A FAST SOLVER FOR AN $\mathcal{H}_1$ REGULARIZED PDE-CONSTRAINED OPTIMIZATION PROBLEM

ANDREW T. BARKER*, TYRONE REES†, AND MARTIN STOLL‡

Abstract. In this paper we consider a PDE-constrained optimization problem where an $\mathcal{H}_1$ regularization control term is introduced. We address both time-independent and time-dependent versions with both distributed and boundary control. We introduce bound constraints on the state, and show how these can be handled by a Moreau-Yosida penalty function. We propose Krylov solvers and Schur complement preconditioning strategies for the different problems and illustrate their performance with numerical examples.

1. Introduction. In recent years the development of numerical methods for optimal control problems with constraints given by partial differential equations (PDEs) has seen many contributions: see [59, 36, 34] and the references mentioned therein. The canonical PDE-constrained optimization problem takes a given desired state $\bar{y}$ and finds a state y and a control u to minimize the functional

$$\|y - \bar{y}\|_{\mathcal{Y}}^2 + \frac{\beta}{2} R(u) \tag{1.1}$$

subject to the constraints

$$\begin{aligned} \mathcal{A}y &= u \\ u_a \leq u &\leq u_b \\ y_a \leq y &\leq y_b \end{aligned}$$

where $\mathcal{Y}$ is some norm and $R(u)$ is a regularization functional, both of which are free to be chosen and often depend on the underlying application. Here $\mathcal{A}$ denotes a PDE with appropriate boundary conditions and β denotes a regularization parameter, which determines how much weight to give to the regularization term.

The simplest choice of $R(u)$ is $\|u\|_{L_2(\Omega)}^2$, where Ω denotes the domain on which the PDE is posed. This case has been well-studied in the literature, both from a theoretical and algorithmic perspective. Of increasing interest, driven by the requirements of real-world problems, has been the inclusion of alternative regularization terms. There has been much interest recently in regularization using L_1 norms. For the standard L_1 case see, e.g., the recent articles [10, 63]. Total variation, $R(u) = \|\nabla u\|_{L_1(\Omega)}$, has also generated much interest recently – see e.g. [51, 12] and the references therein. The L_1 norms have the benefit that they allow discontinuous controls, which can be important in certain applications. Another field where the standard L_2 regularization may not be the most appropriate is flow control – see, e.g., Gunzburger [23, Chapter 4].

For some applications we would like a very smooth control – for this reason the $\mathcal{H}_1$ semi-norm, $R(u) = \|\nabla u\|_{L_2(\Omega)}^2$, has long been studied in the context of parameter-estimation problems [8, 38, 64], image-deblurring [11, 14, 40], image reconstruction

*Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg Germany (barker@mpi-magdeburg.mpg.de)

†Numerical Analysis Group, Scientific Computing Department, Rutherford Appleton Laboratory, Chilton, Didcot, Oxfordshire, OX11 0QX, United Kingdom (tyrone.rees@stfc.ac.uk)

‡Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg Germany (stollm@mpi-magdeburg.mpg.de)

[41], and flow control [15, 28], for example. Recently van den Doel, Ascher and Haber [16] argued that this norm can be a superior choice to its L_1 based cousin, total variation, for problems with very noisy data due to the smooth nature of controls which arise. The test problems in PDE constrained optimization by Haber and Hanson [26], which were designed to get academics solving problems more inline with the needs of the real-world, suggest a regularization functional of the form $R(u) = \|u\|^2_{L_2(\Omega)} + \alpha\|\nabla u\|^2_{L_2(\Omega)}$ for some parameter α. Indeed, this form of regularization is again commonly used in the ill-posed and inverse problem communities.

At the heart of many techniques for solving the optimization problem, whether it is a linear problem or the linearization of some non-linear problem, lies the solution of a linear system. These systems are very often so-called saddle point matrices [2, 19], which have the form

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix}. \tag{1.2}$$

where A represents the misfit and regularization terms in (1.1) and B represents the PDE constraint. The systems we consider in this paper have A which is symmetric positive semi-definite. Such saddle point matrices are invertible if B has full rank and $\ker(A) \cap \ker(B) = \{\mathbf{0}\}$: we will assume this condition holds for the remainder of this paper. We are then left with the challenge of efficiently solving linear systems of the form (1.2).

Direct solvers based on factorizations [18] can be effective in some cases, but for many large and, in particular, three-dimensional problems these are no longer sufficient. In such cases we turn to iterative Krylov subspace methods, which can deal with these large and sparse systems efficiently provided that they are used with a preconditioner that enhances the convergence behavior, ideally independent of problem-dependent parameters, such as the mesh-size or the regularization parameter. For a general overview of preconditioners we refer to [52, 24], and in the particular case of saddle point problems see [2, 19, 65].

A number of preconditioners which are robust with respect to regularization parameters and mesh-parameters have recently been developed for PDE-constrained optimization [55, 45, 44, 58, 39, 17, 29], although these methods are tailored for an optimization problem with $R(u) = \|u\|_{L_2(\Omega)}$ and heavily rely on the corresponding presence of a mass matrix in the A block of (1.2). Benzi, Haber and Taralli [3] consider a block preconditioner with of $R(u)$ given by (a variant of) the $\mathcal{H}_1-$norm, but their approach is general enough to work with most regularization and the form of the regularization is not exploited in the method. To the authors' knowledge there has been no other attempts to apply block preconditioners – which have proved so successful with L_2 regularization – in the case of other choices of $R(u)$. We address this issue here by considering a cost-functional where $R(u) = \|u\|_{L_2} + \|\nabla u\|_{L_2}$, and we present preconditioners that show robustness with respect to the regularization parameter for this problem, which is more challenging from a linear algebra perspective.

The paper is structured as follows. We begin in Section 2 by stating the optimal control problem in the time-dependent and time-independent cases with both distributed and boundary control. We illustrate how the discretized first order conditions can be obtained from a so-called discretize-then-optimize approach. In Section 3 we describe how the state constraints can be handled using a Moreau-Yosida penalty approach and show how to incorporated this into possible preconditioning strategies. In Section 4 we discuss the choice of possible Krylov solvers and introduce preconditioning strategies for both the time-dependent and time-independent control problem,

with an emphasis on how to handle the $\mathcal{H}_1$ regularization term. Our numerical results shown in Section 5 illustrate the efficiency of our approach.

2. Problem setup and discretization.

2.1. Time-independent control. First we consider the time-independent optimal control problem, where the following objective function should be minimized:

$$\mathcal{J}_1(y,u) = \frac{1}{2}\left\|y-\bar{y}\right\|^2_{L_2(\Omega_1)} + \frac{\beta}{2}\left\|u\right\|^2_{\mathcal{H}_1(\Omega_2)} \tag{2.1}$$

$$= \frac{1}{2}\left\|y-\bar{y}\right\|^2_{L_2(\Omega_1)} + \frac{\beta}{2}\left\|u\right\|^2_{L_2(\Omega_2)} + \frac{\beta}{2}\left\|\nabla u\right\|^2_{L_2(\Omega_2)}, \tag{2.2}$$

where both Ω_1 and Ω_2 are subdomains of $\Omega \in \mathbb{R}^d$ with $d = 2, 3$. The constraint is given by the following elliptic PDE

$$-\triangle y = \begin{cases} u & \text{in } \Omega_2 \\ 0 & \text{in } \Omega \backslash \Omega_2 \end{cases} \tag{2.3}$$

together with Dirichlet boundary conditions, $y = g$ on $\partial\Omega$. We refer to y as the state and u as the corresponding control, which is used to drive the state variable as close as possible to the desired state (or observations) $\bar{y}$. The above problem is the distributed control problem, as u defines the forcing of the PDE over the interior subdomain Ω_2. Another important case is given by the boundary control problem, where $\Omega_2 = \partial\Omega$ together with the PDE constraint

$$-\triangle y = f \text{ in } \Omega \tag{2.4}$$

$$\frac{\partial y}{\partial n} = u \text{ on } \partial\Omega \tag{2.5}$$

where f represents a fixed forcing term.

Problems of this type frequently appear in practical situations [34, 47, 13, 48]. Additionally, many practical applications require the introduction of so-called box constraints on the control and/or the state, motivated by physical characteristics of the application of interest. Typical bounds would be

$$u_a \leq u \leq u_b$$

for the control and

$$y_a \leq y \leq y_b$$

for the state. The numerical treatment of these constraints is by now well established [31, 4, 5] but nevertheless represents a computational challenge, in particular for the state constraints [9].

There are two approaches that can be taken to solve such PDE-constrained optimization problems numerically: discretize-then-optimize, where the infinite-dimensional problem is discretized and then a finite-dimensional optimization problem is solved; and optimize-then-discretize, where we optimize the infinite dimensional problem first, and then discretize the first order optimality conditions accordingly (see [34]). Current research follows the paradigm that we should use discretization schemes for which both approaches coincide [33].

We will follow the discretize-then-optimize approach and discretize the PDE and the objective function using Q1 finite elements [19, 57]; we employ the deal.II [1] finite element package for our numerical experiments.

