Martin Stoll

One-shot solution of a time-dependent time-periodic PDE-constrained optimization problem

Max Planck Institute Magdeburg Preprints

MPIMD/11-04 **July 19, 2011**

Impressum:

Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg

Publisher:
Max Planck Institute for Dynamics of Complex Technical Systems

Address:
Max Planck Institute for Dynamics of Complex Technical Systems
Sandtorstr. 1
39106 Magdeburg

www.mpi-magdeburg.mpg.de/preprints

IMA Journal of Numerical Analysis (2011) Page 1 of 22
doi:10.1093/imanum/drnxxx

One-shot solution of a time-dependent time-periodic PDE-constrained optimization problem

MARTIN STOLL†
Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg Germany,

[Received on 2011; revised on]

In this paper we describe the efficient solution of a PDE-constrained optimization problem subject to the time-periodic heat equation. We propose a space-time formulation for which we develop a monolithic solver. We present preconditioners well suited to approximate the Schur-complement of the saddle point system associated with the first order conditions. This means that in addition to a Richardson iteration based preconditioner we also introduce a preconditioner based on the tensor product structure of the PDE discretization, which allows the use of a FFT based preconditioner. We also consider additional bound constraints that can be treated using a semi-smooth Newton method. Moreover, we introduce robust preconditioners with respect to the regularization parameter. Numerical results will illustrate the competitiveness and flexibility of our approach.

Keywords:

PDE-constrained optimization, Saddle point systems, Time-dependent PDE-constrained optimization, Preconditioning, Krylov subspace solver

1. Introduction

For many years the solution of so-called forward partial differential equations (PDE) problems has been in the focus of the numerical analysis and scientific computing community. Although, there are many challenges left the progress made over the last decades has enabled the search for, in some sense, optimal solutions of PDEs. The task in the field often labeled PDE-constrained optimization is to minimize an objective function subject to constraints given by PDEs. Introductions to the field can be found in Tröltzsch (2005, 2010); Hinze *et al.* (2009); Ito & Kunisch (2008).

A typical example will look like the following

$$\min \quad J(y,u) \tag{1.1}$$
$$\text{s.t } \mathscr{L}(y,u) = 0, \tag{1.2}$$

where $J(y,u)$ is the function we want to minimize and $\mathscr{L}(y,u) = 0$ represents a differential operator, typically a PDE, that links the state y and the control u. We assume that suitable boundary conditions are given and in the case of time-dependent problems initial conditions are specified. Often the introduction of additional constraints, such as bound constraints on the control and/or the state poses additional challenges (cf. Tröltzsch (2005, 2010); Hinze *et al.* (2009); Ito & Kunisch (2008) for suitable methods to deal with this).

†Corresponding author. Email: martin.stoll80@gmail.com

Our focus in this paper is to solve a problem of the above type where the PDE constraint is equipped with appropriate boundary conditions and the state y exhibits time-periodicity, i.e., $y(0,;) = y(T,:)$ where T is the final time we are interested in. Problems of this type have recently been analyzed in Potschka *et al.* (2010); Abbeloos *et al.* (2011), where the motivation for the optimal control problem was a chemical engineering application such as the one given in Kawajiri & Biegler (2006).

The paper is organized as follows. In the next section, we are introducing the problem formulation and the PDE constraint with time-periodicity. We will discuss the discretization and then give the first order optimality system. In Section 3 we introduce the semi-smooth Newton method developed in Bergounioux *et al.* (1999); Hintermüller *et al.* (2002) that will allow us to handle box constraints on the control. We will then briefly motivate our choice of Krylov subspace solver. In Section 5, we discuss the preconditioners that are suitable for our approach. Namely, a Richardson iteration based preconditioner and also a preconditioner using the circulant structure of the discretized PDE. We will discuss issues that arise when boundary control is employed and the dependency on the regularization parameter. Numerical results in Section 6 illustrate the efficiency of our approach.

