smoothing factor $\mu(\mathcal{S}_{j,\alpha,\beta})$ for smoothing iterations of the type (27). For a detailed introduction into the topic of the local Fourier analysis we refer to [25].

The local Fourier analysis considers the effect of all appearing operators in the multigrid method when applied to the frequency functions

$$\phi(\theta, x) = \mathrm{e}^{\mathrm{i}\theta\cdot x/h} = \mathrm{e}^{\mathrm{i}\theta_1 x_1/h}\mathrm{e}^{\mathrm{i}\theta_2 x_2/h} \tag{49}$$

on the infinite grid[5]

$$G_h = \{x = hk = (hk_1, hk_2) | k \in \mathbb{Z}^2\}. \tag{50}$$

Thus the local Fourier analysis neglects the boundary treatment within the multigrid algorithm and in this sense only predicts rates which can be obtained provided a proper boundary treatment is conducted. Due to the periodicity of (49) it is sufficient to consider the frequencies

$$\theta = (\theta_1, \theta_2) \in [-\pi, \pi)^2. \tag{51}$$

With G_h we associate the infinite coarse grid G_H, which is defined analogously to (50) for the coarse mesh size $H = 2h$. Due to aliasing, on G_H only those frequencies with $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2})^2$ can be distinguished, which leads to the definition that

$$\phi(\theta, x) \text{ is a low frequency component, if } \quad \theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)^2, \tag{52}$$

$$\phi(\theta, x) \text{ is a high frequency component, if } \quad \theta \in [-\pi, \pi)^2 \setminus \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)^2. \tag{53}$$

For $\theta \in [-\pi, \pi)^2$ all functions $\phi(\theta, x)$ are the formal eigenfunctions of any discrete operator represented by a difference stencil, i.e. the relation

$$L_h\phi(\theta, x) = \tilde{L}_h(\theta)\phi(\theta, x) \tag{54}$$

holds with

$$\tilde{L}_{h(\theta)} = \sum_{\mathbf{j}} s_{\mathbf{j}} \mathrm{e}^{\mathrm{i}\theta\cdot\mathbf{j}}, \tag{55}$$

where $\mathbf{j} \in \mathbb{Z}^2$ and $s_{\mathbf{j}}$ are the stencil elements of L_h. Due to (54) we call $\tilde{L}_h(\theta)$ the formal eigenvalue or *symbol* of the operator L_h. For the standard five-point stencil of the discrete Laplacian, for example, one obtains

$$\tilde{L}_h(\theta) = \frac{1}{h^2}(4 - (\mathrm{e}^{\mathrm{i}\theta_1} + \mathrm{e}^{\mathrm{i}\theta_2} + \mathrm{e}^{-\mathrm{i}\theta_1} + \mathrm{e}^{-\mathrm{i}\theta_2})) = \frac{2}{h^2}(2 - \cos\theta_1 - \cos\theta_2). \tag{56}$$

In order to apply the local Fourier-based smoothing analysis we have to assume that the smoothing iteration can be given by a local linear splitting which can be written in stencil notation as

$$L_h^+\tilde{x}_h + L_h^- x_h = f_h, \tag{57}$$

where, as in Section 6, $\tilde{x}_h$ denotes the smoothed approximation of x_h. This is a natural assumption for many classical methods such as ω-JAC and GS-LEX, however it is not satisfied for, e.g. coloring-based Gauss–Seidel methods. For the necessary modifications to treat for instance red–black Gauss–Seidel within the local Fourier analysis framework we refer to [25]. Note that the use of the conjugate gradient iteration for the approximate inversion of (25) prevents to define (27) by a splitting of a system corresponding to (57). However, we can apply the local Fourier analysis to the iteration $\mathcal{S}_{j,1}$ as defined in Section 6, since in this case a splitting is obtained easily.

[5] For notational simplicity, we confine ourselves to the case of a uniform mesh size h in both coordinate directions.