We derive the discrete optimality system for the cost functional (2.1) with the PDE constraint (2.3), together with homogenous Dirichlet boundary conditions for ease of exposition – the extension to other boundary conditions proceeds similarly. Let $\phi_1, \dots \phi_n$ be a finite element basis for the interior of Ω, and suppose we extend this by $\phi_{n+1}, \dots, \phi_{n+\partial n}$ to include the boundary. Let $Y_0^h = \langle \phi_1 \dots \phi_n \rangle$, $U^h = \langle \phi_1 \dots \phi_n, \phi_{n+1}, \phi_{n+\partial n} \rangle$. Furthermore, let $Y_{\Omega_1} := \langle \hat{\phi}_1 \ \dots \hat{\phi}_{\hat{m}} \rangle$ and $U_{\Omega_2} := \langle \bar{\phi}_1 \ \dots \bar{\phi}_{\bar{m}} \rangle$ denote the subsets of U^h with support on Ω_1 and Ω_2 respectively.

The finite dimensional analogue to (2.1), (2.3) is to find $y_h \in Y_0^h \subset \mathcal{H}_0^1(\Omega)$ and $u_h \in U^h \subset \mathcal{H}_1(\Omega)$ which satisfy

$$\min_{y_h \in Y_{\Omega_1}, u_h \in U_{\Omega_2}} \frac{1}{2} ||y_h - I_h \bar{y}||^2_{L_2(\Omega_1)} + \frac{\beta}{2} ||u_h||^2_{\mathcal{H}_1(\Omega_2)},$$

$$\text{s.t.} \qquad \int_\Omega \nabla y_h \cdot \nabla v_h = \int_\Omega u_h v_h, \ \forall v_h \in Y_0^h,$$

where $I_h : \mathcal{H}_1(\Omega) \to Y^h$ is the finite element projection. We can write the optimization problem in terms of matrices as

$$\min_{\mathbf{y},\mathbf{u}} \frac{1}{2}(\mathbf{y} - \bar{\mathbf{y}})^T \bar{M}_y (\mathbf{y} - \bar{\mathbf{y}}) + \frac{\beta}{2} \mathbf{u}^T \bar{M}_u \mathbf{u} + \frac{\beta}{2} \mathbf{u}^T \bar{K}_u \mathbf{u} \tag{2.6}$$

$$\text{s.t. } \bar{K}\mathbf{y} = \bar{M}\mathbf{u}, \tag{2.7}$$

where

$$(\bar{M}_y)_{i,j} = \int_\Omega \hat{\phi}_i \hat{\phi}_j, \ i,j = 1, \dots, \hat{m}, \qquad (\bar{K}_u)_{i,j} = \int_\Omega \nabla \bar{\phi}_i \cdot \nabla \bar{\phi}_j, \ i,j = 1, \dots, \bar{m},$$

$$(\bar{M}_u)_{i,j} = \int_\Omega \bar{\phi}_i \bar{\phi}_j, \ i,j = 1, \dots, \bar{m}, \qquad (\bar{K})_{i,j} = \int_\Omega \nabla \phi_i \cdot \nabla \phi_j, i,j = 1, \dots, n,$$

$$(\bar{M})_{i,j} = \int_\Omega \phi_i \bar{\phi}_j, i = 1, \dots, n, \ j = 1, \dots, \bar{m}$$

In our finite element implementation it is more convenient to write matrices on the whole space, and we can write

$$M_y = \begin{bmatrix} \bar{M}_y & 0 \\ 0 & 0 \end{bmatrix}, \ M_u = \begin{bmatrix} \bar{M}_u & 0 \\ 0 & 0 \end{bmatrix}, \ K_u = \begin{bmatrix} \bar{K} & 0 \\ 0 & 0 \end{bmatrix}$$

$$M = \begin{bmatrix} \bar{M} & 0 \\ 0 & 0 \end{bmatrix}, \text{ and } K = \begin{bmatrix} \bar{K} & 0 \\ 0 & I \end{bmatrix}. \tag{2.8}$$

Note that here we have split the $(n+\partial n) \times (n+\partial n)$ matrices into 2×2 blocks, where the (1,1) and (2,2) blocks describe the terms which are present or not, respectively. Note that the size of the blocks are not uniform: in M and K they correspond to interior and boundary nodes; if $\Omega_1 = \Omega$, then the size of the zero block would also correspond to the boundary nodes as y is fixed by the boundary condition of the PDE; if $\Omega_2 = \Omega$, then the zero blocks in M_u and K_u will vanish.

In the distributed control case the first order optimality conditions lead to the following saddle point system:

$$\begin{bmatrix} M_y & 0 & -K^T \\ 0 & \beta M_u + \beta K_u & M \\ -K & M & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} M_y \bar{\mathbf{y}} \\ 0 \\ \mathbf{d} \end{bmatrix}. \tag{2.9}$$

Note that the addition of an $\mathcal{H}_1$ norm in the regularization leads to an optimality system with substantially different properties compared to the L_2 case; in particular, if $\boldsymbol{p} = 0$ on the boundary, we do not necessarily have that $\mathbf{u} = 0$ on the boundary here, which is known to be true if we have just L_2 regularization (see [59, 50]).

For the incorporation of the boundary conditions into the saddle point system we argued in [50] that it is not convenient to work with matrices on the interior as most finite element packages will assemble the matrices on the whole of the domain including the boundary. It can be seen, assuming all matrices are of the form (2.8), that if the current approximations $\mathbf{y}$ and $\boldsymbol{p}$ contain zeros corresponding to the zero blocks in (2.9), these zeros are maintained throughout any Krylov solver. This allows the undisturbed maintenance of the boundary conditions for the state and adjoint state. With adjustment of the right hand sides, which is a cheap operation compared to the elimination of degrees of freedom from a matrix, this means that one can use matrices that are readily available from most finite element packages to solve the saddle point system with a Krylov subspace solver. We refer the interested reader to [50] for more details. Note that for non-homogeneous boundary conditions all that changes is that a constant vector appears on the right hand side of (2.7).

The boundary control problem can be treated similarly. Here we get

$$\mathcal{J}_1(\mathbf{y}, \mathbf{u}) = \frac{1}{2} (\mathbf{y} - \bar{\mathbf{y}})^T M_y (\mathbf{y} - \bar{\mathbf{y}}) + \frac{\beta}{2} \mathbf{u}^T M_{u,b} \mathbf{u} + \frac{\beta}{2} \mathbf{u}^T K_{u,b} \mathbf{u} \tag{2.10}$$

together with

$$K\mathbf{y} = N\mathbf{u} + \mathbf{f}. \tag{2.11}$$

Here: $M_{u,b}$ and $K_{u,b}$ are the boundary mass matrix and Laplacian, respectively; the vector $\mathbf{f}$ represents the discretized forcing term, which for simplicity we take to be zero for the remainder of the paper; the matrix N connects interior and boundary basis functions, in particular

$$(N)_{ij} = \int_{\partial\Omega} \phi_i \mathrm{tr}(\phi_j)\, ds.$$

where tr is the trace operator. We obtain the following first order optimality system

$$\begin{bmatrix} M_y & 0 & -K^T \\ 0 & \beta M_{u,b} + \beta K_{u,b} & N^T \\ -K & N & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} M_y \bar{\mathbf{y}} \\ 0 \\ \mathbf{d} \end{bmatrix}. \tag{2.12}$$

2.2. Time-dependent problem. We now present a time-dependent version, which is of wide practical interest. The objective function is now given by

$$\mathcal{J}_2(y, u) = \frac{1}{2} \int_0^T \int_{\Omega_1} (y - \bar{y})^2 dx dt + \frac{\beta}{2} \int_0^T \int_{\Omega_2} u^2 dx dt + \frac{\beta}{2} \int_0^T \int_{\Omega_2} (\nabla u)^2 dx dt, \tag{2.13}$$

where all functions are simply time-dependent versions of their steady counterparts presented above. The constraint is given by the following time-dependent parabolic PDE

$$y_t - \triangle y = u$$

for the distributed control problem with Dirichlet boundary conditions, i.e. $y(\mathbf{x}, t) = g(\mathbf{x}, \mathbf{t})$ on $\partial\Omega$ for some prescribed function g. In case of a boundary control problem, we consider the following PDE constraint

$$y_t - \triangle y = f \tag{2.14}$$

$$\frac{\partial y}{\partial n} = u \text{ on } \partial\Omega. \tag{2.15}$$

For the discretization of the time-dependent objective function we use the trapezoidal rule for the time integral and finite elements in space to give

$$\mathcal{J}_2(\mathbf{y}, \mathbf{u}) = \frac{1}{2}\left(\mathbf{y} - \bar{\mathbf{y}}\right)^T \mathcal{M}_y \left(\mathbf{y} - \bar{\mathbf{y}}\right) + \frac{\beta}{2}\mathbf{u}^T \mathcal{M}_u \mathbf{u} + \frac{\beta}{2}\mathbf{u}^T \mathcal{K}_u \mathbf{u} \tag{2.16}$$

