FAST INTERIOR POINT SOLUTION OF QUADRATIC PROGRAMMING PROBLEMS ARISING FROM PDE-CONSTRAINED OPTIMIZATION

JOHN W. PEARSON* AND JACEK GONDZIO†

Abstract. Interior point methods provide an attractive class of approaches for solving linear, quadratic and nonlinear programming problems, due to their excellent efficiency and wide applicability. In this paper, we consider PDE-constrained optimization problems with bound constraints on the state and control variables, and their representation on the discrete level as quadratic programming problems. To tackle complex problems and achieve high accuracy in the solution, one is required to solve matrix systems of huge scale resulting from Newton iteration, and hence fast and robust methods for these systems are required. We present preconditioned iterative techniques for solving a number of these problems using Krylov subspace methods, considering in what circumstances one may predict rapid convergence of the solvers in theory, as well as the solutions observed from practical computations.

Key words. Interior point methods, PDE-constrained optimization, Krylov subspace methods, Preconditioning, Schur complement.

AMS subject classifications. 65F08, 65F10, 65F50, 76D05, 76D55, 93C20

1. Introduction.

We are concerned with optimization problems which involve partial differential equations. Problems of this type appear for example in numerous applications of optimal control, where one wishes state variables to be close to a certain desired form and hopes to achieve it by an appropriate choice of control variables. Let $\Omega \subset \mathbb{R}^d$ be a bounded open domain with sufficiently smooth boundary $\partial\Omega$. An optimal control problem with constraints may be written as:

$$\min_{y \in Y,\ u \in U} \quad \mathcal{J}(y,u) \quad \text{s.t.} \quad c(y,u) = 0, \tag{1.1}$$

where the state y and control u belong to appropriate functional spaces Y and U, respectively. The objective $\mathcal{J} : Y \times U \mapsto \mathbb{R}$ and the constraints $c : Y \times U \mapsto \Lambda$, where Λ is another functional space, are assumed to satisfy certain smoothness conditions to guarantee the existence and uniqueness of the solution. Many real-life problems may be modelled as optimal control problems (1.1). There exists rich literature on the subject which addresses specific applications and provides theoretical background to such problems. The rigorous analysis of optimal control problems requires using nontrivial functional spaces and involves sophisticated techniques from functional analysis. We refer the interested reader to excellent books on the subject [13, 14, 32], while for simplicity in this paper we assume that Y, U and Λ are all equal to $L_2(\Omega)$.

The objective function $\mathcal{J}$ may take many different forms but it is often given as:

$$\mathcal{J}(y,u) = \frac{1}{2}\|y - \widehat{y}\|^2_{L_2(\Omega)} + \frac{\beta}{2}\|u\|^2_{L_2(\Omega)}, \tag{1.2}$$

which corresponds to balancing between two goals: keeping the state y close to a certain desired form $\widehat{y}$, and minimizing the "energy" of the applied control u. The

*School of Mathematics, Statistics and Actuarial Science, University of Kent, Cornwallis Building (East), Canterbury, CT2 7NF, UK (j.w.pearson@kent.ac.uk),

†School of Mathematics, The University of Edinburgh, James Clerk Maxwell Building, The King's Buildings, Peter Guthrie Tait Road, Edinburgh, EH9 3FD, UK (j.gondzio@ed.ac.uk)

constraints c in (1.1) involve some PDE operator(s), and restrict y and u to Ω and its boundary $\partial\Omega$. Additionally they may include simple bounds on y and u. In Section 3 we will introduce two particular classes of optimal control problems: time-invariant and time-dependent PDE-constrained problems.

Computational techniques for PDE-constrained optimal control problems involve a discretization of the underlying PDE. There are two options for doing this, and the typical paradigm in PDE-constrained optimization literature is for both approaches to solve the problem in a similar manner. The first is to apply an *optimize-then-discretize* method, involving constructing continuous optimality conditions, and then discretizing these. However we find that this approach is inconvenient when considering the resulting discrete systems for the problems considered in this paper, specifically with regard to the reduction of the dimension of the system, as well as symmetry of the matrix involved. The alternative method, which we apply in this paper, is the *discretize-then-optimize* approach: here a discrete cost functional is constructed and discretized constraints are formulated. Then optimality conditions are derived for such (possibly huge) problems. Our motivation for using this approach originates from an observation that for a particular (quadratic) cost functional (1.2) the discretized PDE-constrained problem takes the form of a quadratic optimization problem. The use of fine discretization leads to a substantial size of the resulting optimization problem. Therefore we will apply an interior point algorithm to solve it.

Interior point methods (IPMs) are very well-suited to solving quadratic optimization problems and they excel when sizes of problems grow large [10, 39], which makes them perfect candidates for discretized PDE-constrained optimal control problems. The use of IPMs in PDE-constrained optimization is not new. There have been several developments which address the theoretical aspects of it, include the functional analysis viewpoint, and study the convergence properties of an interior point algorithm [33, 36, 38], and many others which focus on the practical (computational) aspects. IPMs belong to a broad class of methods which rely on the use of Newton methods to compute optimizing directions. There have been several successful attempts to use Newton-based approaches in the PDE-constrained optimization context [3, 4, 15, 17]. The main computational challenge in these approaches is the solution of linear system which determines the Newton direction. For fine PDE discretizations such systems quickly get very large. Additionally, when IPMs are applied, the added interior point diagonal scaling matrices degrade the conditioning of such systems [10] and make them numerically challenging. Direct methods for sparse linear algebra [6] can handle the ill-conditioning well but struggle with excessive memory requirements when problems get larger. Inexact interior point methods [11, 12, 37] overcome this difficulty by employing iterative methods to solve the Newton equations.

Because of the unavoidable ill-conditioning of these equations the success of any iterative scheme for their solution depends on the ability to design efficient *preconditioners* which can improve spectral properties of linear systems. A development of such preconditioners is a very active research area. Preconditioners for IPMs in PDE-constrained optimization exploit the vast experience gathered for saddle point systems [1], but face an extra difficulty originating from the presence of IPM scaling. There have been already several successful attempts to design preconditioners for such systems, see [2, 12] and the references therein.

In this paper, we propose a general methodology to design efficient preconditioners for such systems. Our approach is derived from the *matching strategy* originally developed for a particular Poisson control problem [26]. We adapt it to much more

challenging circumstances of saddle point systems arising in IPMs applied to solve the PDE-constrained optimal control problems. We briefly comment on the enjoyable spectral properties of the preconditioned system, and provide computational results to demonstrate that they work well in practice.

This paper is structured as follows. In Section 2 we briefly recall a few basic facts about interior point methods for quadratic programming. In Section 3 we demonstrate how IPMs can be applied to PDE-constrained optimization problems. In Section 4 we introduce the preconditioners proposed for problems originating from optimal control. We consider separately two different cases of time-independent and time-dependent problems. In Section 5 we illustrate our findings with computational results and, finally, in Section 6 we give our conclusions.

2. Interior point methods for quadratic programming. Within this paper, we are interested in the solution of *quadratic programming* (QP) problems. In their most basic form, such problems may be written as

$$\begin{aligned} \min_{\vec{x}} \quad & \vec{c}^{\top}\vec{x} + \frac{1}{2}\vec{x}^{\top}Q\vec{x} \\ \text{s.t.} \quad & A\vec{x} = \vec{b}, \\ & \vec{x} \geq \vec{0}. \end{aligned} \tag{2.1}$$

We consider the case where $A \in \mathbb{R}^{m\times n}$ $(m \leq n)$ has full row rank, $Q \in \mathbb{R}^{n\times n}$ is positive semidefinite, $\vec{x}, \vec{c} \in \mathbb{R}^n$, and $\vec{b} \in \mathbb{R}^m$. This formulation is frequently considered alongside its *dual problem*

$$\begin{aligned} \max_{\vec{y}} \quad & \vec{b}^{\top}\vec{y} - \frac{1}{2}\vec{x}^{\top}Q\vec{x} \\ \text{s.t.} \quad & A^{\top}\vec{y} + \vec{z} - Q\vec{x} = \vec{c}, \\ & \vec{y} \ \text{ free}, \quad \vec{z} \geq \vec{0}, \end{aligned}$$

where $\vec{z} \in \mathbb{R}^n$, and $\vec{y} \in \mathbb{R}^m$. We note that a subset of this setup is that of linear programming (LP) problems, where $Q = 0$.

In this manuscript, we consider the solution of quadratic programming problems using interior point methods [10]. The nonnegativity constraints $\vec{x} \geq \vec{0}$ are "replaced" with the logarithmic barrier penalty function, and the Lagrangian associated with the barrier subproblem is formed:

$$\mathcal{L}(\vec{x}, \vec{y}) = \vec{c}^{\top}\vec{x} + \frac{1}{2}\vec{x}^{\top}Q\vec{x} + \vec{y}^{\top}\left(A\vec{x} - \vec{b}\right) - \mu\sum_j \log(x_j).$$

Applying Lagrangian duality theory [5], stationarity conditions for the Lagrangian are derived

$$\begin{aligned} A\vec{x} &= \vec{b}, \\ A^{\top}\vec{y} + \vec{z} - Q\vec{x} &= \vec{c}, \\ x_j z_j &= \mu, \quad j = 1, ..., n, \\ (\vec{x}, \vec{z}) &\geq 0, \end{aligned} \tag{2.2}$$

in which the standard complementarity condition for (2.1), that is $x_j z_j = 0, \forall j$, is replaced with the purturbed complementarity condition $x_j z_j = \mu, \forall j$. IPMs drive the

barrier term μ to zero and gradually reveal the activity of the primal variables x_j and dual slacks z_j. This is achieved by applying Newton's method to the system of (mildly) nonlinear equations (2.2)

$$\begin{bmatrix} -Q & A^\top & I \\ A & 0 & 0 \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \vec{s}_x \\ \vec{s}_y \\ \vec{s}_z \end{bmatrix} = \begin{bmatrix} \vec{\xi}_d \\ \vec{\xi}_p \\ \vec{\xi}_c \end{bmatrix}, \tag{2.3}$$

where $\vec{s}_x$, $\vec{s}_y$ and $\vec{s}_z$ denote Newton direction, $\vec{\xi}_d$, $\vec{\xi}_p$ and $\vec{\xi}_c$ denote primal and dual infeasibilities and the violation of complementarity conditions, respectively. X and Z denote diagonal matrices with elements of $\vec{x}$ and $\vec{z}$ spread on the diagonals, respectively. By eliminating $\vec{s}_z$, the Newton system (2.3) is further reduced to a saddle point form

$$\begin{bmatrix} -Q - X^{-1}Z & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \vec{s}_x \\ \vec{s}_y \end{bmatrix} = \begin{bmatrix} \vec{\xi}_d - X^{-1}\vec{\xi}_c \\ \vec{\xi}_p \end{bmatrix}. \tag{2.4}$$

Since for any $j = 1, 2, \ldots, n$ at least one of the variables x_j and z_j reaches zero at optimality, the elements of diagonal scaling matrix $X^{-1}Z$ added to the (1,1) block may significantly differ in magnitude: some of them go to zero while the other go to infinity. This feature of IPMs [10] is a challenge for any linear equation solver applied to (2.4). We skip further details about IPMs and refer the interested reader to [10, 39]. We also highlight that $\vec{y}$ in this description relates to a dual variable, whereas for PDE-constrained optimization the function y corresponds to a primal variable – we elect to use the standard notation within the respective fields.

However, before moving on to PDE-constrained optimization, it is worth drawing reader's attention to the fact that although in (2.1) we assume only one-sided bound $\vec{x} \geq \vec{0}$, IPMs can be also easily applied to variables with two-sided bounds:

$$\vec{x}_a \leq \vec{x} \leq \vec{x}_b.$$

This requires introducing two nonnegative Lagrange multipliers associated with two inequalities. Later on we will denote them as $\vec{z}_a$ and $\vec{z}_b$, respectively.

3. PDE-constrained optimization. We now wish to demonstrate how interior point methods may be applied to PDE-constrained optimization problems. These are a crucial class of problems which may be used to model a range of applications in science and industry, for example fluid flow, chemical and biological processes, shape optimization, imaging problems, and mathematical finance, to name but a few. However the problems are often of complex structure, and sophisticated techniques are frequently required to achieve accurate solutions for the models being considered. We recommend the papers [13, 32], which provide an excellent introduction to the field.

Let us first consider a time-independent linear PDE-constrained optimization problem with additional bound constraints:

$$\begin{aligned} \min_{y,u} \quad & \frac{1}{2}\left\|y - \widehat{y}\right\|^2_{L_2(\Omega)} + \frac{\beta}{2}\left\|u\right\|^2_{L_2(\Omega)} \\ \text{s.t.} \quad & \mathcal{L}y = u, \quad \text{in } \Omega, \\ & y = f, \quad \text{on } \partial\Omega, \\ & y_a \leq y \leq y_b, \quad \text{a.e. in } \Omega, \\ & u_a \leq u \leq u_b, \quad \text{a.e. in } \Omega. \end{aligned} \tag{3.1}$$

Here y, $\widehat{y}$, u denote the *state*, *desired state* and *control variables*, with $\mathcal{L}$ some PDE operator, and β a positive *regularization parameter*. The problem is solved on domain Ω (with boundary $\partial\Omega$), for given functions f, y_a, y_b, u_a, u_b. We will now apply the discretize-then-optimize approach to (3.1).