2. Problem and discretization

In this paper, we will analyze tracking type functionals subject to a time-periodic partial differential equation. The functional that we want to minimize is given by

$$J_1(y,u) := \frac{1}{2}\int_0^T \int_{\Omega_1} (\mathrm{y}(\mathrm{x},\mathrm{t}) - \overline{\mathrm{y}}(\mathrm{x},\mathrm{t}))^2\, dxdt + \frac{\beta}{2}\int_0^T \int_{\Omega_2} (\mathrm{u}(\mathrm{x},\mathrm{t}))^2\, dxdt. \tag{2.1}$$

where $\Omega_{1,2} \subseteq \Omega$ are domains in $\mathbb{R}^d$ with $d = \{2,3\}$, y is the state, $\overline{\mathrm{y}}$ the desired state and u the control. We want to minimize this functional subject to the time-periodic heat equation that links the state and the control and is hence called the state equation. In more detail, the equation now reads as

$$y_t - \triangle y = u \tag{2.2}$$

defined over $\Omega \times [0,T]$, with Dirichlet boundary condition $y = 0$ on the spatial boundary $\partial\Omega$ and time-periodic condition $y(x,0) = y(x,T)$. In addition, we will allow for variations of this problem. The first is the so-called boundary control problem given by

$$J_{bnd}(y,u) := \frac{1}{2}\int_0^T \int_{\Omega_1} (\mathrm{y}(\mathrm{x},\mathrm{t}) - \overline{\mathrm{y}}(\mathrm{x},\mathrm{t}))^2\, dxdt + \frac{\beta}{2}\int_0^T \int_{\partial\Omega} (\mathrm{u}(\mathrm{x},\mathrm{t}))^2\, dxdt. \tag{2.3}$$

subject to

$$y_t - \triangle y = f \tag{2.4}$$

with Neumann boundary condition

$$\frac{\partial y}{\partial n} = u$$

and some forcing term f, that we assume to be zero. Additionally, the introduction of bound constraints on control and/or state pose additional challenges to numerical algorithms. Bounds such as

$$u_a \leqslant u \leqslant u_b \text{ and } y_a \leqslant y \leqslant y_b$$

have to be accounted for by more sophisticated algorithms (Tröltzsch (2005, 2010); Ito & Kunisch (2008); Hinze *et al.* (2009).) We will discuss the necessary approaches for control constraints in later

parts of this paper (see Section 3). There are two ways to proceed from the above problems. First, one can write down the infinite dimensional first order conditions and then discretize them, this is the so-called *Optimize-then-Discretize* approach. The other approach is to discretize first and then write down the first order or KKT conditions, this is the so-called *Discretize-then-Optimize* approach. It is desired for many problems that these two approaches coincide, which is taken into account when devising discretization schemes such as the ones derived in Hinze *et al.* (2008). We will follow the Discretize-then-Optimize approach. Hence, we discretize both the functional and the PDE using standard Galerkin finite elements, rectangular in our case because of the underlying use of deal.II (cf. Bangerth *et al.* (2007)), which does not use triangular elements.

For the time-discretization of the PDE we use a backward Euler scheme that leads to the following semi-discretized form of (2.2)

$$\frac{y^k - y^{k-1}}{\tau} - \triangle y^k = u^k \tag{2.5}$$

with τ being the time-step and the number of grid points in time is denoted by N_T. And similarly for the second PDE (2.4). The finite element discretization in space is straightforward and putting all time steps into one system, a so-called one-shot approach, leads to

$$\underbrace{\begin{bmatrix} M+\tau K & & & & -M \\ -M & M+\tau K & & & \\ & -M & M+\tau K & & \\ & & \ddots & \ddots & \\ & & & -M & M+\tau K \end{bmatrix}}_{\mathscr{K}} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \vdots \\ \mathbf{y}_N \end{bmatrix} - \tau \mathscr{N} \mathbf{u} = d, \tag{2.6}$$