where

$$\mathcal{M}_y = \text{blkdiag}(1/2M_y, M_y, \ldots, M_y, 1/2M_y),$$

$$\mathcal{M}_u = \text{blkdiag}(1/2M_u, M_u, \ldots, M_u, 1/2M_u)$$

and

$$\mathcal{K}_u = \text{blkdiag}(1/2K_u, K_u, \ldots, K_u, 1/2K_u),$$

which are simply block-variants of the previously defined matrices over the domains Ω_1 and Ω_2. Note that in the time-dependent case we abuse the notation $\mathbf{y}$, $\mathbf{u}$ defined earlier, i.e., $\mathbf{y} = \left[\mathbf{y}_1^T, \mathbf{y}_2^T, \ldots, \mathbf{y}_{N_T}^T\right]^T$, etc. We believe it will be clear from the context which of the two we are currently considering. Using this notation and a backward Euler scheme, we can write down a one-shot discretization of the time-dependent PDE as follows

$$\begin{bmatrix} L & & & \\ -M & L & & \\ & \ddots & \ddots & \\ & & -M & L \end{bmatrix} \mathbf{y} - \tau\mathcal{M}\mathbf{u} = \mathbf{d} \tag{2.17}$$

with $L = M + \tau K$ and $\mathbf{d}$ representing the boundary conditions for the heat equation. For more details see [56, 3, 17].

Again, we form the Lagrangian and write down the first order conditions in a linear system,

$$\begin{bmatrix} \tau\mathcal{M}_y & 0 & -\mathcal{K}^T \\ 0 & \tau\beta(\mathcal{M}_u + \mathcal{K}_u) & \tau\mathcal{M} \\ -\mathcal{K} & \tau\mathcal{M} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} \tau\mathcal{M}_y\bar{\mathbf{y}} \\ 0 \\ \mathbf{d} \end{bmatrix}, \tag{2.18}$$

in the case of the distributed control problem, and

$$\begin{bmatrix} \tau\mathcal{M}_y & 0 & -\mathcal{K}^T \\ 0 & \tau\beta(\mathcal{M}_{u,b} + \mathcal{K}_{u,b}) & \tau\mathcal{N}^T \\ -\mathcal{K} & \tau\mathcal{N} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} M_y\bar{\mathbf{y}} \\ 0 \\ \mathbf{d} \end{bmatrix} \tag{2.19}$$

for boundary control.

3. Handling the state constraints. Box constraints both for the control $\mathbf{u}$ and the state $\mathbf{y}$ can be dealt with efficiently using a penalty term. For the case of constraints on both the control and the state of an optimal control problem the Moreau-Yosida penalty function has proven to be a viable tool: see [35, 29, 44] and the references mentioned therein. We describe this modification for time-independent distributed control, but the other cases are similar. The modified objective function becomes

$$\mathcal{J}_{MY}(y,u) = \mathcal{J}_1(y,u) + \frac{1}{2\varepsilon}\left\|\max\left\{0, y - y_b\right\}\right\|^2 + \frac{1}{2\varepsilon}\left\|\min\left\{0, y - y_a\right\}\right\|^2 \tag{3.1}$$

for the state constrained case and similarly for control constraints. In accordance with [29], we can employ a semi-smooth Newton scheme that leads to the following linear system

$$\begin{bmatrix} M_y + \varepsilon^{-1} G_{\mathcal{A}} M_y G_{\mathcal{A}} & 0 & -K^T \\ 0 & \beta M_u + \beta K_u & M \\ -K & M & 0 \end{bmatrix} \begin{bmatrix} \delta_{\mathbf{y}} \\ \delta_{\mathbf{u}} \\ \delta_{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} M_y \bar{\mathbf{y}} + \varepsilon^{-1}\left(G_{\mathcal{A}_+} M_y G_{\mathcal{A}_+} y_b + G_{\mathcal{A}_-} M_y G_{\mathcal{A}_-} y_a\right) \\ 0 \\ \mathbf{d} \end{bmatrix} \tag{3.2}$$

where we define the active sets as $\mathcal{A}_+ = \{i : \mathbf{y}_i > (y_b)_i\}$, and $\mathcal{A}_- \{i : \mathbf{y}_i < (y_a)_i\}$, and $\mathcal{A} = \mathcal{A}_+ \cup \mathcal{A}_-$; the matrices G are diagonal matrix variants of the characteristic function for the corresponding sets, i.e.,

$$(G_{\mathcal{A}})_{ii} = \begin{cases} 1 & \text{for } i \in \mathcal{A} \\ 0 & \text{otherwise.} \end{cases}$$

Our focus is on the efficient solution of the linear systems (3.2), which are of saddle point type. Note that the active sets defined above within an iterative process such as the semi-smooth Newton scheme are computed based on the state at the previous iteration, but for simplicity we neglect the iteration index. For more details of semi-smooth Newton methods we refer to [36, 60, 34]; there is also recent theory introducing path-following approaches for the penalty parameter ε [32].

4. Preconditioning.

4.1. Choice of Krylov solver and Schur complement preconditioning. As mentioned in the introduction, the linear systems that arise from PDE-constrained optimization are very often too large for direct solvers to be effective, and for scalable and efficient solution of these linear systems the combination of a state-of-the-art solver with an efficient preconditioning technique is crucial. In this section we derive preconditioners for each of the problems presented earlier, but first mention the choice of the iterative scheme. Krylov solvers are for many applications the method of choice [54], as they are cheap to apply — at each step they only require a matrix vector product, the evaluation of the preconditioners, and the evaluation of inner products. These methods build up a low-dimensional subspace that can be used to approximate the solution to the linear system.

There are a variety of Krylov subspace methods, and the most effective to use depends on the properties of the linear system. For symmetric and positive definite matrices the conjugate gradient (CG) method of Hestenes and Stiefel [30] – with a

symmetric and positive definite preconditioner – is the method of choice. For symmetric and indefinite problems, such as the ones we are dealing with here, the minimal residual method (MINRES) introduced by Paige and Saunders [43], as well as modified variations of the CG method [7], are often effective. The only requirement for MINRES to be applicable to a symmetric system is that a symmetric and positive definite preconditioner is used.

If an indefinite or block-triangular preconditioner is used we must use a nonsymmetric Krylov subspace solver, even when the underlying matrix is symmetric. There is in general no ideal choice and we mention GMRES [53], BICG [20], BICGSTAB [61] or QMR [21] as suitable candidates. It is also not always clear what guides the convergence of these methods but as in the symmetric case a small number of eigenvalue clusters typically corresponds to a fast convergence of the iterative solver. Here we focus on the development of effective preconditioners and we will focus less on the choice of linear solver—for most systems MINRES will be our method of choice but if we choose a nonsymmetric preconditioner we will use BICG.

We now want to describe preconditioners which have proven to be efficient for solving systems of the form (1.2) in combination with MINRES. As the system matrix is indefinite it is not immediately obvious that a good preconditioner can be found that is symmetric and positive definite. In the notation of the generic saddle point problem (1.2), Murphy *et al.* [42] show that the preconditioned matrix $\mathcal{P}^{-1}\mathcal{A}$ with $\mathcal{P} = \text{blkdiag}(A, S)$, where $S := BA^{-1}B^T$ is the Schur-complement of $\mathcal{A}$, has three eigenvalues. This results in the termination of MINRES after at most three steps. Naturally, this $\mathcal{P}$ is too expensive for any realistic problem but it illustrates that if we can find good approximations to both the $(1,1)$-block and the Schur-complement of $\mathcal{A}$, then the method will converge in a small number of steps. In some cases in what follows we will use a nonsymmetric approximation to S, and since then we have already broken symmetry, we can consider the block-triangular preconditioner

$$\mathcal{P}_2 = \begin{pmatrix} A & 0 \\ B & S \end{pmatrix}.$$

This reduces the issue of approximating the solution of the linear system to finding good approximations to the $(1,1)$-block and the Schur-complement of $\mathcal{A}$. Parts of the $(1,1)$-block in most of the cases presented here consists of lumped mass matrices and can simply be inverted. If the mass matrices are consistent we can use the Chebyshev semi-iteration [62] and if the $(1,1)$-block is only semi-definite we can add a small perturbation to make it positive definite so the above applies, i.e., we replace the zero blocks in A by blocks of the form ηI with η a small parameter greater than zero. Note that this technique can also be used for an approximation of the Schur-complement in case the $(1,1)$-block is semi-definite [3, 56]. For the rest of the paper our preconditioners will rely on blocks $\hat{A}$ and $\hat{S}$ where $\hat{A}$ approximates the $(1,1)$-block and $\hat{S}$ the Schur-complement. We discuss appropriate approaches for the approximation of the Schur-complement next.