We wish to construct a finite element discretization of it: for the problems considered in this paper it is beneficial to use equal order finite elements for state and control variables, and observe that a discretized approximation of the cost functional is

$$\frac{1}{2}\left\|y-\widehat{y}\right\|^2_{L_2(\Omega)}+\frac{\beta}{2}\left\|u\right\|^2_{L_2(\Omega)}\approx\frac{1}{2}\vec{y}^{\top}M\vec{y}-\vec{y}_d^{\top}\vec{y}+\underbrace{\frac{1}{2}\int_{\Omega}\widehat{y}^2\,\mathrm{d}\Omega}_{\text{constant}}+\frac{\beta}{2}\vec{u}^{\top}M\vec{u},$$

where $\vec{y}$, $\vec{u}$ are the discretized versions of y, u. The finite element symmetric *mass matrix* M contains entries of the form $[M]_{ij}=\int_\Omega\phi_i\phi_j\,\mathrm{d}\Omega$, where $\{\phi_i\}$ are the finite element basis functions used, and $\vec{y}_d$ contains entries of the form $\int_\Omega\widehat{y}\phi_i\,\mathrm{d}\Omega$.

We therefore write (3.1) on the discrete level as

$$\begin{aligned}\min_{\vec{y},\vec{u}}\quad&\frac{1}{2}\vec{y}^{\top}M\vec{y}-\vec{y}_d^{\top}\vec{y}+\frac{\beta}{2}\vec{u}^{\top}M\vec{u}\\ \text{s.t.}\quad&K\vec{y}-M\vec{u}=\vec{f},\\ &\vec{y}_a\le\vec{y}\le\vec{y}_b,\\ &\vec{u}_a\le\vec{u}\le\vec{u}_b,\end{aligned}\tag{3.2}$$

with $\vec{f}$, $\vec{y}_a$, $\vec{y}_b$, $\vec{u}_a$, $\vec{u}_b$ the discrete versions of f, y_a, y_b, u_a, u_b. The matrix K depends on the PDE operator $\mathcal{L}$ considered: for example when a Poisson control problem (with $\mathcal{L}=-\nabla^2$) is examined, K denotes a finite element *stiffness matrix* with entries $[K]_{ij}=\int_\Omega\nabla\phi_i\cdot\nabla\phi_j\,\mathrm{d}\Omega$. Alternatively for convection-diffusion control problems (with $\mathcal{L}=-\nu\nabla^2+(\vec{w}\cdot\nabla)$, and without stabilization applied within the solution method), K contains a sum of diffusion and convection terms with $[K]_{ij}=\int_\Omega\left(\nu\nabla\phi_i\cdot\nabla\phi_j+(\vec{w}\cdot\nabla\phi_j)\phi_i\right)\mathrm{d}\Omega$.

We observe that, using our equal order finite element method, the matrices $M,K\in\mathbb{R}^{N\times N}$, where N denotes the number of finite element nodes used, and furthermore that $\vec{y},\vec{u}\in\mathbb{R}^N$.

It can easily be seen that the problem statement (3.2) is in the form of the quadratic programming problem (2.1), with

$$\vec{x}=\begin{bmatrix}\vec{y}\\ \vec{u}\end{bmatrix},\quad Q=\begin{bmatrix}M&0\\0&\beta M\end{bmatrix},\quad A=\begin{bmatrix}K&-M\end{bmatrix},$$
$$\vec{c}=\begin{bmatrix}-\vec{y}_d\\ \vec{0}\end{bmatrix},\quad \vec{x}_a=\begin{bmatrix}\vec{y}_a\\ \vec{u}_a\end{bmatrix},\quad \vec{x}_b=\begin{bmatrix}\vec{y}_b\\ \vec{u}_b\end{bmatrix}.$$

In the next section we consider interior point methods for solving a range of problems of structure (3.2). Although there has at this point been relatively little research into such strategies, we highlight that the paper [33] considers numerical solution of problems of this type with control constraints only.

3.1. Newton iteration. We now wish to derive the equations arising from a Newton iteration applied to the (nonlinear) problem (3.1). Let us define

$$\mathcal{J}\left(\vec{y},\vec{u}\right)=\frac{1}{2}\vec{y}^{\top}M\vec{y}-\vec{y}_d^{\top}\vec{y}+\frac{\beta}{2}\vec{u}^{\top}M\vec{u}$$

to be the function which we wish to minimize. Applying the discretized version of the PDE constraint, alongside a barrier function for the bound constraints as in the previous section, leads to the Lagrangian

$$\begin{aligned}\mathcal{L}(\vec{y},\vec{u},\vec{p}) = \mathcal{J}(\vec{y},\vec{u}) + \vec{p}^{\top}\left(K\vec{y} - M\vec{u} - \vec{f}\right) \\ - \mu\sum_j \log\left(y_j - y_{a,j}\right) - \mu\sum_j \log\left(y_{b,j} - y_j\right) \\ - \mu\sum_j \log\left(u_j - u_{a,j}\right) - \mu\sum_j \log\left(u_{b,j} - u_j\right),\end{aligned}$$

of which we wish to find the stationary point(s). Here $\vec{p}$ denotes the discretized *adjoint variable* (or *Lagrange multiplier*), y_j, $y_{a,j}$, $y_{b,j}$, u_j, $u_{a,j}$, $u_{b,j}$ denote the j-th entries of $\vec{y}$, $\vec{y}_a$, $\vec{y}_b$, $\vec{u}$, $\vec{u}_a$, $\vec{u}_b$, and μ is the *barrier parameter* used.

Differentiating $\mathcal{L}$ with respect to $\vec{y}$, $\vec{u}$ and $\vec{p}$ give the *first order optimality conditions* (or *Karush-Kuhn-Tucker conditions*):

$$M\vec{y} - \vec{y}_d + K^{\top}\vec{p} - \vec{z}_{y,a} + \vec{z}_{y,b} = \vec{0}, \tag{3.3}$$

$$\beta M\vec{u} - M\vec{p} - \vec{z}_{u,a} + \vec{z}_{u,b} = \vec{0}, \tag{3.4}$$

$$K\vec{y} - M\vec{u} - \vec{f} = \vec{0}, \tag{3.5}$$

where

$$(\vec{y} - \vec{y}_a) \circ \vec{z}_{y,a} = \mu\vec{e}, \tag{3.6}$$

$$(\vec{y}_b - \vec{y}) \circ \vec{z}_{y,b} = \mu\vec{e}, \tag{3.7}$$

$$(\vec{u} - \vec{u}_a) \circ \vec{z}_{u,a} = \mu\vec{e}, \tag{3.8}$$

$$(\vec{u}_b - \vec{u}) \circ \vec{z}_{u,b} = \mu\vec{e}, \tag{3.9}$$

$\vec{e}$ defines the vector of ones of appropriate dimension, and $\circ$ relates to the multiplication componentwise of two vectors. Note that, by construction, the following bound constraints apply for the Lagrange multipliers enforcing the bound constraints:

$$\vec{z}_{y,a} \geq \vec{0}, \qquad \vec{z}_{y,b} \geq \vec{0}, \qquad \vec{z}_{u,a} \geq \vec{0}, \qquad \vec{z}_{u,b} \geq \vec{0}.$$

Applying a Newton iteration to (3.3)–(3.9) gives, at each Newton step,

$$M\vec{s}_y + K^{\top}\vec{s}_p - \vec{s}_{z_{y,a}} + \vec{s}_{z_{y,b}} = \vec{y}_d - M\vec{y}^* - K^{\top}\vec{p}^* + \vec{z}^*_{y,a} - \vec{z}^*_{y,b}, \tag{3.10}$$

$$\beta M\vec{s}_u - M\vec{s}_p - \vec{s}_{z_{u,a}} + \vec{s}_{z_{u,b}} = -\beta M\vec{u}^* + M\vec{p}^* + \vec{z}^*_{u,a} - \vec{z}^*_{u,b}, \tag{3.11}$$

$$K\vec{s}_y - M\vec{s}_u = \vec{f} - K\vec{y}^* + M\vec{u}^*, \tag{3.12}$$

$$(\vec{y}^* - \vec{y}_a) \circ \vec{s}_{z_{y,a}} + \vec{z}^*_{y,a} \circ \vec{s}_y = \mu\vec{e} - (\vec{y}^* - \vec{y}_a) \circ \vec{z}^*_{y,a}, \tag{3.13}$$

$$(\vec{y}_b - \vec{y}^*) \circ \vec{s}_{z_{y,b}} - \vec{z}^*_{y,b} \circ \vec{s}_y = \mu\vec{e} - (\vec{y}_b - \vec{y}^*) \circ \vec{z}^*_{y,b}, \tag{3.14}$$

$$(\vec{u}^* - \vec{u}_a) \circ \vec{s}_{z_{u,a}} + \vec{z}^*_{u,a} \circ \vec{s}_u = \mu\vec{e} - (\vec{u}^* - \vec{u}_a) \circ \vec{z}^*_{u,a}, \tag{3.15}$$

$$(\vec{u}_b - \vec{u}^*) \circ \vec{s}_{z_{u,b}} - \vec{z}^*_{u,b} \circ \vec{s}_u = \mu\vec{e} - (\vec{u}_b - \vec{y}^*) \circ \vec{z}^*_{u,b}. \tag{3.16}$$

Here, $\vec{y}^*$, $\vec{u}^*$, $\vec{p}^*$, $\vec{z}^*_{y,a}$, $\vec{z}^*_{y,b}$, $\vec{z}^*_{u,a}$, $\vec{z}^*_{u,b}$ denote the most recent Newton iterates for $\vec{y}$, $\vec{u}$, $\vec{p}$, $\vec{z}_{y,a}$, $\vec{z}_{y,b}$, $\vec{z}_{u,a}$, $\vec{z}_{u,b}$, with $\vec{s}_y$, $\vec{s}_u$, $\vec{s}_p$, $\vec{s}_{z_{y,a}}$, $\vec{s}_{z_{y,b}}$, $\vec{s}_{z_{u,a}}$, $\vec{s}_{z_{u,b}}$ the Newton updates.

In matrix form, (3.10)–(3.16) read

$$\begin{bmatrix} M & 0 & K^\top & -I & I & 0 & 0 \\ 0 & \beta M & -M & 0 & 0 & -I & I \\ K & -M & 0 & 0 & 0 & 0 & 0 \\ Z_{y,a} & 0 & 0 & Y-Y_a & 0 & 0 & 0 \\ -Z_{y,b} & 0 & 0 & 0 & Y_b-Y & 0 & 0 \\ 0 & Z_{u,a} & 0 & 0 & 0 & U-U_a & 0 \\ 0 & -Z_{u,b} & 0 & 0 & 0 & 0 & U_b-U \end{bmatrix} \begin{bmatrix} \vec{s}_y \\ \vec{s}_u \\ \vec{s}_p \\ \vec{s}_{z_{y,a}} \\ \vec{s}_{z_{y,b}} \\ \vec{s}_{z_{u,a}} \\ \vec{s}_{z_{u,b}} \end{bmatrix}$$
$$= \begin{bmatrix} \vec{y}_d - M\vec{y}^* - K^\top \vec{p}^* + \vec{z}^*_{y,a} - \vec{z}^*_{y,b} \\ -\beta M\vec{u}^* + M\vec{p}^* + \vec{z}^*_{u,a} - \vec{z}^*_{u,b} \\ \vec{f} - K\vec{y}^* + M\vec{u}^* \\ \mu\vec{e} - (\vec{y}^* - \vec{y}_a) \circ \vec{z}^*_{y,a} \\ \mu\vec{e} - (\vec{y}_b - \vec{y}^*) \circ \vec{z}^*_{y,b} \\ \mu\vec{e} - (\vec{u}^* - \vec{u}_a) \circ \vec{z}^*_{u,a} \\ \mu\vec{e} - (\vec{u}_b - \vec{y}^*) \circ \vec{z}^*_{u,b} \end{bmatrix},$$

where Y, U, $Z_{y,a}$, $Z_{y,b}$, $Z_{u,a}$, $Z_{u,b}$ are diagonal matrices, with the most recent iterates for $\vec{y}$, $\vec{u}$, $\vec{z}_{y,a}$, $\vec{z}_{y,b}$, $\vec{z}_{u,a}$, $\vec{z}_{u,b}$ appearing on the diagonal entries. Similarly, the matrices Y_a, Y_b, U_a, U_b are diagonal matrices corresponding to $\vec{y}_a$, $\vec{y}_b$, $\vec{u}_a$, $\vec{u}_b$.