where M and K are the finite element (lumped) mass and stiffness matrix and d the right-hand side representing the boundary conditions and forcing terms. Note that $\mathscr{K}$ exhibits circulant structure, which we will discuss in more detail later. Further, we have the matrix $\mathscr{N} = \text{blkdiag}(N, N, \ldots, N)$, where N can be a rectangular matrix depending on the nature of the optimal control problem and its discretization. N is a square mass matrix if y and u are discretized using the same finite elements and u represents a distributed control. In the case of u being discretized using a different finite element than the state y, e.g., piecewise constant elements for the control and linear finite elements for the state, or if u is a boundary control, then N is a rectangular matrix. In the case of boundary control N will consist of entries coming from the integral $\int_{\partial\Omega} u\,\text{tr}(v)$, where u is the boundary control and tr is the trace operator acting on the test function v from the test space used for the discretization of the state y.

We now need to discretize the objective function $J(y,u)$ and for this we use the trapezoidal rule to get the discretized objective function as

$$J(\mathbf{y},\mathbf{u}) = \frac{\tau}{2}(\mathbf{y}-\overline{\mathbf{y}})^T \mathscr{M}_y (\mathbf{y}-\overline{\mathbf{y}}) + \frac{\tau\beta}{2}\mathbf{u}^T \mathscr{M}_u \mathbf{u} \tag{2.7}$$

where $\mathscr{M}_y = \text{blkdiag}(1/2\tilde{M}, \tilde{M}, \ldots, 1/2\tilde{M})$ where $\tilde{M}$ is the mass matrix over the domain Ω_1 and $\mathscr{M}_u = \text{blkdiag}(1/2\hat{M}, \hat{M}, \ldots, 1/2\hat{M})$ where $\hat{M}$ represents the mass matrix for the domain Ω_2.

Once all these ingredients are available, we can combine them into a Lagrangian and write down the first order conditions, which can be written as the following KKT system

$$\begin{bmatrix} \tau\mathscr{M}_y & 0 & -\mathscr{K}^T \\ 0 & \beta\tau\mathscr{M}_u & \tau\mathscr{N}^T \\ -\mathscr{K} & \tau\mathscr{N} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathscr{M}_y \overline{\mathbf{y}} \\ 0 \\ d \end{bmatrix}. \tag{2.8}$$

Note that 2.8 represents a saddle point system with a symmetric and positive semi-definite $(1,1)$ block given by blkdiag$(\tau\mathscr{M}_y, \beta\tau\mathscr{M}_u)$ and a full rank block $[-\mathscr{K} \quad \tau\mathscr{N}]$. Here, p represents the discrete Lagrange multiplier or equivalently the solution to the adjoint PDE. These conditions are sufficient for the invertibility of the saddle point system. Note that if instead of the trapezoidal rule we would have used a rectangular rule for the discretization of the objective function the block blkdiag$(\tau\mathscr{M}_y, \beta\tau\mathscr{M}_u)$ would contain zero blocks. Note that in the case of only a final time observation in $J(y,u)$, i.e. , the first term in (2.1) changes to

$$\frac{1}{2}\int_{\Omega_1} (\mathrm{y}(\mathrm{x},\mathrm{T}) - \overline{\mathrm{y}}(\mathrm{x},\mathrm{T}))^2\, dx$$

and in this case the $(1,1)$ block of the saddle point system will be highly singular (see Stoll & Wathen (2010, 2011); Benzi *et al.* (2010); Simoncini (2011)). Note that the above linear system will typically be of very large dimension, i.e., the dimension is given by $3nN_T$, where n is the number of the degrees of freedom of the PDE discretization and N_T the number of grid points in time.