4.2. Time-independent problem.

No state constraints. We are interested in finding a good preconditioner for the matrix

$$\begin{bmatrix} M_y & 0 & -K^T \\ 0 & \beta M_u + \beta K_u & M \\ -K & M & 0 \end{bmatrix}$$

from (2.9). It is fairly straightforward to deal with the blocks M_y and $\beta M_u + \beta K_u$ efficiently. In particular, the mass matrix M_y can be approximated as outlined in the previous section, by the Chebyshev semi-iteration in the case of a consistent mass matrix or simply inverted whenever it is lumped. The inverse of $\beta M_u + \beta K_u$ can be efficiently approximated using (algebraic) multigrid.

The performance of our preconditioner therefore depends on having a good approximation of the Schur-complement

$$S = KM_y^{-1}K^T + M(\beta M_u + \beta K_u)^{-1}M.$$

One possible approximation would be

$$\hat{S}_1 = KM_y^{-1}K^T$$

(see [49]), which neglects the second term in the Schur-complement. This typically results in good convergence properties for relatively large β, but performance deteriorates as β approaches zero. Another approach [46], that in some cases can overcome or weaken this dependence on the regularization parameter, is given by

$$\hat{S}_2 = (K + \hat{M})M_y^{-1}(K^T + \hat{M}^T), \tag{4.1}$$

where the matrix $\hat{M}$ is chosen to approximate the second term in the Schur-complement well. In more detail, we would ideally construct $\hat{M}$ such that

$$\hat{M}M_y^{-1}\hat{M}^T = M(\beta M_u + \beta K_u)^{-1}M, \tag{4.2}$$

which is the case for $\hat{M} = M(\beta M_u + \beta K_u)^{-1/2}M_y^{1/2}$. Note that with this choice we cannot easily form and invert $(K + \hat{M})$. We instead choose the diagonal $\mathrm{diag}(K_u)$ as an approximation for K_u. Note that the approximation of K_u by its diagonal is, in the case of a forward Poisson problem, not ideal as no mesh-independence can be expected. Nevertheless, the inverse of $(K + \hat{M})$ needs to be approximated cheaply and, as we are using lumped mass matrices, we now get

$$\hat{M} = M(\beta M_u + \beta D_K)^{-1/2}M_y^{1/2},$$

where $D_K = \mathrm{diag}(K_u)$. This allows us to form $K + \hat{M}$, whose inverse in turn can be approximated using an algebraic or geometric multigrid preconditioner.

A nonsymmetric Schur complement approximation. From the previous section it is clear that the ideal candidate $\hat{M} = ML_u^{-1/2}M_y^{1/2}$ in (4.2) would yield

$$\hat{M}M_y^{-1}\hat{M}^T = ML_u^{-1/2}M_y^{1/2}M_y^{-1}M_y^{1/2}L_u^{-1/2}M = ML_u^{-1}M,$$

using $L_u = \beta M_u + \beta K_u$. Unfortunately, from a numerical viewpoint it is not possible to work with the matrix $K + ML_u^{-1/2}M^{1/2}$ in (4.1) as even if we can rewrite this more efficiently we would have to compute the square root of large matrices. We therefore consider a nonsymmetric approximation that would also require the use of a nonsymmetric Krylov subspace solver. In detail, we use

$$\hat{S}_3 = (K + \hat{M}_1)M_y^{-1}(K + \hat{M}_2)^T, \tag{4.3}$$

where we choose $\hat{M}_1 = ML_u^{-1}M$ and $\hat{M}_2 = M_y$. This leads to

$$\hat{M}_1M_y^{-1}\hat{M}_2^T = ML_u^{-1}M,$$

that is, (4.2) is satisfied and we may expect $\hat{S}_3$ to work well for a wide range of β. On the other hand, we now have a complicated structure within $K + ML_u^{-1}M$ and we have to switch to a nonsymmetric outer solver. This complicated structure can be dealt with by noting the equivalence between the following two systems

$$\left(K + ML_u^{-1}M\right) x = b, \quad \begin{bmatrix} K & M \\ M & -L_u \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix}. \tag{4.4}$$

We now use a preconditioned Uzawa-type method to approximately solve the second system, which has the advantage of never needing the inverse of L_u, which every iterative solver used for the first equation would require. In this case we have already broken symmetry so we can take advantage of the block-triangular preconditioner $\mathcal{P}_2$.

State constraints. We are now interested in finding a good preconditioner for the matrix coming from the state constrained problem treated with a Moreau-Yosida penalty term,

$$\begin{bmatrix} L & 0 & -K^T \\ 0 & \beta M_u + \beta K_u & M \\ -K & M & 0 \end{bmatrix},$$

where $L = M_y + \varepsilon^{-1} G_{\mathcal{A}} M_y G_{\mathcal{A}}$. Due to the diagonal nature of the mass matrices the matrix L is simply a diagonal matrix and can be treated trivially in the preconditioner. For the block $\beta M_u + \beta K_u$ we can again use a multigrid process to approximate the inverse. Now we need an efficient way to approximate the Schur-complement

$$S = KL^{-1}K + M(\beta M_u + \beta K_u)^{-1}M. \tag{4.5}$$

We want to employ the technique used for the case without state constraints. We start by looking for an approximation of the form

$$\hat{S} = (K + \hat{M})L^{-1}(K + \hat{M})^T, \tag{4.6}$$

where we have to determine the matrix $\hat{M}$ in such a way that the second term in S is accounted for. For this we want

$$\hat{M}L^{-1}\hat{M}^T \approx M(\beta M_u + \beta K_u)^{-1}M.$$

In order to simplify this process we make the following approximation

$$\beta M_u + \beta K_u \approx \beta M_u + \beta D_K := D_u,$$

where $D_K = \mathrm{diag}(K_u)$ and hence D_u is a diagonal matrix. We now proceed to

$$\hat{M}L^{-1}\hat{M}^T = MD_u^{-1}M \Rightarrow \hat{M} = MD_u^{-1/2}L^{1/2}$$

as all matrices involved are diagonal matrices and hence commute, i.e., $\hat{M} = MD_u^{-1/2}L^{1/2} = L^{1/2}D_u^{-1/2}M$.

Boundary control. In the boundary control problem the saddle point matrix is given by

$$\begin{bmatrix} M_y & 0 & -K^T \\ 0 & \beta M_{u,b} + \beta K_{u,b} & N^T \\ -K & N & 0 \end{bmatrix} \tag{4.7}$$

where the two blocks in the upper-left can be handled by previous techniques. The Schur-complement here is

$$S = KM_y^{-1}K^T + N(\beta M_{u,b} + \beta K_{u,b})^{-1}N^T.$$

We can again approximate the Laplacian by its diagonal to get

$$S \approx KM_y^{-1}K^T + N(\beta M_{u,b} + \beta D_{K,b})^{-1}N^T = KM_y^{-1}K^T + ND_u^{-1}N^T.$$

Once again we proceed by assuming that an approximation of the form

$$\hat{S} = (K + \hat{M})M_y^{-1}(K + \hat{M})^T$$

will give a good approximation to the Schur-complement, with

$$\hat{M}M_y^{-1}\hat{M}^T = ND_u^{-1}N^T. \tag{4.8}$$

Since the mass matrices are lumped, we can assume that all the matrices are diagonal, and we get an expression for the diagonal elements of (4.8) corresponding to boundary degrees of freedom. Note that $ND_u^{-1}N^T$ is a diagonal matrix with non-zero entries only for boundary nodes. We also do not account for the difference in scalings with respect to the mesh parameter h between a boundary mass matrix and a mass matrix on the whole domain. The diagonal elements of $\hat{M}$ can be obtained from

$$m_{ii}\hat{m}_{ii}^2 = \frac{m_{ii}^2}{d_{u,ii}}$$

or equivalently

$$\hat{m}_{ii}^2 = \frac{m_{ii}^3}{d_{u,ii}} \Rightarrow \hat{m}_{ii} = \frac{m_{ii}^{3/2}}{\sqrt{d_{u,ii}}}. \tag{4.9}$$

We already mentioned that the boundary mass matrix scales differently compared to the mass matrix on the whole domain by an order of h. We first consider the case when we only have an L_2-term for the control, i.e. $K_u = 0$, and want to compute $\hat{M}$ such that

$$\hat{M}M_y^{-1}\hat{M} = \beta^{-1}NM_u^{-1}N^T$$

and using the approximations $M_y \approx h^2 I$ and $M_u \approx hI$ we get

$$h^{-2}\hat{M}^2 \approx \hat{M}M_y^{-1}\hat{M} = \beta^{-1}NM_u^{-1}N^T \approx \beta^{-1}h^{-1}NN^T.$$

Since all matrices in the last expression are diagonal (recall N is a rectangular matrix with entries only when boundary degree of freedom is paired with boundary degree of freedom) we get

$$\hat{m}_{ii}^2 = h\beta^{-1}m_{ii}^2 \Rightarrow \hat{m}_{ii} = \sqrt{h\beta^{-1}}m_{ii}.$$

Based on this analysis we proceed by multiplying $\hat{m}_{ii}$ in (4.9) with $\sqrt{h}$ to account for the different orders of the boundary matrices and the matrices over the whole domain to finally get

$$\hat{m}_{ii} = \frac{\sqrt{h}m_{ii}^{3/2}}{\sqrt{d_{u,ii}}}.$$

Note that it is also possible to use the nonsymmetric Schur-complement approach for this case. In the numerical experiments section we only show nonsymmetric Schur-complement results for the time-dependent setup.