Now, we may write that fourth, fifth, sixth and seventh rows lead to

$$\vec{s}_{z_{y,a}} = -(Y-Y_a)^{-1} Z_{y,a}\vec{s}_y - Z_{y,a} + \mu(Y-Y_a)^{-1}\vec{e}, \tag{3.17}$$
$$\vec{s}_{z_{y,b}} = (Y_b-Y)^{-1} Z_{y,b}\vec{s}_y - Z_{y,b} + \mu(Y_b-Y)^{-1}\vec{e}, \tag{3.18}$$
$$\vec{s}_{z_{u,a}} = -(U-U_a)^{-1} Z_{u,a}\vec{s}_u - Z_{u,a} + \mu(U-U_a)^{-1}\vec{e}, \tag{3.19}$$
$$\vec{s}_{z_{u,b}} = (U_b-U)^{-1} Z_{u,b}\vec{s}_u - Z_{u,b} + \mu(U_b-U)^{-1}\vec{e}, \tag{3.20}$$

whereupon we may consider instead the solution of the reduced system

$$\begin{bmatrix} M+D_y & 0 & K^\top \\ 0 & \beta M + D_u & -M \\ K & -M & 0 \end{bmatrix} \begin{bmatrix} \vec{s}_y \\ \vec{s}_u \\ \vec{s}_p \end{bmatrix} \tag{3.21}$$
$$= \begin{bmatrix} \mu(Y-Y_a)^{-1}\vec{e} - \mu(Y_b-Y)^{-1}\vec{e} + \vec{y}_d - M\vec{y}^* - K^\top\vec{p}^* \\ \mu(U-U_a)^{-1}\vec{e} - \mu(U_b-U)^{-1}\vec{e} - \beta M\vec{u}^* + M\vec{p}^* \\ \vec{f} - K\vec{y}^* + M\vec{u}^* \end{bmatrix},$$

where

$$D_y = (Y-Y_a)^{-1} Z_{y,a} + (Y_b-Y)^{-1} Z_{y,b}, \tag{3.22}$$
$$D_u = (U-U_a)^{-1} Z_{u,a} + (U_b-U)^{-1} Z_{u,b}. \tag{3.23}$$

The conditions written in (3.21) are applied, alongside the imposition of (3.17)–(3.20), at each Newton iteration.

Note that, due to the fact that state and control bounds are enforced as strict inequalities at each Newton step, the diagonal matrices D_y and D_u are positive definite.

Of course, it is perfectly natural to consider a problem with only state constraints or only control constraints (or indeed only lower or upper bound constraints). For such cases we may follow exactly the same working to obtain a matrix system of the form (3.21), removing individual matrices corresponding to constraints that we do not apply.

3.2. Algorithm. We now present the structure of the interior point algorithm that we apply to the problems considered in this paper. The essence of the method is to traverse the interior of the feasible region where solutions may arise – we do this by applying a relaxed Newton iteration, reducing the barrier parameter by a factor σ at each Newton step. Having computed the Newton updates $\vec{s}_y,\ \vec{s}_u,\ \vec{s}_p$, $\vec{s}_{z_{y,a}}$, $\vec{s}_{z_{y,b}}$, $\vec{s}_{z_{u,a}}$, $\vec{s}_{z_{u,b}}$, we make a step in this direction that also guarantees that the strict bounds are enforced at each iteration. Upon convergence the iterates approach the true solution of the optimization problem, with the additional state and control constraints automatically satisfied.

Interior Point Method for Quadratic Programming

Parameters

$\alpha_0 = 0.995$, step-size factor to boundary

$\sigma \in (0,1)$, barrier reduction parameter

$\epsilon_p,\ \epsilon_d,\ \epsilon_c$, stopping tolerances,

Interior point method stops when $\left\|\vec{\xi}_p^{\,k}\right\| \le \epsilon_p,\ \left\|\vec{\xi}_d^{\,k}\right\| \le \epsilon_d,\ \left\|\vec{\xi}_c^{\,k}\right\| \le \epsilon_c$

Initialize IPM

Initial guesses for $\vec{y}^{\,0},\ \vec{u}^{\,0},\ \vec{p}^{\,0},\ \vec{z}_{y,a}^{\,0},\ \vec{z}_{y,b}^{\,0},\ \vec{z}_{u,a}^{\,0},\ \vec{z}_{u,b}^{\,0}$

Barrier parameter μ_0

Primal infeasibility $\vec{\xi}_p^{\,0} = \vec{f} - K\vec{y}^{\,0} + M\vec{u}^{\,0}$

Dual infeasibility $\vec{\xi}_d^{\,0} = \begin{bmatrix} \vec{y}_d - M\vec{y}^{\,0} - K^\top\vec{p}^{\,0} + \vec{z}_{y,a}^{\,0} - \vec{z}_{y,b}^{\,0} \\ -\beta M\vec{u}^{\,0} + M\vec{p}^{\,0} + \vec{z}_{u,a}^{\,0} - \vec{z}_{u,b}^{\,0} \end{bmatrix}$

Complementarity products $\vec{\xi}_c^{\,0}$, as in (3.24) with $k = 0$

Interior Point Method

while $\left(\left\|\vec{\xi}_p^{\,k}\right\| > \epsilon_p \ \text{ or } \ \left\|\vec{\xi}_d^{\,k}\right\| > \epsilon_d \ \text{ or } \ \left\|\vec{\xi}_c^{\,k}\right\| > \epsilon_c\right)$

Reduce barrier parameter $\mu_{k+1} = \sigma\mu_k$

Solve Newton system (3.21) for primal-dual Newton direction $\vec{s}_y,\ \vec{s}_u,\ \vec{s}_p$

Use (3.17)–(3.20) to find $\vec{s}_{z_{y,a}},\ \vec{s}_{z_{y,b}},\ \vec{s}_{z_{u,a}},\ \vec{s}_{z_{u,b}}$

Find $\alpha_P,\ \alpha_D$ s.t. bound constraints on primal and dual variables hold

Set $\alpha_P = \alpha_0\alpha_P,\ \alpha_D = \alpha_0\alpha_D$

Make step: $\vec{y}^{\,k+1} = \vec{y}^{\,k} + \alpha_P\vec{s}_y,\ \vec{u}^{\,k+1} = \vec{u}^{\,k} + \alpha_P\vec{s}_u,\ \vec{p}^{\,k+1} = \vec{p}^{\,k} + \alpha_D\vec{s}_p$

$\vec{z}_{y,a}^{\,k+1} = \vec{z}_{y,a}^{\,k} + \alpha_D\vec{s}_{z_{y,a}},\ \vec{z}_{y,b}^{\,k+1} = \vec{z}_{y,b}^{\,k} + \alpha_D\vec{s}_{z_{y,b}}$

$\vec{z}_{u,a}^{\,k+1} = \vec{z}_{u,a}^{\,k} + \alpha_D\vec{s}_{z_{u,a}},\ \vec{z}_{u,b}^{\,k+1} = \vec{z}_{u,b}^{\,k} + \alpha_D\vec{s}_{z_{u,b}}$

Update infeasibilities:

$\vec{\xi}_p^{\,k+1} = \vec{f} - K\vec{y}^{\,k+1} + M\vec{u}^{\,k+1},$

$\vec{\xi}_d^{\,k+1} = \begin{bmatrix} \vec{y}_d - M\vec{y}^{\,k+1} - K^\top\vec{p}^{\,k+1} + \vec{z}_{y,a}^{\,k+1} - \vec{z}_{y,b}^{\,k+1} \\ -\beta M\vec{u}^{\,k+1} + M\vec{p}^{\,k+1} + \vec{z}_{u,a}^{\,k+1} - \vec{z}_{u,b}^{\,k+1} \end{bmatrix}$

Compute error of complementarity products as in (3.24)

Set iteration number $k = k + 1$

end

Let us now consider appropriate stopping criteria for the method. Two natural requirements are for the norms of the primal and dual infeasibilities (at the k-th iteration)

$$\vec{\xi}_p^k = \vec{f} - K\vec{y}^{\,k} + M\vec{u}^{\,k}, \qquad \vec{\xi}_d^k = \begin{bmatrix} \vec{y}_d - M\vec{y}^{\,k} - K^\top \vec{p}^{\,k} + \vec{z}_{y,a}^k - \vec{z}_{y,b}^k \\ -\beta M\vec{u}^{\,k} + M\vec{p}^{\,k} + \vec{z}_{u,a}^k - \vec{z}_{u,b}^k \end{bmatrix}$$

to be lower than some prescribed tolerances ϵ_p, ϵ_d, respectively. Additionally, we require the error in the complementarity products

$$\vec{\xi}_c^k = \begin{bmatrix} \mu\vec{e} - (\vec{y}^{\,k} - \vec{y}_a) \circ \vec{z}_{y,a}^k \\ \mu\vec{e} - (\vec{y}_b - \vec{y}^{\,k}) \circ \vec{z}_{y,b}^k \\ \mu\vec{e} - (\vec{u}^{\,k} - \vec{u}_a) \circ \vec{z}_{u,a}^k \\ \mu\vec{e} - (\vec{u}_b - \vec{y}^{\,k}) \circ \vec{z}_{u,b}^k \end{bmatrix}, \tag{3.24}$$

to fall below some specified tolerance ϵ_c.

We present the algorithm that we apply – its structure is similar to the algorithm outlined in [10, Section 2].

It is clear from the presentation of this method that the dominant computational work arises from the solution of Newton system (3.21). It is therefore crucial to construct fast and robust solvers for this system, and this is what we focus on in Section 4.

3.3. Time-dependent problems. It is also important to be able to handle time-dependent problems using this methodology, due to the complexity and practical utility of such setups. To provide a brief illustration of how this may be accomplished, let us consider the time-dependent problem:

$$\begin{aligned} \min_{y,u} \quad & \frac{1}{2}\int_0^T \int_\Omega (y - \widehat{y})^2 \, \mathrm{d}\Omega \mathrm{d}t + \frac{\beta}{2}\int_0^T \int_\Omega u^2 \, \mathrm{d}\Omega \mathrm{d}t \\ \text{s.t.} \quad & y_t - \mathcal{L}y = u, \quad \text{in } \Omega \times [0,T], \\ & y = f, \quad \text{on } \partial\Omega \times [0,T], \\ & y = y_0, \quad \text{at } t = 0, \\ & y_a \le y \le y_b, \quad \text{a.e. in } \Omega \times [0,T], \\ & u_a \le u \le u_b, \quad \text{a.e. in } \Omega \times [0,T]. \end{aligned}$$

The state, control and adjoint variables are now solved in a space-time domain $\Omega \times [0,T]$, with $\mathcal{L}$ the time-independent component of the PDE operator.

As in [24, 31] for heat equation control problems, we may apply a discretize-then-optimize approach, using the trapezoidal rule in space, and the backward Euler method to account for the time derivative. We thus rewrite the problem in the discrete setting as follows:

$$\begin{aligned} \min_{y,u} \quad & \frac{\tau}{2}\vec{y}^{\,\top}\mathcal{M}_{1/2}\vec{y} - \tau\vec{y}_{d,T}^{\,\top}\vec{y} + \frac{\beta\tau}{2}\vec{u}^{\,\top}\mathcal{M}_{1/2}\vec{u} \\ \text{s.t.} \quad & \mathcal{K}\vec{y} - \mathcal{M}\vec{u} = \vec{f}_T, \\ & \vec{y}_a \le \vec{y} \le \vec{y}_b, \\ & \vec{u}_a \le \vec{u} \le \vec{u}_b. \end{aligned}$$

Here the matrix $\mathcal{M}_{1/2} = \text{blkdiag}(\frac{1}{2}M, M, ..., M, \frac{1}{2}M)$, $\mathcal{M} = \text{blkdiag}(M, ..., M)$, and

$$\mathcal{K} = \begin{bmatrix} M+\tau K & & & & & \\ -M & M+\tau K & & & & \\ & \ddots & \ddots & & & \\ & & -M & M+\tau K & & \\ & & & -M & M+\tau K \end{bmatrix},$$

$$\vec{y}_{d,T} = \begin{bmatrix} \frac{1}{2}\vec{y}_{d,1} \\ \vec{y}_{d,2} \\ \vdots \\ \vec{y}_{d,N_t-1} \\ \frac{1}{2}\vec{y}_{d,N_t} \end{bmatrix}, \qquad \vec{f}_T = \begin{bmatrix} M\vec{y}_0 + \vec{f} \\ \vec{f} \\ \vdots \\ \vec{f} \\ \vec{f} \end{bmatrix},$$

where K corresponds to the time-independent part of the PDE operator, and τ denotes the (constant) time-step taken. The vectors $\vec{y}_{d,i}$ relate to the values of $\widehat{y}$ at the i-th time-step, and $\vec{y}_0$ is the vector representation of y_0. We denote by $N_t := \frac{T}{\tau}$ the number of time-steps taken.