Note that a reduction of the dimensionality of the above system is possible by eliminating the control $\mathbf{u}$ a technique that is also discussed in Simoncini (2011); Hinze (2005). In case that the $(1,1)$ block is positive definite, we can also only work with the Schur-complement reduction as also the mass matrices are lumped and hence the evaluation of $\mathscr{M}_y^{-1}$ and $\mathscr{M}_u{}^{-1}$ is trivial. This case would enable the use of the classical CG method Hestenes & Stiefel (1952) but the challenge in developing good preconditioners for the Schur-complement stays intact. Note this does not apply if $\mathscr{M}_y$ and $\mathscr{M}_u$ are not invertible, e.g., $\Omega_1 \subsetneq \Omega$.

3. Bound constraints

For the treatment of bound constraints on the control we want to use a semi-smooth Newton method. This method was first derived in Bergounioux *et al.* (1999) under the name primal-dual active set method and it was later shown (cf. Hintermüller *et al.* (2002)) that the method is a non-smooth (also called semi-smooth) Newton method. For an introduction to semi-smooth Newton methods we refer to Hinze *et al.* (2009); Qi & Sun (1993); Ito & Kunisch (2008); Ulbrich (2011). The minimization problem $\min J(\mathbf{u},\mathbf{y})$ can be identified with the discrete optimality system

$$\begin{cases} \min \quad F(\mathbf{u}) \quad \text{s.t.} \\ \underline{u}_a \leqslant \mathbf{u} \leqslant \overline{u}_b \quad \text{in } \Omega, \end{cases} \tag{3.1}$$

where we used the state equation to remove the state via the control from the functional $J(y,u)$. In our case, the optimality condition for $F(\mathbf{u})$ becomes

$$\Phi(\mathbf{u}) := P_{[\underline{u}_a,\overline{u}_b]}(\mathbf{u} - D(\beta\mathscr{M}_u\mathbf{u} + H'(\mathbf{u}))) - \mathbf{u} = 0$$

where D is a diagonal matrix with positive entries, P the projection onto the interval $[\underline{u}_a,\overline{u}_b]$, and $H'(\mathbf{u}) = \mathscr{N}^T\mathscr{K}^{-T}\mathscr{M}_y\mathscr{K}^{-1}\mathscr{N}\mathbf{u} - \mathscr{N}^T\mathscr{K}^{-T}\mathscr{M}_y\overline{\mathbf{y}}$(see Theorem 5.2.4 in Kelley (1999)). Note that the gradient of $F(\mathbf{u})$ is given by $\nabla F(\mathbf{u}) = \beta\mathscr{M}_u\mathbf{u} + H'(\mathbf{u})$ and with the choice $D = \mathscr{M}_u^{-1}$ we get

$$\Phi(\mathbf{u}) := P_{[\underline{u}_a,\overline{u}_b]}(\mathbf{u} - \beta\mathbf{u} - \mathscr{M}_u^{-1}H'(\mathbf{u})) - \mathbf{u} = 0.$$

As $\Phi(\mathbf{u})$ is a non-smooth functional the Newton system is given by

$$M_k s_{\mathbf{u}}^{(k)} = -\Phi(\mathbf{u}^{(k-1)}) \tag{3.2}$$

where the generalized differential is defined as

$$M_k = G - \beta G - G\mathscr{M}_u^{-1}\mathscr{N}^T\mathscr{K}^{-T}\mathscr{M}_y\mathscr{K}^{-1}\mathscr{N} - I$$

with

$$(G)_{jj} = \begin{cases} 0 & \mathbf{u} - \mu \notin (\underline{u}_a, \overline{u}_b) \\ 1 & otherwise \end{cases}$$

and $\mu = \beta\mathbf{u} + \mathscr{M}_u^{-1}H'(\mathbf{u})$. Without loss of generality, we can assume that the variables are ordered such that

$$G = \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}.$$

Note that solving the system (3.2) with M_k can be achieved by solving

$$\begin{bmatrix} \mathscr{M}_y & 0 & -\mathscr{K}^T \\ 0 & -I + G - \beta G & -G\mathscr{M}_u^{-1}\mathscr{N}^T \\ -\mathscr{K} & \mathscr{N} & 0 \end{bmatrix} \begin{bmatrix} s_{\mathbf{y}}^{(k)} \\ s_{\mathbf{u}}^{(k)} \\ s_p^{(k)} \end{bmatrix} = \begin{bmatrix} 0 \\ -\Phi(\mathbf{u}^{(k-1)}) \\ 0 \end{bmatrix}. \tag{3.3}$$