4.3. Time-dependent problem. We now extend the previous techniques to the time-dependent case.

No state constraints. Recall that the first order conditions of the time-dependent problem are represented by the following saddle point system

$$\begin{bmatrix} \tau\mathcal{M}_y & 0 & -\mathcal{K}^T \\ 0 & \tau\beta(\mathcal{M}_u + \mathcal{K}_u) & \tau\mathcal{M} \\ -\mathcal{K} & \tau\mathcal{M} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} \tau\mathcal{M}_y\bar{\mathbf{y}} \\ 0 \\ \mathbf{d} \end{bmatrix} \tag{4.10}$$

and assume that $\mathcal{M}_y$ and $\tau\beta(\mathcal{M}_u + \mathcal{K}_u)$ are invertible so we can form the Schur-complement

$$S = \tau^{-1}\mathcal{K}\mathcal{M}_y^{-1}\mathcal{K}^T + \tau\mathcal{M}(\beta\mathcal{M}_u + \beta\mathcal{K}_u)^{-1}\mathcal{M}.$$

For strategies to handle a semi-definite $\mathcal{M}_y$ we refer to [56]. Again we approximate S via

$$\hat{S}_2 = \tau^{-1}(\mathcal{K} + \hat{\mathcal{M}})\mathcal{M}_y^{-1}(\mathcal{K} + \hat{\mathcal{M}})^T,$$

with a not yet specified but symmetric matrix $\hat{\mathcal{M}}$. As we want $\hat{\mathcal{M}}\mathcal{M}_y^{-1}\hat{\mathcal{M}}$ to resemble the second block in the Schur-complement S we obtain

$$\hat{\mathcal{M}}\mathcal{M}_y^{-1}\hat{\mathcal{M}}^T \approx \mathcal{M}(\beta\mathcal{M}_u + \beta\mathcal{K}_u)^{-1}\mathcal{M}.$$

As all matrices are block-diagonal, we want that

$$\tau^{-1}\hat{M}M_y^{-1}\hat{M}^T \approx \tau M(\beta M_u + \beta K_u)^{-1}M.$$

Using again the approximation

$$\beta M + \beta K_u \approx \beta M + \beta D_K := D_u,$$

we get

$$\tau^{-1}\hat{M}M^{-1}\hat{M}^T = \tau M D_u^{-1} M \Rightarrow \hat{M} = \tau M D_u^{-1/2} M_y^{1/2}.$$

In this case we can also use a nonsymmetric Schur complement analogous to the $\hat{S}_3$ we derived in the time-independent case.

State constraints. The system obtained from the state constrained case has a change in the $(1,1)$-block, i.e.,

$$\begin{bmatrix} \tau\mathcal{L} & 0 & -\mathcal{K}^T \\ 0 & \tau\beta(\mathcal{M}_u + \mathcal{K}_u) & \tau\mathcal{M} \\ -\mathcal{K} & \tau\mathcal{M} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} \tau\mathcal{M}_y\bar{\mathbf{y}} \\ 0 \\ d \end{bmatrix} \tag{4.11}$$

where $\mathcal{L} = \text{blkdiag}(L_i)$ with the $L_i = M_y + \varepsilon^{-1}G_{\mathcal{A}^{(i)}}M_yG_{\mathcal{A}^{(i)}}$ and $\mathcal{A}^{(i)}$ the active sets for a grid point in time. Assuming invertibility of $\mathcal{L}$, the Schur-complement now becomes

$$S = \tau^{-1}\mathcal{K}\mathcal{L}^{-1}\mathcal{K}^T + \tau\mathcal{M}(\beta\mathcal{M}_u + \beta\mathcal{K}_u)^{-1}\mathcal{M}.$$

We again want to derive an approximation of the form

$$\hat{S} = \tau^{-1}(\mathcal{K} + \hat{\mathcal{M}})\mathcal{L}^{-1}(\mathcal{K}^T + \hat{\mathcal{M}}^T)$$

that resembles the Schur-complement as closely as possible. As we again want $\tau^{-1}\hat{\mathcal{M}}\mathcal{L}^{-1}\hat{\mathcal{M}}$ to resemble the second block in the Schur-complement S we obtain

$$\tau^{-1}\hat{\mathcal{M}}\mathcal{L}^{-1}\hat{\mathcal{M}}^T \approx \tau\mathcal{M}(\beta\mathcal{M}_u + \beta\mathcal{K}_u)^{-1}\mathcal{M}.$$

As all matrices are block-diagonal we want that for all blocks $(i = 1, \ldots, n)$

$$\tau^{-1}\hat{M}L_i^{-1}\hat{M}^T \approx \tau M(\beta M_u + \beta K_u)^{-1}M.$$

and with the approximation

$$\beta M + \beta K_u \approx \beta M + \beta D_K := D_u,$$

we get

$$\tau^{-1}\hat{M}L_i^{-1}\hat{M}^T = \tau M D_u^{-1} M \Rightarrow \hat{M} = \tau M D_u^{-1/2} L_i^{1/2}.$$

Note that now the blocks of $\mathcal{L}$ are different for each point in time as the active sets will be different for each i. In an efficient implementation this issue has to be addressed as recomputing the preconditioner with each application is not feasible. We have not done this for the results presented in Section 5.

5. Numerical Results. We now want to illustrate how the preconditioners presented above perform when applied to a variety of problems. As mentioned earlier we employ a finite element discretization, here done with the finite element package deal.II [1]. We discretize the state, control and adjoint state variables using **Q1** elements. We stop all computations when the relative pseudo-residual minimized in MINRES or BICG falls below 10^{-4}. For the algebraic multigrid preconditioner we use the Trilinos ML package [22] that implements a smoothed aggregation AMG. Within the algebraic multigrid we used 6 steps of a Chebyshev smoother in combination with the application of two V-cycles. Note that, especially in the time-dependent case with state constraints, our implementation at this point is only a proof-of-concept as we are simply recomputing the preconditioner for every active set. Future research should address the issue of efficiently updating the AMG preconditioner or employing a geometric multigrid method that takes the changes of the active set into account.

For time-dependent problems we show the degrees of freedom only for one grid point in time (i.e. for a single time-step) and we are implicitly solving a linear system of dimension 3 times the number of time-steps (N_t) times the degrees of freedom of the spatial discretization (n). For example, a spatial discretization with 274625 spatial unknowns and 20 time-steps corresponds to an overall linear system of dimension 16 477 500.

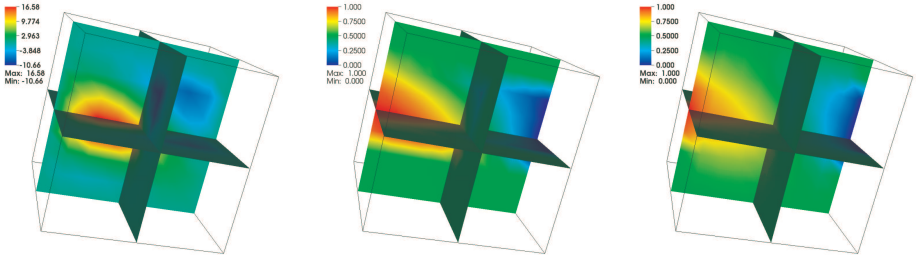
5.1. The time-independent case.

No state constraints. In this section we show numerical results for the time-independent control problem. The desired state is given by

$$\bar{\mathbf{y}} = \begin{cases} \sin(2\pi x_0 x_1 x_2) & \text{if } x_0, x_1 \in [0.2, 0.7] \\ 0.5 & \text{otherwise.} \end{cases}$$

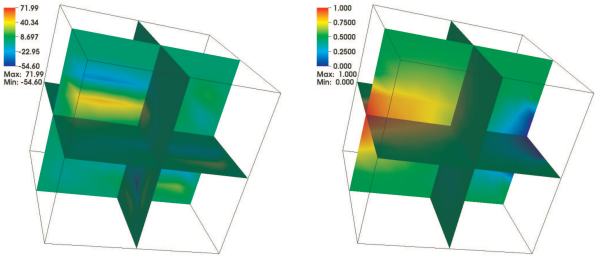
with the Dirichlet condition that $y = 0$ on $\partial\Omega$. The desired state, computed control and computed state for $\beta = 10^{-6}$ are shown in Figure 5.1. We show the results for a variety of β parameters in Table 5.1 and the tolerance 10^{-4} for the pseudo-residual.

We see from the results in Table 5.1 that there is some benign growth in the iteration numbers with respect to the regularization parameter and no dependence with respect to the mesh-parameter h.



(a) Control (b) Desired state (c) Computed state

Fig. 5.1: Control, desired state, and state for time-independent distributed control with $\beta = 10^{-6}$.



(a) Control (b) Computed state

Fig. 5.2: Control and computed state for time-independent distributed control with $\beta = 10^{-6}$ and no $\mathcal{H}_1$-term.