We apply Newton iteration to the discrete optimality conditions, in an analogous way to the time-independent problem. This yields the matrix system

$$\begin{bmatrix} \tau\mathcal{M}_{1/2} & 0 & \mathcal{K}^\top & -I & I & 0 & 0 \\ 0 & \beta\tau\mathcal{M}_{1/2} & -\tau\mathcal{M} & 0 & 0 & -I & I \\ \mathcal{K} & -\tau\mathcal{M} & 0 & 0 & 0 & 0 & 0 \\ Z_{y,a} & 0 & 0 & Y-Y_a & 0 & 0 & 0 \\ -Z_{y,b} & 0 & 0 & 0 & Y_b-Y & 0 & 0 \\ 0 & Z_{u,a} & 0 & 0 & 0 & U-U_a & 0 \\ 0 & -Z_{u,b} & 0 & 0 & 0 & 0 & U_b-U \end{bmatrix} \begin{bmatrix} \vec{s}_y \\ \vec{s}_u \\ \vec{s}_p \\ \vec{s}_{z_{y,a}} \\ \vec{s}_{z_{y,b}} \\ \vec{s}_{z_{u,a}} \\ \vec{s}_{z_{u,b}} \end{bmatrix} \tag{3.25}$$

$$= \begin{bmatrix} \tau\vec{y}_{d,T} - \tau\mathcal{M}_{1/2}\vec{y}^* - \mathcal{K}^\top\vec{p}^* + \vec{z}^*_{y,a} - \vec{z}^*_{y,b} \\ -\beta\tau\mathcal{M}_{1/2}\vec{u}^* + \tau\mathcal{M}\vec{p}^* + \vec{z}^*_{u,a} - \vec{z}^*_{u,b} \\ \vec{f}_T - \mathcal{K}\vec{y}^* + \tau\mathcal{M}\vec{u}^* \\ \mu\vec{e} - (\vec{y}^* - \vec{y}_a)\circ\vec{z}^*_{y,a} \\ \mu\vec{e} - (\vec{y}_b - \vec{y}^*)\circ\vec{z}^*_{y,b} \\ \mu\vec{e} - (\vec{u}^* - \vec{u}_a)\circ\vec{z}^*_{u,a} \\ \mu\vec{e} - (\vec{u}_b - \vec{y}^*)\circ\vec{z}^*_{u,b} \end{bmatrix},$$

with $\vec{z}_{y_a}$, $\vec{z}_{y_b}$, $\vec{z}_{u_a}$, $\vec{z}_{u_b}$ the same as for the time-independent setting, except now measured over all points in space and time.

Reducing (3.25) as for the time-independent case gives a block matrix system

$$\begin{bmatrix} \tau\mathcal{M}_{1/2} + \mathcal{D}_y & 0 & \mathcal{K}^\top \\ 0 & \beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u & -\tau\mathcal{M} \\ \mathcal{K} & -\tau\mathcal{M} & 0 \end{bmatrix} \begin{bmatrix} \vec{s}_y \\ \vec{s}_u \\ \vec{s}_p \end{bmatrix} \tag{3.26}$$

$$= \begin{bmatrix} \mu(Y-Y_a)^{-1}\vec{e} - \mu(Y_b-Y)^{-1}\vec{e} + \vec{y}_{d,T} - \tau\mathcal{M}_{1/2}\vec{y}^* - \mathcal{K}^\top\vec{p}^* \\ \mu(U-U_a)^{-1}\vec{e} - \mu(U_b-U)^{-1}\vec{e} - \beta\tau\mathcal{M}_{1/2}\vec{u}^* + \tau\mathcal{M}\vec{p}^* \\ \vec{f}_T - \mathcal{K}\vec{y}^* + \tau\mathcal{M}\vec{u}^* \end{bmatrix},$$

with $\mathcal{D}_y$, $\mathcal{D}_u$ analogous to D_y, D_u, as defined in (3.22), (3.23), except with the quantities measured within the entire space-time domain.

4. Preconditioning for the Newton system. For the matrix systems considered in this paper, particularly those arising from time-dependent problems, great care must be taken when seeking an appropriate scheme for obtaining an accurate solution. The dimensions of these systems mean that a direct method is often infeasible, so we find that the natural approach is to develop preconditioned Krylov subspace solvers.

When seeking preconditioners for such methods, we exploit the fact that the matrix systems for the PDE-constrained optimization problems are of *saddle point form*:

$$\underbrace{\begin{bmatrix} \Phi & \Psi^\top \\ \Psi & \Theta \end{bmatrix}}_{\mathcal{A}} \begin{bmatrix} \vec{x}_1 \\ \vec{x}_2 \end{bmatrix} = \begin{bmatrix} \vec{b}_1 \\ \vec{b}_2 \end{bmatrix}. \tag{4.1}$$

Here $\Phi \in \mathbb{R}^{n\times n}$, $\Psi \in \mathbb{R}^{m\times n}$ and $\Theta \in \mathbb{R}^{m\times m}$ (with $m \leq n$, as in Section 2). Further Φ and Θ are symmetric matrices, meaning that $\mathcal{A}$ is itself symmetric, and all of the matrices are sparse for the finite element method used. We recommend [1] for a thorough overview of saddle point systems and their numerical properties.

The study of preconditioners for systems of this form is a well-established subject area: indeed it is known that two 'ideal' preconditioners are given by

$$\mathcal{P}_D = \begin{bmatrix} \Phi & 0 \\ 0 & S \end{bmatrix}, \qquad \mathcal{P}_T = \begin{bmatrix} \Phi & 0 \\ \Psi & -S \end{bmatrix},$$

where $S := -\Theta + \Psi\Phi^{-1}\Psi^T$ defines the (negative) *Schur complement* of $\mathcal{A}$. It can be shown [16, 18] that the eigenvalues of the preconditioned systems are given by

$$\begin{aligned} \lambda\left(\mathcal{P}_D^{-1}\mathcal{A}\right) &\in \left\{1, \frac{1}{2}(1 \pm \sqrt{5})\right\}, && \text{if } \Theta = 0, \\ \lambda\left(\mathcal{P}_T^{-1}\mathcal{A}\right) &\in \{1\}, && \text{generally,} \end{aligned}$$

provided that these systems are invertible.

In practice, of course, one would not wish to invert Φ and S exactly within a preconditioner, so the main challenge is to devise effective approximations $\widehat{\Phi}$ and $\widehat{S}$ which can be applied within a block diagonal or block triangular preconditioner of the form

$$\mathcal{P} = \begin{bmatrix} \widehat{\Phi} & 0 \\ 0 & \widehat{S} \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} \widehat{\Phi} & 0 \\ \Psi & -\widehat{S} \end{bmatrix}. \tag{4.2}$$

Such preconditioners are very often found to be extremely potent in practice, and in many cases one can prove their effectiveness as well (we discuss this further in Section 4.1).

A major objective within the remainder of this paper is to develop effective representations of the $(1,1)$-block Φ and Schur complement S for matrix systems arising from interior point solvers.

4.1. Time-independent problems. We now wish to apply saddle point theory to matrix systems arising from time-independent problems. So consider the matrix system (3.21), in the case where the matrix K arises from a Laplacian operator (considered for Poisson control) or convection-diffusion operator. This system is of saddle

point form (4.1), with

$$\Phi = \begin{bmatrix} M + D_y & 0 \\ 0 & \beta M + D_u \end{bmatrix}, \qquad \Psi = \begin{bmatrix} K & -M \end{bmatrix}, \qquad \Theta = \begin{bmatrix} 0 \end{bmatrix}.$$

Let us consider approximating the $(1,1)$-block and Schur complement of this matrix system. For this problem M is a positive definite matrix, with positive diagonal entries, and the same applies to K in the case of Poisson control problems.

We now highlight that mass matrices may in fact be well approximated by their diagonal: for instance, in the case of $Q1$ mass matrices on a uniform two dimensional domain, the eigenvalues of $[\text{diag}(M)]^{-1}M$ are all contained within the interval $[\frac{1}{4}, \frac{9}{4}]$ (see [34]). As D_y and D_u are diagonal and positive definite, one option for approximating Φ is hence to take

$$\widehat{\Phi} = \begin{bmatrix} \text{diag}\,(M + D_y) & 0 \\ 0 & \text{diag}\,(\beta M + D_u) \end{bmatrix}.$$

The effectiveness of the approximation may be measured in some sense by the eigenvalues of $\widehat{\Phi}^{-1}\Phi$, which may themselves be determined by the Rayleigh quotient

$$\begin{aligned} \frac{\vec{v}^\top \Phi \vec{v}}{\vec{v}^\top \widehat{\Phi} \vec{v}} &= \frac{\vec{v}_1^\top (M + D_y)\vec{v}_1 + \vec{v}_2^\top (\beta M + D_u)\vec{v}_2}{\vec{v}_1^\top \big[\text{diag}(M + D_y)\big]\vec{v}_1 + \vec{v}_2^\top \big[\text{diag}(\beta M + D_u)\big]\vec{v}_2} \\ &= \frac{\vec{v}_1^\top M \vec{v}_1 + \beta \vec{v}_2^\top M \vec{v}_2 + \vec{v}_1^\top D_y \vec{v}_1 + \vec{v}_2^\top D_u \vec{v}_2}{\vec{v}_1^\top \big[\text{diag}(M)\big]\vec{v}_1 + \beta \vec{v}_2^\top \big[\text{diag}(M)\big]\vec{v}_2 + \vec{v}_1^\top D_y \vec{v}_1 + \vec{v}_2^\top D_u \vec{v}_2} \\ &\in \left[\min\left\{ \frac{\vec{v}_1^\top M \vec{v}_1 + \beta \vec{v}_2^\top M \vec{v}_2}{\vec{v}_1^\top \big[\text{diag}(M)\big]\vec{v}_1 + \beta \vec{v}_2^\top \big[\text{diag}(M)\big]\vec{v}_2}, 1 \right\}, \right. \qquad (4.3) \\ &\qquad \left. \max\left\{ \frac{\vec{v}_1^\top M \vec{v}_1 + \beta \vec{v}_2^\top M \vec{v}_2}{\vec{v}_1^\top \big[\text{diag}(M)\big]\vec{v}_1 + \beta \vec{v}_2^\top \big[\text{diag}(M)\big]\vec{v}_2}, 1 \right\} \right] \\ &\in \left[\min\left\{ \lambda_{\min}\left(\big[\text{diag}(M)\big]^{-1} M \right), 1 \right\}, \max\left\{ \lambda_{\max}\left(\big[\text{diag}(M)\big]^{-1} M \right), 1 \right\} \right], \end{aligned}$$

where (4.3) follows from the fact that $\vec{v}_1^\top D_y \vec{v}_1 + \vec{v}_2^\top D_u \vec{v}_2$ is non-negative. Here $\vec{v} = \begin{bmatrix} \vec{v}_1^\top, & \vec{v}_2^\top \end{bmatrix}^\top \neq \vec{0}$, with $\vec{v}_1$, $\vec{v}_2$ vectors of appropriate length, and $\lambda_{\min}$, $\lambda_{\max}$ denote the smallest and largest eigenvalues of a matrix. We therefore see that if $[\text{diag}(M)]^{-1}M$ is well-conditioned, then the same is true of $\widehat{\Phi}^{-1}\Phi$.

As an alternative for our approximation $\widehat{\Phi}$, one may apply a multigrid method to approximate the inverses of $M + D_y$ and $\beta M + D_u$ – this is however a more expensive process to approximate a relatively simple component of the entire system (in general the matrices with the most complex structure are K and $K^\top$).

The main task at this stage is to approximate the Schur complement

$$S = K(M + D_y)^{-1}K^\top + M(\beta M + D_u)^{-1}M.$$

The aim is to build an approximation such that the eigenvalues of the preconditioned Schur complement are tightly clustered. We motivate our approximation based on a 'matching' strategy originally derived in [26] for the Poisson control problem without bound constraints: for this particular problem it was shown that by 'capturing' both terms of the Schur complement, one obtains the result

$$\lambda\left(\left[\left(K + \frac{1}{\sqrt{\beta}} M \right) M^{-1} \left(K + \frac{1}{\sqrt{\beta}} M \right) \right]^{-1} \left[K M^{-1} K + \frac{1}{\beta} M \right] \right) \in \left[\frac{1}{2}, 1 \right],$$

independently of problem size, as well as the value of β. Motivated by this strategy, we may consider an approximation for S of the form

$$\widehat{S}_1 := \big(K + \widehat{M}_1\big)(M + D_y)^{-1}\big(K + \widehat{M}_1\big)^\top,$$

where $\widehat{M}_1$ is chosen to incorporate the second term of the exact Schur complement, that is:

$$\widehat{M}_1(M + D_y)^{-1}\widehat{M}_1^\top \approx M(\beta M + D_u)^{-1}M.$$

This leads to the following requirement when selecting $\widehat{M}_1$:

$$\widehat{M}_1 \approx M(\beta M + D_u)^{-1/2}(M + D_y)^{1/2}.$$

We take diagonal approximations where appropriate, in order to avoid having to construct square roots of matrices, which would be extremely expensive computationally. That is, we take

$$\widehat{M}_1 = M\big[\mathrm{diag}(\beta M + D_u)\big]^{-1/2}\big[\mathrm{diag}(M + D_y)\big]^{1/2}.$$

In practice, one may often approximate the inverses of $K + \widehat{M}_1$ and its transpose effectively using a multigrid process. We apply the Aggregation-based Algebraic Multigrid (AGMG) software [19, 20, 21, 22] for this purpose within our iterative solvers.

It is possible to prove a lower bound of the preconditioned Schur complement for a very general matrix form, as demonstrated below.