We could stop here for the implementation of a semi-smooth Newton method but as we want to obtain the implementation of the active set method presented earlier as well as obtain a symmetric linear system that can be solved much more efficiently than the one given in (3.3), we use the definition

$$\begin{aligned} \Phi(\mathbf{u}^{(k-1)}) &= P_{[\underline{u}_a, \overline{u}_b]}(\mathbf{u}^{(k-1)} - (\beta\mathbf{u}^{(k-1)} + \mathscr{M}_u^{-1}H'(\mathbf{u}^{(k-1)})) - \mathbf{u}^{(k-1)} \\ &= P_{[\underline{u}_a, \overline{u}_b]}(\mathbf{u}^{(k-1)} - \mu^{(k-1)}) - \mathbf{u}^{(k-1)} \end{aligned}$$

with $\mu^{(k-1)} = \beta\mathbf{u}^{(k-1)} + \mathscr{M}_u^{-1}H'(\mathbf{u}^{(k-1)})$. We now use the above to rewrite (3.3) to get

$$\begin{bmatrix} \mathscr{M}_y & 0 & -\mathscr{K}^T \\ 0 & -I + G - \beta G & G\mathscr{M}_u^{-1}\mathscr{N}^T \\ -\mathscr{K} & \mathscr{N} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}^{(k)} - \mathbf{y}^{(k-1)} \\ \mathbf{u}^{(k)} - \mathbf{u}^{(k-1)} \\ p^{(k)} - p^{(k-1)} \end{bmatrix} = \begin{bmatrix} 0 \\ -\Phi(\mathbf{u}^{(k-1)}) \\ 0 \end{bmatrix}, \tag{3.4}$$

which is also equivalent to

$$\begin{bmatrix} \mathscr{M}_y & 0 & -\mathscr{K}^T \\ 0 & -I + G - \beta G & -G\mathscr{M}_u^{-1}\mathscr{N}^T \\ -\mathscr{K} & \mathscr{N} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}^{(k)} \\ \mathbf{u}^{(k)} \\ p^{(k)} \end{bmatrix} = \begin{bmatrix} -\mathscr{K}^T p^{(k-1)} + \mathscr{M}_y\mathbf{y}^{(k-1)} \\ -\Phi(\mathbf{u}^{(k-1)}) + (-I + G - \beta G)\mathbf{u}^{(k-1)} - G\mathscr{M}_u^{-1}\mathscr{N}^T p^{(k-1)} \\ -\mathscr{K}\mathbf{y}^{(k-1)} + \mathscr{N}\mathbf{u}^{(k-1)} \end{bmatrix}. \tag{3.5}$$

We now have to take care of the part in (3.5) that corresponds to the control $\mathbf{u}$. For that we are splitting the control in its parts corresponding to the active sets based in $\mathbf{u}^{(k-1)}$, and consider the following three cases

$$\Phi(\mathbf{u}^{(k-1)}) \begin{cases} = (\underline{u}_a - \mathbf{u}^{(k-1)})_i & \text{for all } i \in \mathbf{A}_-^{(k-1)} \\ = (\mathbf{u}^{(k-1)} - \mu^{(k-1)} - \mathbf{u}^{(k-1)})_i & \text{for all } i \in \mathbf{A}_I^{(k-1)} \\ = (\overline{u}_b - \mathbf{u}^{(k-1)})_i & \text{for all } i \in \mathbf{A}_+^{(k-1)} \end{cases} \tag{3.6}$$