DoF	MINRES(T)	MINRES(T)	MINRES(T)
	$\beta = 10^{-2}$	$\beta = 10^{-4}$	$\beta = 10^{-6}$
729	5(0.23)	10(0.99)	17(0.82)
4913	6(2.07)	10(2.51)	22(5.37)
35937	8(9.16)	10(7.76)	24(18.14)
274625	8(60.89)	10(74.38)	24(161.36)
2146689	8(547.15)	10(660.26)	26(1853.41)

Table 5.1: Results obtained with Schur complement approximation $\hat{S}$ and varying β.

State constraints. In the next example we consider the introduction of state constraints for the time-independent control problem. As was shown in [37] the quality of the preconditioner can have a significant influence on the convergence of the Newton scheme. In our experience for smaller values of β and ε the tolerance of 10^{-4} was not always sufficient for the Newton method to reach convergence and the results shown in Table 5.2 are computed for the tolerance 10^{-6}. We also employed a nested-iteration

technique [29], which starts by solving the optimal control problem on a coarse mesh and then transferring the solution to the next finer mesh as an initial guess for the Newton method. As can be seen from Table 5.2 this leads to a small number of Newton steps on the fine meshes. We here consider

$$\bar{\mathbf{y}} = -\sin(2\pi x_0 x_1 x_2)\exp\left(-\left((x_0-0.5)^2+(x_1-0.5)^2+(x_2-0.5)^2\right)\right)$$

and the Dirichlet condition is defined as $y = P_{[y_a,y_b]}(\bar{\mathbf{y}})$ on $\partial\Omega$ as the projection of the desired state onto the feasible region. Here we only consider the lower bound given by $y_a = -0.7$. We show the results for two different values of ε using $\beta = 10^{-6}$ in Table 5.2 and it can be seen that the number of Newton iterations is very similar and the MINRES iterations grow but stay reasonable with the reduction of the parameter ε.

DoF	AS	MINRES (tl/av)	T	AS	MINRES (tl/av)	T
	$\varepsilon = 10^{-2}$			$\varepsilon = 10^{-4}$		
729	5	145/29	11.38	5	156/31	12.23
4913	4	137/34	78.21	5	280/56	158.79
35937	4	154/39	285.59	5	351/70	689.61
274625	4	164/41	3589.12	6	448/74	10795.51

Table 5.2: Results obtained for state-constrained problem for different values of the penalty parameter. Total and average number of MINRES iterations for all Newton steps are shown as well as the timings for $\beta = 10^{-6}$.

Boundary control. The control of the PDE via the boundary of the domain represents a relevant and interesting scenario. We now illustrate how our preconditioner performs for this case. The desired state is given by

$$\bar{\mathbf{y}} = \begin{cases} \sin(x_1) + x_2 x_0 & \text{if } x_0 > 0.5 \text{ and } x_1 < 0.5 \\ 1 & \text{otherwise.} \end{cases}$$

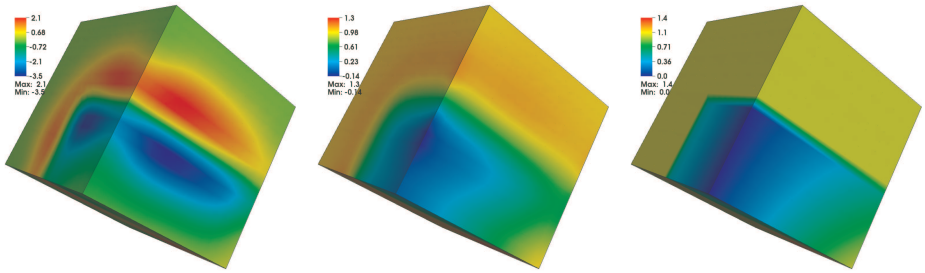
Table 5.3 shows the MINRES iteration numbers and timings for different meshes and values of β. Here we see that in the case of boundary control our diagonal approximation of the Neumann Laplacian is no longer sufficient, as there is now a mesh dependence on h ([6]) and a fairly strong dependence on the regularization parameter β. It is possible to alleviate these problems by using the nonsymmetric Schur complement $\hat{S}_3$ together with BICG, but for reasons of space we will only show results for this strategy in the more complicated time–dependent case later in this section.

5.2. The time-dependent case.

No state constraints. In this Section we show results for the time-dependent case. First, we consider the case when no state constraints are present. Here, we work with a fixed time-step $\tau = 0.05$, which results in 20 time-steps. In all tables we only show the degrees of freedom associated with the discretization of the spatial domain. The desired state is now given by

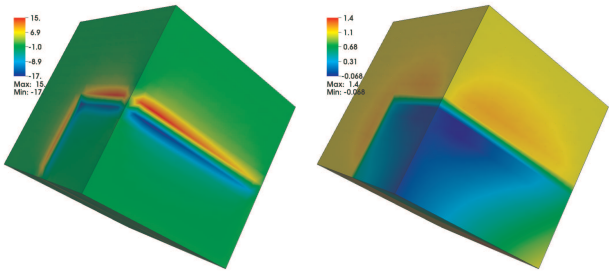
$$\bar{\mathbf{y}} = -\exp(t)\sin(2\pi x_0 x_1 x_2)\exp\left(-\left((x_0-0.5)^2+(x_1-0.5)^2+(x_2-0.5)^2\right)\right)$$

and $\mathbf{y} = \bar{\mathbf{y}}$ on $\partial\Omega$. The results for this setup are shown in Table 5.4 for various mesh-parameters and values of the regularization parameter β. We can see that the



(a) Control (b) Computed state (c) Desired state

Fig. 5.3: Desired state, computed state and control for a boundary control problem with $\beta = 10^{-6}$.



(a) Control (b) Computed state

Fig. 5.4: Same setup as in Figure 5.3 only with L_2 instead of $\mathcal{H}_1$ term.

DoF	MINRES(T)	MINRES(T)	MINRES(T)
	$\beta = 10^{-2}$	$\beta = 10^{-4}$	$\beta = 10^{-6}$
729	22 (0.08)	22 (0.09)	44 (0.17)
4913	24 (0.58)	32 (0.77)	62 (1.51)
35937	28 (5.08)	42 (7.35)	90 (15.59)
274625	26 (43.1)	56 (79.9)	127 (180.0)
2146689	24 (318.6)	74 (827.9)	152 (1686.)

Table 5.3: MINRES timings and iteration numbers for the time-independent boundary control problem with no state constraints, with varying mesh sizes as well as different values of the regularization parameter β.

results are no longer independent of β but the growth is moderate and the problem is still feasible even for quite small β.

In Table 5.5 we show results for the same problem, except using the alternative Schur complement approximation $\hat{S}_3$. In this distributed control case the diagonal approximation to the Laplacian works reasonably well in the preconditioner and is

DoF	MINRES(T)	MINRES(T)	MINRES(T)
	$\beta = 10^{-2}$	$\beta = 10^{-4}$	$\beta = 10^{-6}$
729	4 (0.51)	19 (1.80)	53 (4.89)
4913	6 (4.75)	21 (13.72)	69 (43.88)
35937	7 (43.07)	21 (109.3)	79 (385.2)

Table 5.4: MINRES iteration numbers and timings for various meshes and varying regularization parameter β using the Schur complement approximation $\hat{S}_2$ for the time-dependent distributed control problem with no state constraints.

very cheap to apply, so $\hat{S}_3$ is not that attractive, but the results do show significantly smaller iteration counts and better robustness with respect to h and β, and the expense of dramatically increased computational time, mostly due to the Uzawa iteration. Since we will be forced to use this nonsymmetric preconditioning strategy for our time–dependent boundary control problems, using it here allows us to compare its cost to the symmetric version.

DoF	BICG(T)	BICG(T)	BICG(T)
	$\beta = 10^{-2}$	$\beta = 10^{-4}$	$\beta = 10^{-6}$
729	6 (9.93)	8 (13.85)	15 (21.21)
4913	5 (60.7)	8 (93.30)	13 (130.6)
35937	5 (495.3)	8 (700.4)	13 (1051.)

Table 5.5: BICG iteration numbers and timings for various meshes and varying regularization parameter β using the Schur complement approximation $\hat{S}_3$ for the time-dependent distributed control problem with no state constraints. For approximating the system (4.4) we use 10 damped Uzawa steps.

State constraints. We now present results for the time-dependent case in the presence of state constraints. We again consider the unit square with the desired state defined by

$$\bar{\mathbf{y}} = -\exp(t)\sin(2.0\pi x_1 x_2 x_3)\exp\left(-\left((x_0-0.5)^2+(x_1-0.5)^2+(x_2-0.5)^2\right)\right).$$

The iteration numbers for the outer active set method and MINRES are shown in Table 5.6. We only show results for two small mesh-sizes as our implementation is currently only a proof-of-concept implementation that does not update the preconditioner for each active set but rather recompute it during each iteration, which in practice will be too expensive. Though this is a difficult problem and the iteration numbers are large, especially for small ϵ, we see that it is possible to solve such a problem using our approach.