THEOREM 4.1. *Let S_G and $\widehat{S}_G$ be the general matrices*

$$S_G = XX^\top + YY^\top, \qquad \widehat{S}_G = (X + Y)(X + Y)^\top,$$

which we assume to be invertible, and with real X, Y.[1] *Then the eigenvalues of $\widehat{S}_G^{-1}S_G$ are real, and satisfy $\lambda \geq \frac{1}{2}$.*

Proof. As S_G and $\widehat{S}_G$ are invertible, they are symmetric positive definite by constuction. To examine the spectrum of $\widehat{S}_G^{-1}S_G$ we therefore consider the Rayleigh quotient (for real $\vec{v} \neq \vec{0}$):

$$R := \frac{\vec{v}^\top S_G \vec{v}}{\vec{v}^\top \widehat{S}_G \vec{v}} = \frac{\vec{\chi}^\top\vec{\chi} + \vec{\omega}^\top\vec{\omega}}{(\vec{\chi} + \vec{\omega})^\top(\vec{\chi} + \vec{\omega})}, \qquad \vec{\chi} = X^\top\vec{v}, \quad \vec{\omega} = Y^\top\vec{v},$$

which is itself clearly real. By the invertibility of S_G and $\widehat{S}_G$, both numerator and denominator are positive. Therefore

$$\frac{1}{2}(\vec{\chi} - \vec{\omega})^\top(\vec{\chi} - \vec{\omega}) \geq 0 \quad \Leftrightarrow \quad \vec{\chi}^\top\vec{\chi} + \vec{\omega}^\top\vec{\omega} \geq \frac{1}{2}(\vec{\chi} + \vec{\omega})^\top(\vec{\chi} + \vec{\omega}) \quad \Leftrightarrow \quad R \geq \frac{1}{2}.$$

□

Note that to demonstrate an upper bound, one would write

$$R = \left(1 + \frac{2\vec{\chi}^\top\vec{\omega}}{\vec{\chi}^\top\vec{\chi} + \vec{\omega}^\top\vec{\omega}}\right)^{-1}.$$

[1] For the example considered here, $X = K(M + D_y)^{-1/2}$ and $Y = M(\beta M + D_u)^{-1/2}$.

This quantity is certainly finite, as $\frac{2\vec{\chi}^\top\vec{\omega}}{\vec{\chi}^\top\vec{\chi}+\vec{\omega}^\top\vec{\omega}} > -1$ by simple manipulation (the case $\frac{2\vec{\chi}^\top\vec{\omega}}{\vec{\chi}^\top\vec{\chi}+\vec{\omega}^\top\vec{\omega}} = -1$ is disallowed by the assumption of invertibility of $\widehat{S}_G$). It is generally not possible to demonstrate an upper bound unless X and Y have structures which can be exploited. However, using the methodology of Theorem 4.1, results of this form have been demonstrated for problems such as convection-diffusion control [25] and heat equation control [24] (without additional bound constraints). In general, the eigenvalues of $\widehat{S}_G^{-1}S_G$ are better clustered if the term $XY^\top + YX^\top$ is positive semi-definite, or 'nearly' positive semi-definite. The worst case would arise in the setting where $\vec{\chi} \approx -\vec{\omega}$, however for our problem the matrices X and Y do not relate closely to each other as the activities in the state and control variables do not share many common features.

The 'matching strategy' presented here guarantees a lower bound for the preconditioned Schur complement of matrices of this form, provided some very weak assumptions hold,[2] and often results in the largest eigenvalue being of moderate magnitude. We therefore wish to make use of this approach to generate effective Schur complement approximations for the matrix systems considered in this manuscript.

Combining our approximations of Φ and S, we propose the following block diagonal preconditioner of the form (4.2):

$$\mathcal{P}_1 = \begin{bmatrix} (M+D_y)_{\text{approx}} & 0 & 0 \\ 0 & (\beta M + D_u)_{\text{approx}} & 0 \\ 0 & 0 & \widehat{S}_1 \end{bmatrix},$$

where $(M+D_y)_{\text{approx}}$, $(\beta M+D_u)_{\text{approx}}$ indicate our choice of approximations for $M+D_y$, $\beta M+D_u$ (i.e. diagonal approximation or multigrid process). This preconditioner is symmetric positive definite, and may thus be applied within a symmetric solver such as MINRES [23].

It is useful to consider the distribution of eigenvalues of the preconditioned system, as this will control the convergence properties of the MINRES method. The fundamental result we use for our analysis is stated below [28].

THEOREM 4.2. *If Φ is symmetric positive definite, Ψ is full rank, and $\Theta = 0$, the eigenvalues of $\mathcal{A}$ are contained within the following intervals:*

$$\lambda(\mathcal{A}) \in \left[\frac{1}{2}\left(\mu_{\min} - \sqrt{\mu_{\min}^2 + 4\sigma_{\max}^2}\right), \frac{1}{2}\left(\mu_{\max} - \sqrt{\mu_{\max}^2 + 4\sigma_{\min}^2}\right)\right] \cup \left[\mu_{\min}, \frac{1}{2}\left(\mu_{\max} + \sqrt{\mu_{\max}^2 + 4\sigma_{\max}^2}\right)\right],$$

where $\mu_{\max}$, $\mu_{\min}$ denote the largest and smallest eigenvalues of Φ, with $\sigma_{\max}$, $\sigma_{\min}$ the largest and smallest singular values of Ψ.

We now wish to apply a result of this form to the preconditioned system. The preconditioned matrix, when a general block diagonal preconditioner of the form (4.2) is used, is given by

$$\mathcal{P}^{-1}\mathcal{A} = \begin{bmatrix} \widehat{\Phi} & 0 \\ 0 & \widehat{S} \end{bmatrix}^{-1} \begin{bmatrix} \Phi & \Psi^\top \\ \Psi & 0 \end{bmatrix} = \begin{bmatrix} \widehat{\Phi}^{-1}\Phi & \widehat{\Phi}^{-1}\Psi^\top \\ \widehat{S}^{-1}\Psi & 0 \end{bmatrix}.$$

[2]The main assumption made is that $\widehat{S}_G$ is invertible. This certainly holds unless $(X+Y)\vec{v} = \vec{0}$ for some $\vec{v}$, which in our setting implies that $(M+D_y)^{1/2}K^{-1}M(\beta M+D_u)^{-1/2}$ has an eigenvalue exactly equal to -1. As the matrices M, D_y, D_u and K are unlikely to interact closely at any Newton step, this is extremely unlikely to occur and our assumption is therefore reasonable.

Now, to analyse the properties of this system, let

$$\lambda(\widehat{\Phi}^{-1}\Phi) \in [\phi_{\min}, \phi_{\max}], \qquad \lambda(\widehat{S}^{-1}S) \in [s_{\min}, s_{\max}],$$

where $\phi_{\min}, s_{\min} > 0$.

By the similarity property of matrix systems (using that for our problem $\widehat{\Phi}$ and $\widehat{S}$ are positive definite) the eigenvalues will be the same as those of

$$\begin{aligned}\mathcal{P}^{-1/2}\mathcal{A}\mathcal{P}^{-1/2} &= \begin{bmatrix} \widehat{\Phi}^{-1/2} & 0 \\ 0 & \widehat{S}^{-1/2} \end{bmatrix} \begin{bmatrix} \Phi & \Psi^\top \\ \Psi & 0 \end{bmatrix} \begin{bmatrix} \widehat{\Phi}^{-1/2} & 0 \\ 0 & \widehat{S}^{-1/2} \end{bmatrix} \\ &= \begin{bmatrix} \widehat{\Phi}^{-1/2}\Phi\widehat{\Phi}^{-1/2} & \widehat{\Phi}^{-1/2}\Psi^\top\widehat{S}^{-1/2} \\ \widehat{S}^{-1/2}\Psi\widehat{\Phi}^{-1/2} & 0 \end{bmatrix}.\end{aligned}$$

The eigenvalues of the $(1,1)$-block of this matrix, $\widehat{\Phi}^{-1/2}\Phi\widehat{\Phi}^{-1/2}$, are the same as those of $\widehat{\Phi}^{-1}\Phi$ by similarity, and so are contained in $[\phi_{\min}, \phi_{\max}]$. The singular values of the $(2,1)$-block are given by the square roots of the eigenvalues of $\widehat{S}^{-1/2}\Psi\widehat{\Phi}^{-1}\Psi^\top\widehat{S}^{-1/2}$, i.e. the square roots of the eigenvalues of $\widehat{S}^{-1}(\Psi\widehat{\Phi}^{-1}\Psi^\top)$ by similarity. Writing the Rayleigh quotient (for $\vec{v} \neq \vec{0}$),

$$\frac{\vec{v}^\top\Psi\widehat{\Phi}^{-1}\Psi^\top\vec{v}}{\vec{v}^\top\widehat{S}\vec{v}} = \frac{\vec{v}^\top\Psi\widehat{\Phi}^{-1}\Psi^\top\vec{v}}{\vec{v}^\top\Psi\Phi^{-1}\Psi^\top\vec{v}} \cdot \frac{\vec{v}^\top\Psi\Phi^{-1}\Psi^\top\vec{v}}{\vec{v}^\top\widehat{S}\vec{v}} = \underbrace{\frac{\vec{w}^\top\widehat{\Phi}^{-1}\vec{w}}{\vec{w}^\top\Phi^{-1}\vec{w}}}_{\in[\phi_{\min},\phi_{\max}]} \cdot \underbrace{\frac{\vec{v}^\top\Psi\Phi^{-1}\Psi^\top\vec{v}}{\vec{v}^\top\widehat{S}\vec{v}}}_{\in[s_{\min},s_{\max}]},$$

where $\vec{w} = \Psi^\top\vec{v}$, enables us to pin the singular values of the $(2,1)$-block within $[\sqrt{\phi_{\min}s_{\min}}, \sqrt{\phi_{\max}s_{\max}}]$.

So, using Theorem 4.2, the eigenvalues of $\mathcal{P}^{-1}\mathcal{A}$ are contained within the interval stated below.

LEMMA 4.3. *If Φ and S are symmetric positive definite, and the above bounds on $\lambda(\widehat{\Phi}^{-1}\Phi)$ and $\lambda(\widehat{S}^{-1}S)$ hold, then the eigenvalues of $\mathcal{P}^{-1}\mathcal{A}$ satisfy*

$$\begin{aligned}\lambda(\mathcal{P}^{-1}\mathcal{A}) \in & \left[\frac{1}{2}\left(\phi_{\min} - \sqrt{\phi_{\min}^2 + 4\phi_{\max}s_{\max}}\right), \frac{1}{2}\left(\phi_{\max} - \sqrt{\phi_{\max}^2 + 4\phi_{\min}s_{\min}}\right)\right], \\ & \cup \left[\phi_{\min}, \frac{1}{2}\left(\phi_{\max} + \sqrt{\phi_{\max}^2 + 4\phi_{\max}s_{\max}}\right)\right].\end{aligned}$$

It is therefore clear that, for our problem, a good approximation of the Schur complement will guarantee clustered eigenvalues of the preconditioned system, and therefore rapid convergence of the MINRES method. As we have observed for our problem, the quantities of interest are therefore the largest eigenvalues of $\widehat{S}^{-1}S$, which can vary at every step of a Newton method.

We now present a straightforward result concerning the eigenvectors of a preconditioned saddle point system of the form under consideration.

PROPOSITION 4.4. *Consider an eigenvalue λ that satisfies*

$$\begin{bmatrix} \Phi & \Psi^\top \\ \Psi & 0 \end{bmatrix} \begin{bmatrix} \vec{v}_1 \\ \vec{v}_2 \end{bmatrix} = \lambda \begin{bmatrix} \widehat{\Phi} & 0 \\ 0 & \widehat{S} \end{bmatrix} \begin{bmatrix} \vec{v}_1 \\ \vec{v}_2 \end{bmatrix}, \tag{4.4}$$

with Φ, $S = \Psi\Phi^{-1}\Psi^\top$, $\widehat{\Phi}$, $\widehat{S}$ symmetric positive definite. Then either λ is an eigenvalue of $\widehat{\Phi}^{-1}\Phi$, or λ, $\vec{v}_1$ and $\vec{v}_2$ satisfy

$$\left(\lambda\widehat{\Phi} - \Phi - \frac{1}{\lambda}\Psi^\top\widehat{S}^{-1}\Psi\right)\vec{v}_1 = \vec{0}, \qquad \vec{v}_2 = \frac{1}{\lambda}\widehat{S}^{-1}\Psi\vec{v}_1.$$

Proof. Equation (4.4) is equivalent to

$$\Psi^\top\vec{v}_2 = \big(\lambda\widehat{\Phi} - \Phi\big)\vec{v}_1, \tag{4.5}$$

$$\Psi\vec{v}_1 = \lambda\widehat{S}\vec{v}_2. \tag{4.6}$$

Let us first consider the case where $\Psi\vec{v}_1 = \vec{0}$ (there are at most $n - m$ such linearly independent vectors that correspond to eigenvectors). Then (4.6) tells us that $\vec{v}_2 = \vec{0}$, from which we conclude from (4.5) that $(\lambda\widehat{\Phi} - \Phi)\vec{v}_1 = \vec{0}$. Therefore, in this case, the eigenvalues are given by eigenvalues of $\widehat{\Phi}^{-1}\Phi$, with eigenvectors of the form $[\vec{v}_1^\top,\ \vec{0}^\top]^\top$ – there are at most $n - m$ such solutions.