From (57) we obtain the symbol of the corresponding smoothing operator S_{L_h} as

$$\tilde{S}_{L_h}(\theta) = -\frac{\tilde{L}_h^-(\theta)}{\tilde{L}_h^+(\theta)}, \tag{58}$$

where $\tilde{L}_h^+(\theta)$ and $\tilde{L}_h^-(\theta)$ are the symbols of L_h^+ and L_h^-, respectively. The smoothing rate $\mu_{\mathrm{LFA}}(S_{L_h})$ is then defined as the largest possible amplification factor with respect to the high frequency components,

$$\mu_{\mathrm{LFA}}(S_{L_h}) = \sup\left\{|\tilde{S}_{L_h}(\theta)| : \theta \in [-\pi,\pi)^2 \setminus \left[-\frac{\pi}{2},\frac{\pi}{2}\right)^2\right\}. \tag{59}$$

In order to perform the local Fourier smoothing analysis for (27) we need the extension of the definitions (54) and (32) to systems of equations. To this end, we introduce the frequency functions

$$\Phi(\theta,x) = (1,\ldots,1)^T\phi(\theta,x). \tag{60}$$

Then we obtain

$$K_h\Phi(\theta,x) = \tilde{K}_h(\theta)\Phi(\theta,x) \tag{61}$$

with the symbol

$$\tilde{K}_h(\theta) = \begin{pmatrix} \tilde{K}_h^{1,1}(\theta) & \ldots & \tilde{K}_h^{1,Q}(\theta) \\ \vdots & & \vdots \\ \tilde{K}_h^{Q,1}(\theta) & \ldots & \tilde{K}_h^{Q,Q}(\theta) \end{pmatrix}, \tag{62}$$

where $\tilde{K}_h^{l,q}(\theta)$, $1 \le l,q \le Q$ are the symbols of the scalar discrete operators $K_h^{l,q}$ of the $Q \times Q$-system given by K_h. Corresponding to the splitting (57) in the scalar case we assume that (27) can be given by a splitting

$$K_h^+\tilde{w}_h + K_h^- w_h = f_h, \tag{63}$$

with the associated symbols $\tilde{K}_h^+(\theta)$, $\tilde{K}_h^-(\theta)$. The smoothing factor for a system of equations is now defined by

$$\mu_{\mathrm{LFA}}(S_h) = \sup\left\{\rho((\tilde{K}_h^+(\theta))^{-1}\tilde{K}_h^-(\theta)) : \theta \in [-\pi,\pi)^2 \setminus \left[-\frac{\pi}{2},\frac{\pi}{2}\right)^2\right\}, \tag{64}$$

with ρ denoting the spectral radius. As mentioned before, $S_{j,1}$ is amenable to a splitting of the form (63), which is given by

$$K_h^+ = \begin{pmatrix} & & L_h^{T,+} \\ & \hat{H}_Z & -M_h \\ L_h^+ & -M_h & \end{pmatrix}, \qquad K_h^- = \begin{pmatrix} M_h & & L_h^{T,-} \\ & & \\ L_h^- & & \end{pmatrix}, \tag{65}$$

with L_h^+, L_h^- and $L_h^{T,+}$, $L_h^{T,-}$ resulting from a splitting (57) of the state and adjoint operators L_h and L_h^T, respectively. For notational simplicity in the following we assume that $L_h = L_h^T$. The symbols corresponding to (65) are then given by

$$\tilde{K}_h^+(\theta) = \begin{pmatrix} & & \tilde{L}_h^+(\theta) \\ & \tilde{\hat{H}}_Z(\theta) & -\tilde{M}_h(\theta) \\ \tilde{L}_h^+(\theta) & -\tilde{M}_h(\theta) & \end{pmatrix}, \qquad \tilde{K}_h^-(\theta) = \begin{pmatrix} \tilde{M}_h(\theta) & & \tilde{L}_h^-(\theta) \\ & & \\ \tilde{L}_h^-(\theta) & & \end{pmatrix}. \tag{66}$$