Boundary control. We now show results for the boundary control case when we are dealing with a time-dependent problem. The desired state is given by

$$\bar{\mathbf{y}} = -\exp(t)\sin(2\pi x_0 x_1 x_2)\exp\left(-\left((x_0-0.5)^2+(x_1-0.5)^2+(x_2-0.5)^2\right)\right).$$

For this setup the diagonal approximation in $\hat{S}_2$ with MINRES completely fails. The results with the Schur complement approximation $\hat{S}_3$ with varying mesh-size and

DoF	AS	MINRES (tl/av)	AS	MINRES (tl/av)
	$\varepsilon = 10^{-2}$		$\varepsilon = 10^{-4}$	
729	6	252/42.0	9	1136/126.2
4913	4	202/50.5	5	545/109.0

Table 5.6: Results obtained for state-constrained problem for different values of the penalty parameter. Total and average number of MINRES iterations for all Newton steps are shown as well as the timings for $\beta = 10^{-4}$.

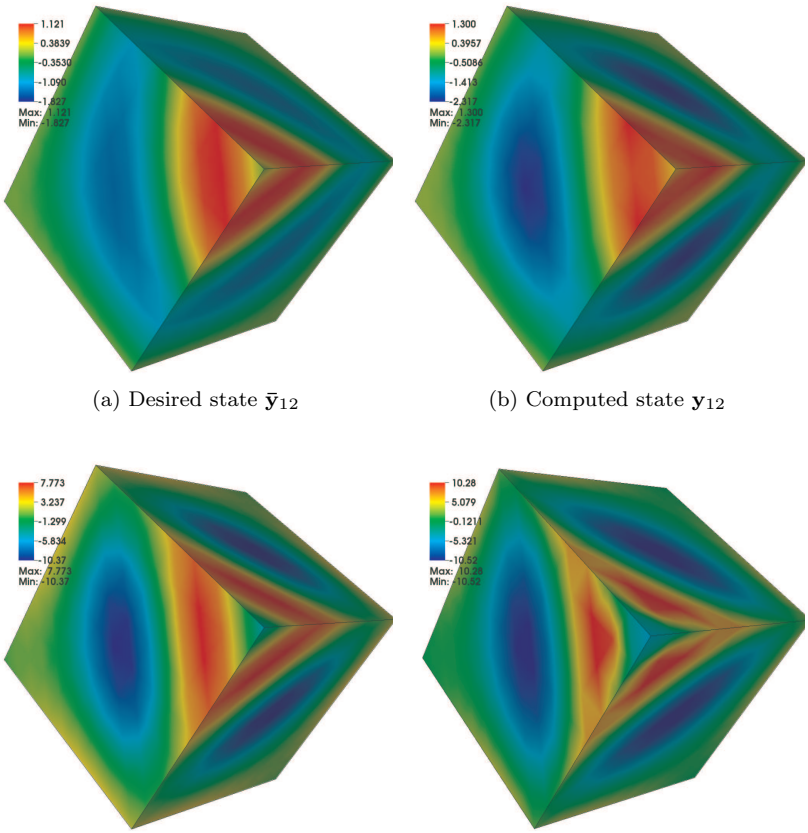
regularization parameter β are shown in Table 5.7. The method performs with good mesh-independence for moderate β, but for very small β this becomes quite a difficult problem. If you are willing to pay the computational cost for a large number of Uzawa steps within the Schur complement approximation, then this approach can be made nearly mesh independent in terms of iteration counts. If on the other hand you use fewer Uzawa steps the computational cost is lower but the mesh independence is lost.

DoF	BICG(T)	BICG(T)	BICG(T)	BICG(T)	BICG(T)
	$\beta = 10^{-2}$ $N_T = 20$ $\rho = 0.15$ 20 U steps	$\beta = 10^{-4}$ $N_T = 20$ $\rho = 0.15$ 20 U steps	$\beta = 10^{-6}$ $N_T = 20$ $\rho = 0.2$ 40 U steps	$\beta = 10^{-6}$ $N_T = 20$ $\rho = 0.2$ 60 U steps	$\beta = 10^{-6}$ $N_T = 20$ $\rho = 0.2$ 100 U steps
729	19 (21.78)	18 (22.13)	66 (134.8)	70 (211.6)	42 (229.7)
4913	19 (133.5)	20 (153.4)	71 (893.1)	72 (1333.)	45 (1543)
35937	19 (1005.)	23 (1205.)	92 (8713.)	79 (11050)	50 (11750)

Table 5.7: BICG iteration numbers and timings for various meshes and varying regularization parameter β for the time-dependent boundary control problem with no state constraints using the Schur complement $\hat{S}_3$.

6. Conclusions and Outlook. In this paper we presented optimal control problems subject to the Poisson equation or the heat equation in a distributed or boundary control setting. The control was added to the objective function as a regularization term in the $\mathcal{H}_1$ norm. We introduced the corresponding discrete optimality system and introduced preconditioners for both the steady as well as the transient problem. Due to the Laplacian term coming from the $\mathcal{H}_1$ norm we were not able to introduce preconditioners that are fully independent of the regularization parameter but for the simple preconditioners we introduced the dependence on the regularization parameter seemed rather weak. We also showed that our approach works for state-constrained problems, which were treated using a Moreau-Yosida penalty approach. Numerical results showed that our preconditioners provided satisfactory results when applied to three-dimensional test problems.

The method presented here has not focused on the storage efficiency of our all-at-once approach. One might employ checkpointing [25] techniques when alternately solving forward and adjoint PDEs. Multiple shooting approaches are one way of splitting up the time-interval [27] and can lead to the same type of system. A possible way forward is to compute suboptimal solutions on a sequential splitting of the time-interval [27] or to use a parallel implementation of our approach. It is also possible



(a) Desired state $\bar{\mathbf{y}}_{12}$ (b) Computed state $\mathbf{y}_{12}$

(c) Control $\mathbf{y}_{12}$ with $\mathcal{H}_1$-term. (d) Control $\mathbf{y}_{12}$ with L_2-term.

Fig. 5.5: Control with and without $\mathcal{H}_1$ term as well as desired state and state for time-dependent boundary control with $\beta = 10^{-6}$.

to reduce the storage requirements by performing block-eliminations of some form, usually via a Schur-complement approach.

REFERENCES

[1] Bangerth, W., Hartmann, R., Kanschat, G.: deal.II—a general-purpose object-oriented finite element library. ACM Trans. Math. Software **33**(4), Art. 24, 27 (2007)

[2] Benzi, M., Golub, G.H., Liesen, J.: Numerical solution of saddle point problems. Acta Numer **14**, 1–137 (2005)

[3] Benzi, M., Haber, E., Taralli, L.: A preconditioning technique for a class of PDE-constrained optimization problems. Advances in Computational Mathematics **35**, 149–173 (2011)

[4] Bergounioux, M., Ito, K., Kunisch, K.: Primal-dual strategy for constrained optimal control problems. SIAM J. Control Optim. **37**(4), 1176–1194 (1999)

[5] Bergounioux, M., Kunisch, K.:Primal-dual strategy for state-constrained optimal control problems, Comput. Optim. Appl. **22**(2), 193–224 (2002) DOI 10.1023/A:1015489608037

[6] Bochev, P., Lehoucq, R.: On the finite element solution of the pure neumann problem. SIAM review **47**(1), 50–66 (2005)

[7] Bramble, J.H., Pasciak, J.E.: A preconditioning technique for indefinite systems resulting from