If $\Psi\vec{v}_1 \neq \vec{0}$, we may rearrange (4.6) to obtain

$$\vec{v}_2 = \frac{1}{\lambda}\widehat{S}^{-1}\Psi\vec{v}_1 \quad \Rightarrow \quad \Psi^\top\vec{v}_2 = \frac{1}{\lambda}\Psi^\top\widehat{S}^{-1}\Psi\vec{v}_1,$$

which we may substitute into (4.5) to obtain

$$\frac{1}{\lambda}\Psi^\top\widehat{S}^{-1}\Psi\vec{v}_1 = \big(\lambda\widehat{\Phi} - \Phi\big)\vec{v}_1.$$

This may be trivially rearranged to obtain the required result. ∎

We observe that the eigenvalues and eigenvectors of the $(1,1)$-block and Schur complement (along with their approximations) interact strongly with each other. This decreases the likelihood of many extreme eigenvalues of $\widehat{S}^{-1}S$ arising in practice, as this would have implications on the numerical properties of Φ and Ψ (which for our problems do not interact at all strongly). However the working provided here shows that this is very difficult to prove rigorously, due to the wide generality of the saddle point systems being examined – we must also rely on the physical properties of the PDE operators within the optimization framework. Our numerical experiments of Section 5 indicate that the eigenvalues of $\widehat{S}^{-1}S$, and therefore the preconditioned system, are tightly clustered, matching some of the observations made in this section.

As an alternative to the block diagonal preconditioner $\mathcal{P}_1$, we may take account of information on the block lower triangular parts of the matrix system, and apply the block triangular preconditioner

$$\mathcal{P}_2 = \begin{bmatrix} (M + D_y)_{\text{approx}} & 0 & 0 \\ 0 & (\beta M + D_u)_{\text{approx}} & 0 \\ K & -M & -\widehat{S}_1 \end{bmatrix},$$

within a non-symmetric solver such as GMRES [29].

It is possible to carry out eigenvalue analysis for the block triangular preconditioner $\mathcal{P}_2$ in the same way as for the block diagonal preconditioner $\mathcal{P}_1$. However it is well known that the convergence of non-symmetric solvers such as GMRES do not solely depend on the eigenvalues of the preconditioned system, and therefore such an analysis would be less useful in practice.

We now consider a completely different strategy for preconditioning the matrix system. We may first rearrange (3.21) to the form

$$\begin{bmatrix} \beta M + D_u & -M & 0 \\ -M & 0 & K \\ 0 & K^\top & M + D_y \end{bmatrix} \begin{bmatrix} \vec{s}_u \\ \vec{s}_p \\ \vec{s}_y \end{bmatrix} \tag{4.7}$$
$$= \begin{bmatrix} \mu(U - U_a)^{-1}\vec{e} - \mu(U_b - U)^{-1}\vec{e} - \beta M\vec{u}^* + M\vec{p}^* \\ \vec{f} - K\vec{y}^* + M\vec{u}^* \\ \mu(Y - Y_a)^{-1}\vec{e} - \mu(Y_b - Y)^{-1}\vec{e} + \vec{y}_d - M\vec{y}^* - K^\top\vec{p}^* \end{bmatrix}.$$

The matrix within (4.7) is a saddle point system of the form (4.1), with

$$\Phi = \begin{bmatrix} \beta M + D_u & -M \\ -M & 0 \end{bmatrix}, \qquad \Psi = \begin{bmatrix} 0 & K^\top \end{bmatrix}, \qquad \Theta = \begin{bmatrix} M + D_y \end{bmatrix}.$$

This approach has the advantage that the $(1,1)$-block Φ can be inverted almost precisely, as all that is required is a method for approximating the inverse of a mass matrix (to be applied twice). For this, a very cheap and accurate method is Chebyshev semi-iteration [8, 9, 35], so we apply this strategy within our preconditioner.

Once again, the main challenge is to approximate the Schur complement:

$$\begin{aligned} S = & -(M + D_y) + \begin{bmatrix} 0 & K^\top \end{bmatrix} \begin{bmatrix} \beta M + D_u & -M \\ -M & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ K \end{bmatrix} \\ = & -(M + D_y) + \begin{bmatrix} 0 & K^\top \end{bmatrix} \begin{bmatrix} 0 & -M^{-1} \\ -M^{-1} & -M^{-1}(\beta M + D_u)M^{-1} \end{bmatrix} \begin{bmatrix} 0 \\ K \end{bmatrix} \\ = & -\Big[K^\top M^{-1}(\beta M + D_u)M^{-1}K + (M + D_y)\Big]. \end{aligned}$$

Let us consider a 'matching' strategy once again, and write for our approximation:

$$\widehat{S}_2 := -\big(K^\top + \widehat{M}_2\big)M^{-1}(\beta M + D_u)M^{-1}\big(K + \widehat{M}_2^\top\big),$$

where $\widehat{M}_2$ is selected to incorporate the second term of S, i.e.

$$\widehat{M}_2 M^{-1}(\beta M + D_u)M^{-1}\widehat{M}_2^\top \approx M + D_y,$$

which may be achieved if

$$\widehat{M}_2 \approx (M + D_y)^{1/2}(\beta M + D_u)^{-1/2}M.$$

For a practical preconditioner, we in fact select

$$\widehat{M}_2 = \big[\mathrm{diag}(M + D_y)\big]^{1/2}\big[\mathrm{diag}(\beta M + D_u)\big]^{-1/2}M.$$

To approximate $K^\top + \widehat{M}_2$ and $K + \widehat{M}_2^\top$ in practice, we again make use of the AGMG software to apply a multigrid process to the relevant matrices within $\widehat{S}_2$.

Rearranging the matrix system (and hence the preconditioner) to the form (3.21), we are therefore able to construct the following preconditioner:

$$\mathcal{P}_3 = \begin{bmatrix} -\widehat{S}_2 & 0 & K^\top \\ 0 & \beta M + D_u & -M_{\mathrm{cheb}} \\ 0 & -M_{\mathrm{cheb}} & 0 \end{bmatrix},$$

where M_{cheb} relates to a Chebyshev semi-iteration process for the mass matrix M. We observe that this relates to observations made on nullspace preconditioners for saddle point systems in [27].

It is clear that to apply the preconditioner $\mathcal{P}_3$, we require a non-symmetric solver such as GMRES, as it is not possible to construct a positive definite preconditioner with this rearrangement of the matrix system. Within such a solver, the key advantage of this stategy is that we may approximate Φ almost perfectly (and cheaply). The associated disadvantage is that our approximation of S is unlikely to be as effective or cheap as the approximation $\widehat{S}_1$ used within the preconditioners $\mathcal{P}_1$ and $\mathcal{P}_2$.

4.2. Time-dependent problems. Due to the huge dimension of the matrix systems arising from time-dependent PDE-constrained optimization problems, it is very important to consider saddle point preconditioners from the resulting systems, which are of the form (3.26). These are again of saddle point type (4.1), with

$$\Phi = \begin{bmatrix} \tau\mathcal{M}_{1/2} + \mathcal{D}_y & 0 \\ 0 & \beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u \end{bmatrix}, \qquad \Psi = \begin{bmatrix} \mathcal{K} & -\tau\mathcal{M} \end{bmatrix}, \qquad \Theta = \begin{bmatrix} 0 \end{bmatrix}.$$

As for the time-independent case we may approximate Φ using diagonal solves or a multigrid method applied to the matrices from each time-step.

To approximate the Schur complement of (3.26),

$$\mathcal{S} = \mathcal{K}\big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)^{-1}\mathcal{K}^\top + \tau^2\mathcal{M}\big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)^{-1}\mathcal{M},$$

we again apply a matching strategy to obtain

$$\widehat{\mathcal{S}}_{1,T} := \big(\mathcal{K} + \widehat{\mathcal{M}}_{1,T}\big)\big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)^{-1}\big(\mathcal{K} + \widehat{\mathcal{M}}_{1,T}\big)^\top,$$

where

$$\widehat{\mathcal{M}}_{1,T}\big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)^{-1}\widehat{\mathcal{M}}_{1,T}^\top \approx \tau^2\mathcal{M}\big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)^{-1}\mathcal{M}.$$

This in turn motivates the choice

$$\widehat{\mathcal{M}}_{1,T} = \tau\mathcal{M}\left[\operatorname{diag}\big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)\right]^{-1/2}\left[\operatorname{diag}\big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)\right]^{1/2},$$

and we require two multigrid processes per time-step to apply $\widehat{\mathcal{S}}_{1,T}^{-1}$ efficiently.

Combining our approximations of $(1,1)$-block and Schur complement, we may again apply

$$\mathcal{P}_{1,T} = \begin{bmatrix} \big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)_{\text{approx}} & 0 & 0 \\ 0 & \big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)_{\text{approx}} & 0 \\ 0 & 0 & \widehat{\mathcal{S}}_{1,T} \end{bmatrix}$$

within MINRES, for example, or

$$\mathcal{P}_{2,T} = \begin{bmatrix} \big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)_{\text{approx}} & 0 & 0 \\ 0 & \big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)_{\text{approx}} & 0 \\ \mathcal{K} & -\tau\mathcal{M} & -\widehat{\mathcal{S}}_{1,T} \end{bmatrix},$$

within a nonsymmetric solver such as GMRES.

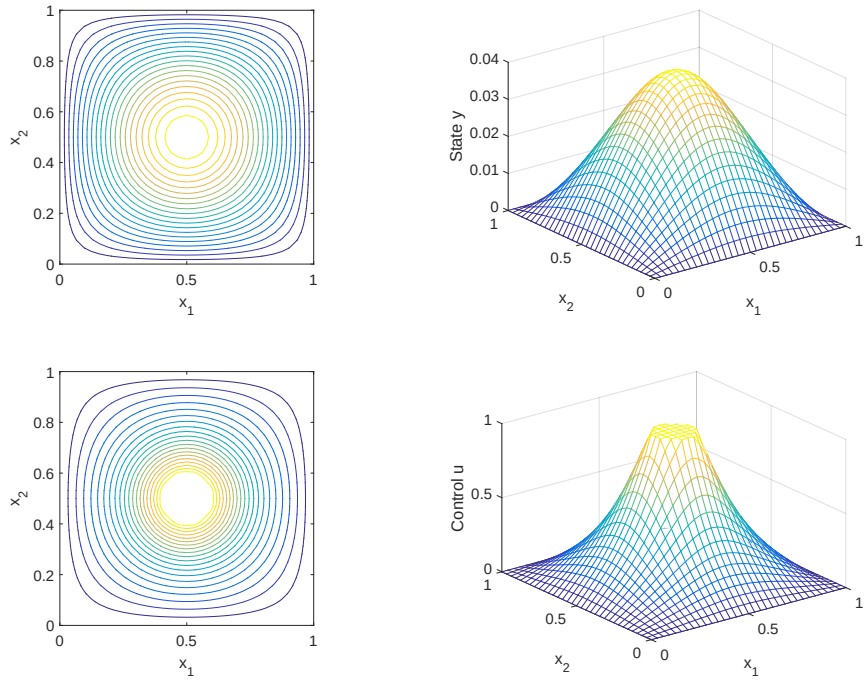

FIG. 5.1. *Contour and mesh plots of the solution to the Poisson control example with control constraints, for state variable y (top) and control variable u (bottom), with $\beta = 10^{-2}$.*

Alternatively, in complete analogy to the time-independent setting, we may rearrange the matrix system such that the $(1,1)$-block may be approximated accurately, and select the preconditioner

$$\mathcal{P}_{3,T} = \begin{bmatrix} -\widehat{\mathcal{S}}_{2,T} & 0 & \mathcal{K}^\top \\ 0 & \beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u & -\tau\mathcal{M}_{\text{cheb}} \\ 0 & -\tau\mathcal{M}_{\text{cheb}} & 0 \end{bmatrix}.$$

Inverting $\mathcal{M}_{\text{cheb}}$ requires the application of Chebyshev semi-iteration to N_t mass matrices M, and the Schur complement approximation is given by

$$\widehat{\mathcal{S}}_{2,T} := -\frac{1}{\tau^2}\big(\mathcal{K}^\top + \widehat{\mathcal{M}}_{2,T}\big)\mathcal{M}^{-1}\big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)\mathcal{M}^{-1}\big(\mathcal{K} + \widehat{\mathcal{M}}_{2,T}^\top\big),$$

with

$$\widehat{\mathcal{M}}_{2,T} = \tau\left[\text{diag}\big(\tau\mathcal{M}_{1/2} + \mathcal{D}_y\big)\right]^{1/2}\left[\text{diag}\big(\beta\tau\mathcal{M}_{1/2} + \mathcal{D}_u\big)\right]^{-1/2}\mathcal{M}.$$

Similar eigenvalue results can be shown for these Schur complement approximations as for the approximations used in the time-independent case.