With

$$(\tilde{K}_h^+(\theta))^{-1} = \begin{pmatrix} \tilde{A}_h(\theta)\tilde{H}_h(\theta)(\tilde{L}_h^+(\theta))^{-1} & \tilde{A}_h(\theta) & (\tilde{L}_h^+(\theta))^{-1} \\ \tilde{H}_h(\theta)(\tilde{L}_h^+(\theta))^{-1} & (\tilde{\hat{H}}_Z(\theta))^{-1} & 0 \\ (\tilde{L}_h^+(\theta))^{-1} & 0 & 0 \end{pmatrix} \tag{67}$$

we obtain

$$\tilde{S}_h(\theta) = \begin{pmatrix} \tilde{A}_h(\theta)\tilde{H}_h(\theta)\tilde{A}_h(\theta) + \tilde{S}_{L_h}(\theta) & 0 & \tilde{A}_h(\theta)\tilde{H}_h(\theta)\tilde{S}_{L_h}(\theta) \\ \tilde{H}_h(\theta)\tilde{A}_h(\theta) & 0 & \tilde{H}_h(\theta)\tilde{S}_{L_h}(\theta) \\ \tilde{A}_h(\theta) & 0 & \tilde{S}_{L_h}(\theta) \end{pmatrix} \tag{68}$$

as the symbol of the smoothing operator corresponding to the iteration $\mathcal{S}_{j,1}$. Both in (67) and (68) we have introduced the abbreviations

$$\tilde{A}_h(\theta) = (\tilde{L}_h^+(\theta))^{-1}\tilde{M}_h(\theta) \quad \text{and} \quad \tilde{H}_h(\theta) = (\tilde{\hat{H}}_Z(\theta))^{-1}\tilde{M}_h(\theta). \tag{69}$$

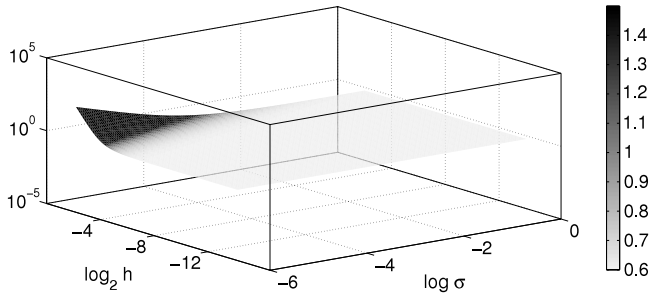

Fig. 13. The smoothing factor $\mu_{\mathrm{LFA}}(S_j)$ (72) for the smoothing iteration $\mathcal{S}_{j,1}$ with ω-JAC as constraint smoother, plotted as a function of the regularization parameter $\sigma \in [1.0, 1.0_{-6}]$ and the mesh size $h_j \in [2^{-2}, 2^{-12}]$.

Furthermore, note that $\tilde{S}_{L_h}(\theta)$ (cf. (58)) is the symbol of the smoothing iteration employed for the constraints L_h, i.e it is based on a splitting of L_h. According to (64), the smoothing factor $\mu_{\mathrm{FLA}(S_h)}$ is given by the largest absolute eigenvalue of (68), which we determine by computing the roots of the polynomial

$$\det(\tilde{S}_h(\theta) - \lambda) = \lambda((\tilde{A}_h(\theta)^2\tilde{H}_h(\theta) + \tilde{S}_{L_h}(\theta) - \lambda)(\lambda - \tilde{S}_{L_h}(\theta)) + \tilde{A}_h(\theta)^2\tilde{H}_h(\theta)\tilde{S}_{L_h}(\theta)). \tag{70}$$