mixed approximations of elliptic problems. Math. Comp **50**(181), 1–17 (1988)
[8] Cai, Xiao-Chuan and Liu, Si and Zou, Jun: An overlapping domain decomposition method for parameter identification problems. Domain Decomposition Methods in Science and Engineering XVII, **60**, 451–458 (2008)
[9] Casas, E.: Control of an elliptic problem with pointwise state constraints. SIAM J. Control Optim. **24**(6), 1309–1318 (1986). DOI 10.1137/0324078. URL `http://dx.doi.org/10.1137/0324078`
[10] Casas, E., Herzog, R. and Wachsmuth, G.: Approximation of sparse controls in semilinear equations by piecewise linear functions. Numerische Mathematik **122**, 645–669 (2012)
[11] Chan, R.H. and Chan, T.F. and Wan, WL and others: Multigrid for differential-convolution problems arising from image processing. Proc. Workshop on Scientific Computing, pp. 58–72 (1997)
[12] Chan, T.F. and Tai, X.C.: Identification of discontinuous coefficients in elliptic problems using total variation regularization. SIAM Journal on Scientific Computing **25**(3) 881–904 (2003)
[13] Christofides, P.: Nonlinear and robust control of PDE systems: Methods and applications to transport-reaction processes. Birkhauser (2001)
[14] Cimrák, I. and Melicher, V.: Mixed Tikhonov regularization in Banach spaces based on domain decomposition submitted to Applied Mathematics and Computation (2012)
[15] Collis, S.S., Ghayour, K., Heinkenschloss, M., Ulbrich, M. and Ulbrich, S.: Numerical solution of optimal control problems governed by the compressible Navier-Stokes equations International series of numerical mathematics, 43–56 (2002)
[16] van den Doel, K. and Ascher, U. and Haber, E.: The lost honour of l2-based regularization Sumbitted (2012)
[17] Du, X., Sarkis, M., Schaerer, C.E., Szyld, D.B.: Inexact and truncated parareal-in-time Krylov subspace methods for parabolic optimal control problems. Tech. Rep. 12-02-06, Department of Mathematics, Temple University (2012)
[18] Duff, I.S., Erisman, A.M., Reid, J.K.: Direct methods for sparse matrices. Monographs on Numerical Analysis. The Clarendon Press Oxford University Press, New York (1989)
[19] Elman, H.C., Silvester, D.J., Wathen, A.J.: Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics. Numerical Mathematics and Scientific Computation. Oxford University Press, New York (2005)
[20] Fletcher, R.: Conjugate gradient methods for indefinite systems, *Lecture Notes in Mathematics*, vol. 506. Springer–Verlag, Berlin (1976), 73–89.
[21] Freund, R.W., Nachtigal, N.M.: QMR: a quasi-minimal residual method for non-Hermitian linear systems. Num. Math. **60**(1) 315–339 (1991)
[22] Gee, M., Siefert, C., Hu, J., Tuminaro, R., Sala, M.: ML 5.0 smoothed aggregation user's guide. Tech. Rep. SAND2006-2649, Sandia National Laboratories (2006)
[23] Gunzburger, Max D.: Perspectives in flow control and optimization, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2003)
[24] Greenbaum, A.: Iterative methods for solving linear systems, *Frontiers in Applied Mathematics*, vol. 17. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (1997)
[25] Griewank, A., Walther, A.: Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. ACM Transactions on Mathematical Software (TOMS) **26**(1), 19–45 (2000)
[26] Haber, E. And Hanson, L.: Model Problems in PDE-Constrained Optimization Emory University TR-2007-009 (2007)
[27] Heinkenschloss, M.: A time-domain decomposition iterative method for the solution of distributed linear quadratic optimal control problems. J. Comput. Appl. Math. **173**(1), 169–198 (2005). DOI 10.1016/j.cam.2004.03.005
[28] Heinkenschloss, M.: Formulation and analysis of a sequential quadratic programming method for the optimal Dirichlet boundary control of Navier-Stokes flow Optimal Control, Theory, Algorithms, and Applications (1998)
[29] Herzog, R., Sachs, E.W.: Preconditioned conjugate gradient method for optimal control problems with control and state constraints. SIAM J. Matrix Anal. Appl. **31**(5), 2291–2317 (2010)
[30] Hestenes, M.R., Stiefel, E.: Methods of conjugate gradients for solving linear systems. J. Res. Nat. Bur. Stand **49**, 409–436 (1953) (1952)
[31] Hintermüller, M., Ito, K., Kunisch, K.: The primal-dual active set strategy as a semismooth Newton method. SIAM J. Optim. **13**(3), 865–888 (2002)
[32] Hintermüller, M., Kunisch, K.: Path-following methods for a class of constrained minimization problems in function space. SIAM J. Optim. **17**(1), 159–187 (2006)
[33] Hinze, M., Köster, M., Turek, S.: A Hierarchical Space-Time Solver for Distributed Control of

the Stokes Equation. Tech. rep., SPP1253-16-01 (2008)
[34] Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications. Springer-Verlag, New York (2009)
[35] Ito, K., Kunisch, K.: Semi-smooth Newton methods for state-constrained optimal control problems. Systems Control Lett. **50**(3), 221–228 (2003). DOI 10.1016/S0167-6911(03)00156-7
[36] Ito, K., Kunisch, K.: Lagrange multiplier approach to variational problems and applications, *Advances in Design and Control*, vol. 15. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2008)
[37] Kanzow, C.: Inexact semismooth Newton methods for large-scale complementarity problems. Optimization Methods and Software **19**(3-4), 309–325 (2004). DOI 10.1080/10556780310001636369. URL `http://www.tandfonline.com/doi/abs/10.1080/10556780310001636369`
[38] Keung, Y.L. and Zou, J.: Numerical identifications of parameters in parabolic systems Inverse Problems **14**(1), 83–100 (1999)
[39] Kollmann, M., Kolmbauer, M.: A Preconditioned MinRes Solver for Time-Periodic Parabolic Optimal Control Problems. Sumitted,Numa-Report 2011-06 (August 2011)
[40] Li, F. and Shen, C. and Li, C.: Multiphase Soft Segmentation with Total Variation and H_1 Regularization Journal of Mathematical Imaging and Vision **37**(2), 98–111 (2010)
[41] Ng, M.K. and Chan, R.H. and Chan, T.F. and Yip, A.M.: Cosine transform preconditioners for high resolution image reconstruction Linear Algebra and its Applications **316**(1) 89–104 (2000)
[42] Murphy, M.F., Golub, G.H., Wathen, A.J.: A note on preconditioning for indefinite linear systems. SIAM J. Sci. Comput **21**(6), 1969–1972 (2000)
[43] Paige, C.C., Saunders, M.A.: Solutions of sparse indefinite systems of linear equations. SIAM J. Numer. Anal **12**(4), 617–629 (1975)
[44] Pearson, J.W., Stoll, M., Wathen, A.: Preconditioners for state constrained optimal control problems with Moreau-Yosida penalty function. Submitted (2011)
[45] Pearson, J.W., Stoll, M., Wathen, A.J.: Regularization-robust preconditioners for time-dependent PDE-constrained optimization problems. To appear SIAM J. Matrix Anal.Appl (2012)
[46] Pearson, J.W., Wathen, A.J.: A new approximation of the schur complement in preconditioners for pde-constrained optimization. Numerical Linear Algebra with Applications **19**, 816–829 (2012). DOI 10.1002/nla.814. URL `http://dx.doi.org/10.1002/nla.814`
[47] Peirce, A., Dahleh, M., Rabitz, H.: Optimal control of quantum-mechanical systems: Existence, numerical approximation, and applications. Physical Review A **37**(12), 4950 (1988)
[48] Pironneau, O.: Optimal shape design for elliptic systems. System Modeling and Optimization pp. 42–66 (1982)
[49] Rees, T., Dollar, H.S., Wathen, A.J.: Optimal solvers for PDE-constrained optimization. SIAM Journal on Scientific Computing **32**(1), 271–298 (2010). DOI http://dx.doi.org/10.1137/080727154
[50] Rees, T., Stoll, M., Wathen, A.: All-at-once preconditioners for PDE-constrained optimization. Kybernetika **46**, 341–360 (2010)
[51] Juan-Carlos De los Reyes and Schönlieb, Carola-Bibiane: Image denoising: learning noise distribution via PDE-constrained optimization, http://arxiv.org/abs/1207.3425 (2012)
[52] Saad, Y.: Iterative methods for sparse linear systems. Society for Industrial and Applied Mathematics, Philadelphia, PA (2003)
[53] Saad, Y., Schultz, M.H.: GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems. SIAM J. Sci. Stat. Comput., **7**(3), 856–869 (1986).
[54] Simoncini, V., Szyld, D.: Recent computational developments in Krylov subspace methods for linear systems. Numer. Linear Algebra Appl **14**(1), 1–61 (2007).
[55] Stoll, M.: All-at-once solution of a time-dependent time-periodic PDE-constrained optimization problems. Submitted (July 2011)
[56] Stoll, M., Wathen, A.: All-at-once solution of time-dependent PDE-constrained optimization problems. Submitted (2010)
[57] Strang, G., Fix, G.: An Analysis of the Finite Element Method 2nd Edition, 2nd edn. Wellesley-Cambridge (2008)
[58] Takacs, S., Zulehner, W.: Convergence analysis of multigrid methods with collective point smoothers for optimal control problems. Computing and Visualization in Science **14**, 131–141 (2011)
[59] Tröltzsch, F.: Optimal Control of Partial Differential Equations: Theory, Methods and Applications. Amer Mathematical Society (2010)
[60] Ulbrich, M.: Semismooth Newton Methods for Variational Inequalities and Constrained Opti-

mization Problems. SIAM Philadelphia (2011)
[61] Van Der Vorst, H.A.: BI–CGSTAB: A fast and smoothly converging variant of BI–CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **13**(2), 631–634 (1992).
[62] Wathen, A.J., Rees, T.: Chebyshev semi-iteration in preconditioning for problems including the mass matrix. Electronic Transactions in Numerical Analysis **34**, 125–135 (2008)
[63] Wachsmuth, D. and Wachsmuth, G.: Necessary conditions for convergence rates of regularizations of optimal control problems, RICAM Report **4** (2012)
[64] Wilson, J. and Patwari, N. and Vasquez, F.G.: Regularization methods for radio tomographic imaging. 2009 Virginia Tech Symposium on Wireless Personal Communications (2009)
[65] Zulehner, W.: Analysis of iterative methods for saddle point problems: a unified approach. Math. Comp **71**(238), 479–505 (2002)