5. Numerical experiments. Having motivated our numerical methods for the solution of the problems considered, we now wish to test our solvers on a range of examples. These test problems are of both time-independent and time-dependent form, and are solved on a desktop with a quad-core 3.2GHz processor. Within the interior point method, the value of the barrier reduction parameter σ is set to be

TABLE 5.1

Results for the Poisson control example with control constraints, for a range of values of h and β, and preconditioner $\mathcal{P}_1$. Presented are the number of interior point (Newton) iterations required to achieve convergence (blue, left), and average number of MINRES *steps per interior point iteration before a relative preconditioned residual norm of 10^{-6} is achieved (black, right). Results are given with an AGMG method applied to the $(1,1)$-block (top), and with a diagonal approximation (bottom).*

$\mathcal{P}_1$ AGMG		$\beta=1$ $u\geq 0$ $u\leq 0.01$		$\beta=10^{-1}$ $u\geq 0$ $u\leq 0.1$		$\beta=10^{-2}$ $u\geq 0$ $u\leq 1$		$\beta=10^{-3}$ $u\geq 0$ $u\leq 3$		$\beta=10^{-4}$ $u\geq 0$ $u\leq 20$		$\beta=10^{-5}$ $u\geq 0$ $u\leq 100$		$\beta=10^{-6}$ $u\geq 0$ $u\leq 300$	
h	2^{-2}	10	5.6	11	6.3	13	6.2	15	6.6	18	7.5	19	7.2	20	7.4
	2^{-3}	10	5.7	13	6.1	14	6.3	16	7.8	19	8.3	20	8.7	21	9.3
	2^{-4}	10	5.6	13	6.1	15	6.5	19	7.4	22	8.6	22	8.5	21	8.8
	2^{-5}	11	5.4	16	5.8	18	6.3	21	7.0	23	8.8	25	8.9	24	9.4
	2^{-6}	11	5.5	16	5.8	20	6.2	22	6.8	26	15.5	24	8.9	30	9.4
	2^{-7}	12	5.2	18	5.5	20	6.2	20	7.1	27	8.4	25	8.6	31	9.2

$\mathcal{P}_1$ Diagonal		$\beta=1$ $u\geq 0$ $u\leq 0.01$		$\beta=10^{-1}$ $u\geq 0$ $u\leq 0.1$		$\beta=10^{-2}$ $u\geq 0$ $u\leq 1$		$\beta=10^{-3}$ $u\geq 0$ $u\leq 3$		$\beta=10^{-4}$ $u\geq 0$ $u\leq 20$		$\beta=10^{-5}$ $u\geq 0$ $u\leq 100$		$\beta=10^{-6}$ $u\geq 0$ $u\leq 300$	
h	2^{-2}	9	9.4	11	10.4	13	9.5	15	9.2	18	10.1	19	9.4	20	17.6
	2^{-3}	10	15.1	12	16.7	14	16.9	16	18.4	19	17.5	20	18.5	21	19.5
	2^{-4}	10	15.5	15	18.6	16	19.9	19	22.7	22	21.6	22	23.4	21	24.3
	2^{-5}	11	16.3	16	16.2	19	19.5	21	21.1	23	24.7	25	25.7	24	25.8
	2^{-6}	11	15.5	16	20.2	20	16.9	22	18.9	26	32.1	24	18.9	31	26.7
	2^{-7}	12	14.3	18	15.7	21	16.1	20	18.5	28	28.8	25	19.3	31	23.4

TABLE 5.2

Results for the Poisson control example with state constraints, for a range of values of h and β. Presented are the number of interior point iterations required to achieve convergence (blue, left), and average number of GMRES *steps needed (black, right). Results are given when the preconditioners $\mathcal{P}_2$ (top) and $\mathcal{P}_3$ (bottom) are used.*

$\mathcal{P}_2$		$\beta=1$ $-0.1\leq y\leq 0.002$		$\beta=10^{-2}$ $-0.1\leq y\leq 0.175$		$\beta=10^{-4}$ $-0.1\leq y\leq 0.9$		$\beta=10^{-6}$ $-0.1\leq y\leq 1$	
h	2^{-2}	11	5.3	8	5.0	9	5.0	10	5.0
	2^{-3}	12	9.9	9	10.2	10	13.3	10	10.9
	2^{-4}	13	11.4	10	12.9	11	16.8	11	13.5
	2^{-5}	14	12.1	11	13.3	13	27.4	12	15.0
	2^{-6}	16	12.5	12	13.6	14	17.8	13	15.7
	2^{-7}	17	12.7	13	14.6	16	16.9	14	16.3

$\mathcal{P}_3$		$\beta=1$ $-0.1\leq y\leq 0.002$		$\beta=10^{-2}$ $-0.1\leq y\leq 0.175$		$\beta=10^{-4}$ $-0.1\leq y\leq 0.9$		$\beta=10^{-6}$ $-0.1\leq y\leq 1$	
h	2^{-2}	11	5.0	8	5.1	9	5.0	10	5.0
	2^{-3}	12	9.6	9	9.1	10	10.5	10	10.5
	2^{-4}	13	11.2	10	10.3	11	12.3	11	12.4
	2^{-5}	14	12.1	11	10.8	13	12.9	12	13.5
	2^{-6}	16	12.6	12	11.4	14	13.3	13	13.9
	2^{-7}	17	13.1	13	13.0	16	13.5	14	14.5

0.1, with $\alpha_0 = 0.995$, and $\epsilon_p = \epsilon_d = \epsilon_c = 10^{-6}$. To solve the Newton systems arising from the interior point method, we use the IFISS software package [7, 30] to construct the relevant finite element matrices. When the symmetric block diagonal preconditioner $\mathcal{P}_1$ is used, we solve the Newton systems using the MINRES algorithm to a relative preconditioned residual norm tolerance of 10^{-8}, and the AGMG software

TABLE 5.3

Number of interior point (Newton) iterations, average number of iterations of the Krylov subspace method per interior point step, and CPU time required to solve the Poisson control example with state constraints, when the preconditioners $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ are used. Results are presented for a range of h, and fixed $\beta = 10^{-2}$.

$\beta = 10^{-2}$		$\mathcal{P}_1$			$\mathcal{P}_2$			$\mathcal{P}_3$		
		IPM	Krylov	CPU	IPM	Krylov	CPU	IPM	Krylov	CPU
h	2^{-2}	8	8.0	**0.15**	8	5.0	**0.30**	8	5.1	**0.22**
	2^{-3}	9	11.8	**0.26**	9	10.2	**0.52**	9	9.1	**0.34**
	2^{-4}	10	14.5	**0.49**	10	12.9	**0.94**	10	10.3	**0.57**
	2^{-5}	11	14.1	**2.1**	11	13.3	**3.9**	11	10.8	**2.4**
	2^{-6}	13	14.8	**10.6**	12	13.6	**17.1**	12	11.4	**10.1**
	2^{-7}	14	14.9	**47.2**	13	14.6	**81.6**	13	13.0	**53.8**

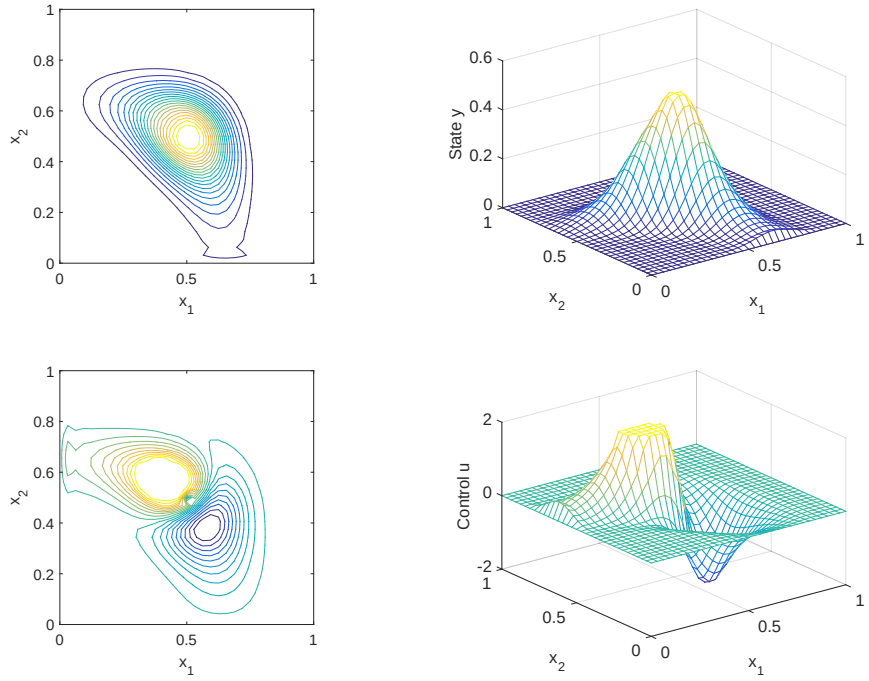

FIG. 5.2. *Contour and mesh plots of the solution to the convection-diffusion control example with state and control constraints, for state variable y (top) and control variable u (bottom), with $\beta = 10^{-2}$.*

to approximate the inverse of the $(1,1)$-block (apart from within one experiment where we use a diagonal approximation), as well as the inverse Schur complement. Where the block triangular preconditioners $\mathcal{P}_2$ and $\mathcal{P}_3$ are applied, we solve the Newton systems with the preconditioned GMRES method to a tolerance of 10^{-8}; we apply AGMG (for $\mathcal{P}_2$) or 20 steps of Chebyshev semi-iteration (for $\mathcal{P}_3$) to approximate the $(1,1)$-block, and once again utilize AGMG for the Schur complement approximation. All results are computed using MATLAB R2015a.

The first experiments we carry out involve a Poisson control problem, with $\mathcal{L} = -\nabla^2$ applied on $\Omega := [0,1]^2$, $y = 0$ on the boundary of Ω, and the desired state given by $\widehat{y} = e^{-64\left((x_1-0.5)^2+(x_2-0.5)^2\right)}$, where the spatial coordinates $\mathbf{x} = [x_1,\ x_2]^\top$. We

TABLE 5.4

Results for the convection-diffusion control example with state and control constraints, for a range of values of h and β. Presented are the number of interior point iterations required to achieve convergence (blue, left), and average number of GMRES *steps needed (black, right). Results are given when the preconditioners $\mathcal{P}_3$ (top) and $\mathcal{P}_2$ (bottom) are used.*

$\mathcal{P}_3$		$\beta = 10^{-1}$ $0 \le y \le 0.2$ $-0.75 \le u \le 0.75$		$\beta = 10^{-2}$ $0 \le y \le 0.5$ $-2 \le u \le 2$		$\beta = 10^{-3}$ $0 \le y \le 0.5$ $-3 \le u \le 3$		$\beta = 10^{-4}$ $0 \le y \le 0.75$ $-5 \le u \le 5$		$\beta = 10^{-5}$ $0 \le y \le 0.75$ $-6 \le u \le 6$	
h	2^{-2}	13	8.9	14	9.1	15	9.5	14	8.7	14	8.8
	2^{-3}	14	11.3	15	10.9	15	12.1	15	12.2	15	11.8
	2^{-4}	15	13.1	15	11.8	16	13.4	16	13.3	16	14.1
	2^{-5}	17	13.9	17	13.3	16	14.7	19	13.7	19	14.9
	2^{-6}	19	14.6	19	14.5	17	17.9	22	16.6	23	16.3
	2^{-7}	21	23.0	21	14.9	17	16.5	26	17.7	27	18.4

$\mathcal{P}_2$		$\beta = 10^{-1}$ $0 \le y \le 0.2$ $-0.75 \le u \le 0.75$		$\beta = 10^{-2}$ $0 \le y \le 0.5$ $-2 \le u \le 2$		$\beta = 10^{-3}$ $0 \le y \le 0.5$ $-3 \le u \le 3$		$\beta = 10^{-4}$ $0 \le y \le 0.75$ $-5 \le u \le 5$		$\beta = 10^{-5}$ $0 \le y \le 0.75$ $-6 \le u \le 6$	
h	2^{-2}	13	10.1	14	11.3	14	11.3	14	11.1	14	11.4
	2^{-3}	14	20.9	15	19.8	15	24.8	15	21.8	15	22.4
	2^{-4}	16	35.1	15	20.6	16	42.6	16	37.6	17	53.0
	2^{-5}	17	44.1	17	40.4	16	45.6	19	64.3	19	69.3
	2^{-6}	19	52.6	19	48.4	17	47.3	22	66.7	23	73.6

solve this problem using the MINRES algorithm with preconditioner $\mathcal{P}_1$, using both the AGMG method and the matrix diagonal to approximate the $(1,1)$-block within the preconditioner. The results obtained are shown in Table 5.1, for a range of mesh-sizes h and regularization parameters β. A solution plot for $\beta = 10^{-2}$ is also shown in Figure 5.1. We select box constraints for the control variable only, based on the value of β used and the behaviour of the optimal control problem when no bound constraints are imposed – we are careful to make sure that the constraints are sensible physically, but also challenging for our interior point solver. The constraints taken for each value of β are stated in Table 5.1. It is worth pointing out that increasing the accuracy of discretization (decreasing h by a factor of 2) typically adds about one extra interior point iteration, which once again demonstrates that interior point methods are not very sensitive to the problem dimension (as discussed in [10], for instance). We find that both the number of iterations of the interior point method, and the average number of MINRES iterations per interior point (Newton) step, are very reasonable for the problem considered. Whereas we observe an increase in iterative steps for the more challenging case of smaller β, all numbers are low, in particular the very encouraging iteration counts for moderate regularization parameters. We also find that, as one might expect, the computational cheapness of a diagonal approximation of the $(1,1)$-block is counteracted by the higher MINRES iteration numbers that result.