The two non-zero roots are given by

$$\lambda_{1,2} = \tilde{S}_{L_h}(\theta) + \frac{\tilde{A}_h(\theta)^2\tilde{H}_h(\theta)}{2} \pm \sqrt{\frac{\tilde{A}_h(\theta)^2\tilde{H}_h(\theta)}{2}\left(2\tilde{S}_{L_h}(\theta) + \frac{\tilde{A}_h(\theta)^2\tilde{H}_h(\theta)}{2}\right)}. \tag{71}$$

From (71) we gather that $\mu_{\mathrm{LFA}}(S_h)$ can be considered as a perturbation of the constraint smoothing factor. This perturbation depends on $\tilde{S}_{L_h}(\theta)$ and on additional terms corresponding to the smoothing of the control component, which are governed by $\tilde{H}_h(\theta)$. More insight can be obtained if we consider the model problem where L_h is the discrete Laplacian. Furthermore, we eliminate the explicit dependence of $\rho(\tilde{S}_h(\theta))$ on θ by considering ω-JAC as constraint smoother with the optimal value $\omega = 4/5$ for this case. Then we obtain $\tilde{S}_{L_h}(\theta) = 3/5$. For ω-JAC one obtains $(\tilde{L}_h^+(\theta))^{-1} = \omega/4$, and since $\tilde{H}_h(\theta) = 1/\sigma$ this yields $\tilde{A}_h(\theta) = \frac{1}{5}h^2$. Substituting these quantities into (71) the smoothing factor for the iteration $\mathcal{S}_{j,1}$ is obtained as

$$\mu_{\mathrm{LFA}}(S_{j,1}) = \frac{3}{5} + \frac{h_j^4}{50\sigma} + \sqrt{\frac{h_j^4}{50\sigma}\left(\frac{6}{5} + \frac{h_j^4}{50\sigma}\right)}. \tag{72}$$

Note that, in contrast to scalar problems, $\mu_{\mathrm{LFA}}(S_{j,1})$ explicitly depends on the mesh size h_j. Due to differential operators of different orders appearing in K_j, this is usually the case when deriving smoothing factors for systems of equations. In Fig. 13, we plot the smoothing factor $\mu_{\mathrm{LFA}}(S_{j,1})$ of (72) as a function of the mesh size h_j and the regularization parameter σ for the range $(\sigma, h_j) \in [1.0, 1.0_{-6}] \times [2^{-2}, 2^{-12}]$. The large light gray area indicates that for these combinations of σ and h_j the smoothing factor $\mu_{\mathrm{LFA}}(S_{j,1})$ is close to that of $\mu_{\mathrm{LFA}}(S_{L_h})$ for the constraint smoother. On the other hand, dark gray shading indicates that $\mu_{\mathrm{LFA}}(S_{j,1}) > 1$, which is obtained for combinations of small σ and large h_j. A smoothing factor $\mu_{\mathrm{LFA}}(S_{j,1}) > 1$ is an indication for degradation of convergence or even divergence of the multigrid process. For (72) it holds that $\mu_{\mathrm{LFA}}(S_{j,1}) = 1$ for $\sigma = h_j^4/4$ and thus we obtain the condition

$$\mu_{\mathrm{LFA}}(S_j) < 1 \quad \text{if } \sigma > \frac{h_j^4}{4}. \tag{73}$$

In Section 7, we further elaborated on (73) with numerical experiments and an ensuing discussion of the consequences of (73) for the robustness of our multigrid solver.

References

[1] M. Bergounioux, K. Ito, K. Kunisch, Primal–dual strategy for constrained optimal control problems, SIAM Journal on Control and Optimization 37 (1999) 1176–1194.
[2] M. Bergounioux, M. Haddou, M. Hintermüller, K. Kunisch, A comparison of a Moreau–Yosida-based active set strategy and interior point methods for constrained optimal control problems, SIAM Journal on Optimization 11 (2000) 495–521.
[3] M. Hintermüller, K. Ito, K. Kunisch, The primal–dual active set strategy as a semismooth Newton method, SIAM Journal on Optimization 13 (2003) 865–888.
[4] S. Mehrotra, On the implementation of a primal–dual interior point method, SIAM Journal on Optimization 2 (1992) 575–601.
[5] S.J. Wright, Primal–Dual Interior Point Methods, SIAM, Philadelphia, 1997.
[6] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numerica 14 (2005) 1–137.