We next examine a Poisson control problem involving state constraints, where $\widehat{y} = \sin(\pi x_1)\sin(\pi x_2)$, and $y = \widehat{y}$ on the boundary of Ω. We apply the preconditioners $\mathcal{P}_2$ (with the AGMG routine to approximate the $(1,1)$-block) and $\mathcal{P}_3$, and solve using GMRES to a tolerance of 10^{-8} for a range of h and β. Again the results, which are presented in Table 5.2, are very promising when either preconditioner is used, and a large degree of robustness is achieved despite the very general matrix systems which can arise at each interior point iteration. We highlight that the iteration counts are likely to vary depending on how severe the box constraints that we impose are, as the structure of the matrices can change drastically. In Table 5.3 we present results

TABLE 5.5

Results for the heat equation control example with control constraints, for a range of values of h, τ and β, and preconditioner $\mathcal{P}_{1,T}$. Presented are the number of interior point iterations required to achieve convergence (blue, left), and average number of MINRES *steps needed (black, right).*

$\mathcal{P}_{1,T}$ $\tau = 0.04$		$\beta = 10^{-1}$ $0 \le u \le 0.1$		$\beta = 10^{-2}$ $0 \le u \le 1$		$\beta = 10^{-3}$ $0 \le u \le 3$		$\beta = 10^{-4}$ $0 \le u \le 30$	
h	2^{-2}	16	7.8	15	9.4	16	11.1	15	9.4
	2^{-3}	15	7.9	16	9.7	19	11.3	21	17.4
	2^{-4}	17	8.4	21	10.3	21	13.5	25	20.9
	2^{-5}	18	8.6	22	10.6	21	13.6	25	20.7
	2^{-6}	17	7.9	19	11.3	22	13.1	27	27.5

$\mathcal{P}_{1,T}$ $\tau = 0.02$		$\beta = 10^{-1}$ $0 \le u \le 0.1$		$\beta = 10^{-2}$ $0 \le u \le 1$		$\beta = 10^{-3}$ $0 \le u \le 3$		$\beta = 10^{-4}$ $0 \le u \le 30$	
h	2^{-2}	17	7.3	18	8.8	19	10.5	24	24.7
	2^{-3}	17	7.4	21	8.7	19	10.7	26	23.5
	2^{-4}	18	7.6	18	10.7	21	13.0	25	20.4
	2^{-5}	17	8.3	19	10.5	22	13.3	30	27.0
	2^{-6}	16	7.5	19	11.0	23	13.0	31	26.3

$\mathcal{P}_{1,T}$ $\tau = 0.01$		$\beta = 10^{-1}$ $0 \le u \le 0.1$		$\beta = 10^{-2}$ $0 \le u \le 1$		$\beta = 10^{-3}$ $0 \le u \le 3$		$\beta = 10^{-4}$ $0 \le u \le 30$	
h	2^{-2}	14	6.5	20	8.4	20	10.0	24	23.5
	2^{-3}	14	6.6	20	8.4	20	10.3	28	16.2
	2^{-4}	16	6.9	23	9.4	22	11.7	27	19.9
	2^{-5}	17	7.3	21	10.0	23	12.8	31	19.9
	2^{-6}	17	7.7	19	11.3	23	12.5	32	21.0

for this problem (for $\beta = 10^{-2}$) with preconditioners $\mathcal{P}_1$, $\mathcal{P}_2$ and $\mathcal{P}_3$ – we observe in particular that the CPU times scale in an approximately linear fashion with the dimension of the matrix system being solved.

In Table 5.4 we investigate a problem of convection-diffusion control type, with $\mathcal{L} = -0.01\nabla^2 + \left[-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right]^\top \cdot \nabla$, and $\widehat{y} = e^{-64\left((x_1-0.5)^2+(x_2-0.5)^2\right)}$. We now impose both state and control constraints (as specified for each value of β), and test the preconditioners $\mathcal{P}_2$ and $\mathcal{P}_3$ using GMRES. We also present a solution plot for $\beta = 10^{-2}$ in Figure 5.2. For convection-diffusion control problems such as this, we find there is a great advantage in applying the preconditioner $\mathcal{P}_3$ over the preconditioner $\mathcal{P}_2$, due to the accurate approximation of the $(1,1)$-block within it. Indeed this is demonstrated by the numbers of GMRES iterations required, which are much lower when using the preconditioner $\mathcal{P}_3$, especially for the final interior point iterations when convergence is close to being achieved. The GMRES solver with $\mathcal{P}_3$ demonstrates excellent robustness considering the complexity of the problem.

Finally, to demonstrate that our solvers are also able to handle matrix systems of vast dimension arising from time-dependent PDE-constrained optimization problems, we present results in Table 5.5 for a heat equation control problem, with the PDE constraint given by $y_t - \nabla^2 y = u$ (for $t \in (0,1]$), and with additional control constraints imposed. The number of interior point iterations, and average MINRES iteration count when $\mathcal{P}_{1,T}$ is applied, are provided for a range of h, β, and time-step τ. We once again observe a high degree of robustness in problem size (whether increased by refining the mesh in the spatial coordinates, or by decreasing the time-step) and regularization parameter. We emphasize once again that the performance of the method is dependent somewhat on the severity of the box constraints imposed,

however the numerical results obtained for a range of time-independent and time-dependent PDE-constrained optimization problems demonstrate the potency of the solvers presented in this manuscript.

6. Concluding remarks. In this paper we have presented a practical method for the interior point solution of a number of PDE-constrained optimization problems with state and control constraints, by reformulating the minimization of the discretized system as a quadratic programming problem. Having outlined the structure of the algorithm for solving these problems, we derived fast and feasible preconditioned iterative methods for solving the resulting Newton systems, which is the dominant portion of the algorithm in terms of computational work. Encouraging numerical results indicate the effectiveness and utility of our approach.

The problems we considered involved Poisson control, heat equation control, and both steady and time-dependent convection-diffusion control. A natural extension of this work would be to consider the control of systems of PDEs, for instance Stokes control and other problems in fluid flow, as well as the control of nonlinear PDEs, which arises in a wide range of practical scientific applications. The latter task would be accomplished by reformulating the discretization as a nonlinear programming problem – the robust solution of such formulations is a substantial challenge within the optimization community, but would represent significant progress in tackling real-world optimal control problems.

Acknowledgements. The first author was partially funded for this research by the Engineering and Physical Sciences Research Council (EPSRC) Fellowship EP/M018857/1.

REFERENCES

[1] M. Benzi, G. H. Golub, and J. Liesen, *Numerical solution of saddle point problems*, Acta Numerica, 14, pp.1–137, 2005.

[2] M. Benzi, E. Haber, and L. Taralli, *Multilevel algorithms for large-scale interior point methods*, SIAM Journal on Scientific Computing, 31, pp.4152–4175, 2009.

[3] L. Bergamaschi and A. Martinez, *RMCP: Relaxed mixed constraint preconditioners for saddle point linear systems arising in geomechanics*, Computer Methods in Applied Mechanics and Engineering, 221–222, pp.54–62, 2012.

[4] M. Bergounioux, M. Haddou, M. Hintermüller, and 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, pp.495–521, 2000.

[5] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Massachusetts, 1995.

[6] I. S. Duff, A. M. Erisman, and J. K. Reid, *Direct Methods for Sparse Matrices*, Oxford University Press, New York, 1987.

[7] H. C. Elman, A. Ramage, and D. J. Silvester, *IFISS: a computational laboratory for investigating incompressible flow problems*, SIAM Review, 56, pp.261–273, 2014.

[8] G. H. Golub and R. S. Varga, *Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second order Richardson iterative methods. I*, Numerische Mathematik, 3, pp.147–156, 1961.

[9] G. H. Golub and R. S. Varga, *Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second order Richardson iterative methods. II*, Numerische Mathematik, 3, pp.157–168, 1961.

[10] J. Gondzio, *Interior point methods 25 years later*, European Journal of Operational Research, 218, pp.587-601, 2012.

[11] J. Gondzio, *Convergence analysis of an inexact feasible interior point method for convex quadratic programming*, SIAM Journal on Optimization, 23, pp.1510–1527, 2013.

[12] M. J. Grotte, J. Huber, D. Kourounis, and O. Schenk, *Inexact interior-point method for PDE-constrained nonlinear optimization*, SIAM Journal on Scientific Computing, 36, pp.A1251–A1276, 2014.

[13] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich, *Optimization with PDE Constraints*, Mathematical Modelling: Theory and Applications, Springer-Verlag, New York, 2009.
[14] K. Ito and K. Kunisch, *Lagrange Multiplier Approach to Variational Problems and Applications*, Vol. 15 of Advances in Design and Control, Society for Industrial and Applied Mathematics, 2008.
[15] C. T. Kelley and E. W. Sachs, *Multilevel algorithms for constrained compact fixed point problems*, SIAM Journal on Scientific Computing, 15, pp.645–667, 1994.
[16] Y. A. Kuznetsov, *Efficient iterative solvers for elliptic finite element problems on nonmatching grids*, Russian Journal of Numerical Analysis and Mathematical Modelling, 10, pp.187–211, 1995.
[17] H. D. Mittelmann and H. Maurer, *Solving elliptic control problems with interior point and SQP methods: Control and state constraints*, Journal of Computational and Applied Mathematics, 120, pp.175–195, 2000.
[18] M. F. Murphy, G. H. Golub, and A. J. Wathen, A note on preconditioning for indefinite linear systems, SIAM Journal on Scientific Computing, 21, pp.1969–1972, 2000.
[19] A. Napov and Y. Notay, *An algebraic multigrid method with guaranteed convergence rate*, SIAM Journal on Scientific Computing, 34(2), pp.A1079–A1109, 2012.
[20] Y. Notay, *An aggregation-based algebraic multigrid method*, Electronic Transactions on Numerical Analysis, 37, pp.123–146, 2010.
[21] Y. Notay, *Aggregation-based algebraic multigrid for convection-diffusion equations*, SIAM Journal on Scientific Computing, 34(4), pp.A2288–A2316, 2012.
[22] Y. Notay, *AGMG software and documentation*; see `http://homepages.ulb.ac.be/~ynotay/AGMG`.
[23] C. C. Paige and M. A. Saunders, *Solutions of sparse indefinite systems of linear equations*, SIAM Journal on Numerical Analysis, 12(4), pp.617–629, 1975.
[24] J. W. Pearson, M. Stoll, and A. J. Wathen, *Regularization-robust preconditioners for time-dependent PDE-constrained optimization problems*, SIAM Journal on Matrix Analysis and Applications, 33(4), pp.1126–1152, 2012.
[25] J. W. Pearson and A. J. Wathen, *Fast iterative solvers for convection-diffusion control problems*, Electronic Transactions on Numerical Analysis, 40, pp.294–310, 2013.
[26] J. W. Pearson and A. J. Wathen, *A new approximation of the Schur complement in preconditioners for PDE-constrained optimization*, Numerical Linear Algebra with Applications, 19, pp.816–829, 2012.
[27] J. Pestana and T. Rees, *Null-space preconditioners for saddle point systems*, Rutherford Appleton Laboratory Technical Report RAL-TR-2015-003, 2015.
[28] T. Rusten and R. Winther, *A preconditioned iterative method for saddle point problems*, SIAM Journal on Matrix Analysis and Applications, 13, pp.887–904, 1992.
[29] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on Scientific Computing, 7(3), pp.856–869, 1986.
[30] D. Silvester, H. Elman, and A. Ramage, *Incompressible Flow and Iterative Solver Software (IFISS), Version 3.3*, `http://www.manchester.ac.uk/ifiss`, 2014.
[31] M. Stoll and A. Wathen, *All-at-once solution of time-dependent PDE-constrained optimization problems*, Oxford Centre for Collaborative Applied Mathematics Technical Report 10/47, 2010.
[32] F. Tröltzsch, *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*, American Mathematical Society, Providence, Rhode Island, 2010.
[33] M. Ulbrich and S. Ulbrich, *Primal-dual interior-point methods for PDE-constrained optimization*, Mathematical Programming, 117(1–2), pp.435–485, 2009.
[34] A. J. Wathen, *Realistic eigenvalue bounds for the Galerkin mass matrix*, IMA Journal of Numerical Analysis, 7(4), pp.449–457, 1987.
[35] A. J. Wathen and T. Rees, *Chebyshev semi-iteration in preconditioning for problems including the mass matrix*, Electronic Transactions on Numerical Analysis, 34, pp.125–135, 2009.
[36] M. Weiser, *Interior point methods in function space*, SIAM Journal on Control and Optimization, 44, pp.1766–1786, 2005.
[37] M. Weiser and P. Deuflhard, *Inexact central path following algorithms for optimal control problems*, SIAM Journal on Control and Optimization, 46, pp.792–815, 2007.
[38] M. Weiser, T. Gänzler, and A. Schiela, *A control reduced primal interior point method for a class of control constrained optimal control problems*, Computational Optimization and Applications, 41, pp.127–145, 2008.
[39] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.