[7] O. Ghattas, G. Biros, Parallel Lagrange–Newton–Krylov–Schur methods for PDE-constrained optimization. Part I: the Krylov–Schur solver, SIAM Journal on Scientific Computing 27 (2) (2005) 687–713.
[8] W. Hackbusch, Fast solution of elliptic control problems, Journal of Optimization Theory and Application 31 (4) (1980) 565–581.
[9] T. Dreyer, B. Maar, V. Schulz, Multigrid optimization in applications, Journal of Computational and Applied Mathematics 120 (2000) 67–84.
[10] A. Borzi, K. Kunisch, A multigrid scheme for elliptic constrained optimal control problems, Computational Optimization and Applications 31 (2005) 309–333.
[11] A. Borzi, G. Borzi, An algebraic multigrid method for a class of elliptic differential systems, SIAM Journal on Scientific Computing 25 (2003) 302–323.
[12] C. Keller, N.I. Gould, A.J. Wathen, Constraint preconditioning for indefinite linear systems, SIAM Journal on Matrix Analysis and Applications 21 (4) (2000) 1300–1317.
[13] J.-L. Lions, Optimal Control of Systems Governed by Partial Differential Equations, Springer, Berlin, 1971.
[14] A.V. Fursikov, Optimal Control of Distributed Systems, American Math. Soc., Providence, RI, 2000.
[15] P.G. Ciarlet, J.L. Lions, Handbook of Numerical Analysis, Volume II: Finite Element Methods (Part 1), North-Holland, 2003.
[16] R. Falk, Approximation of a class of optimal control problems with order of convergence estimates, Journal of Mathematical Analysis and Applications 44 (1973) 28–47.
[17] C. Meyer, A. Rösch, Superconvergence properties of optimal control problems, SIAM Journal on Control and Optimization 43 (2004) 970–985.
[18] M. Hinze, A variational discretization concept in control constrained optimization: the linear–quadratic case, Computational Optimization and Applications 30 (2005) 45–61.
[19] R.A. Raviart, J.M. Thomas, A mixed finite element method for 2nd order elliptic problems, in: Mathematical Aspects of Finite Element Methods, Proc. Conf., Consiglio Naz. delle Ricerche (C.N.R.), Rome, 1975, in: Lecture Notes in Math, vol. 606, Springer, Berlin, 1977, pp. 292–315.
[20] T. Arbogast, C.N. Dawson, P.T. Keenan, M.F. Wheeler, I. Yotov, Enhanced cell-centered finite differences for elliptic equations on general geometry, SIAM Journal of Scientific Computing 19 (1998) 404–425.
[21] J. Nocedal, S.J. Wright, Numerical Optimization, in: Springer Series in Operations Research, Springer, New York, Berlin, Heidelberg, 1999.
[22] M. Engel, A Multigrid Method for the Efficient Numerical Solution of Optimization Problems Constrained by Partial Differential Equations, Dissertation, Institute for Numerical Simulation, University of Bonn, Germany, 2009.
[23] A. Brandt, Rigorous quantitative analysis of multigrid, I: constant coefficients two-level cycle with L_2-norm, SIAM Journal on Numerical Analysis 31 (1994) 1695–1730.
[24] W. Hackbusch, Multi-Grid Methods and Applications, in: Springer Series in Computational Mathematics, Springer, Berlin, Heidelberg, 1985.
[25] U. Trottenberg, C. Oosterlee, A. Schüller, Multigrid, Academic Press, London, 2001.
[26] P. Wesseling, An Introduction to Multigrid Methods, John Wiley & Sons, Chichester, 1992.