where $\mathbf{A}_-^{(k-1)} = \left\{ i : (\mathbf{u}^{(k-1)} - \mu^{(k-1)})_i < (\underline{u}_a)_i \right\}$, $\mathbf{A}_+^{(k-1)} = \left\{ i : (\mathbf{u}^{(k-1)} - \mu^{(k-1)})_i > (\overline{u}_b)_i \right\}$ and $\mathbf{A}_I^{(k-1)}$ for the free variables at step $k-1$. For convenience we neglect the indices of the active sets in the linear systems. We can equivalently split up $-\Phi(\mathbf{u}^{(k-1)}) + (-I + G - \beta G)\mathbf{u}^{(k-1)} - G\mathscr{M}_u^{-1}\mathscr{N}^T p^{(k-1)}$ using (3.6) and the definition of μ to get

$$-\Phi(\mathbf{u}^{(k-1)}) + (-I + G - \beta G)\mathbf{u}^{(k-1)} - G\mathscr{M}_u^{-1}\mathscr{N}^T p^{(k-1)} \begin{cases} = -\underline{u}_a & \text{for all } i \in \mathbf{A}_-^{(k-1)} \\ = 0 & \text{for all } i \in \mathbf{A}_I^{(k-1)} \\ = -\overline{u}_b & \text{for all } i \in \mathbf{A}_+^{(k-1)}. \end{cases} \tag{3.7}$$

Putting this together into a linear system now gives

$$\begin{bmatrix} \mathscr{M}_y & 0 & 0 & 0 & -\mathscr{K}^T \\ 0 & -I & 0 & 0 & 0 \\ 0 & 0 & -I & 0 & 0 \\ 0 & 0 & 0 & \beta I & G^{\mathbf{A}_I,:}\mathscr{M}_u^{-1}\mathscr{N}^T \\ -K & \mathscr{N}^{:,\mathbf{A}_+} & \mathscr{N}^{:,\mathbf{A}_-} & \mathscr{N}^{:,\mathbf{A}_I} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}^{(k)} \\ \mathbf{u}_{\mathbf{A}_+}^{(k)} \\ \mathbf{u}_{\mathbf{A}_-}^{(k)} \\ \mathbf{u}_{\mathbf{A}_I}^{(k)} \\ \lambda^{(k)} \end{bmatrix} = \begin{bmatrix} \mathscr{M}\overline{\mathbf{y}} \\ -\overline{u}_b \\ -\underline{u}_a \\ 0 \\ d \end{bmatrix}. \tag{3.8}$$

We now eliminate the rows corresponding to $\mathbf{u}_{\mathbf{A}_+}^{(k)}$ and $\mathbf{u}_{\mathbf{A}_-}^{(k)}$ and also multiply the row corresponding to $\mathbf{u}_{\mathbf{A}_I}^{(k)}$ by $\mathscr{M}^{\mathbf{A}_I,\mathbf{A}_I}$, a diagonal matrix, and using the fact that $\mathscr{M}_u^{\mathbf{A}_I,\mathbf{A}_I}\left(G^{\mathbf{A}_I,:}\mathscr{M}_u^{-1}\mathscr{N}^T\right) = (\mathscr{N}^{:,\mathbf{A}_I})^T$ to get

$$\begin{bmatrix} \mathscr{M}_y & 0 & -\mathscr{K}^T \\ 0 & \beta\mathscr{M}_u^{\mathbf{A}_I,\mathbf{A}_I} & \left(\mathscr{N}^{:,\mathbf{A}_I}\right)^T \\ -K & \mathscr{N}^{:,\mathbf{A}_I} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}^{(k)} \\ \mathbf{u}_{\mathbf{A}_+}^{(k)} \\ \mathbf{u}_{\mathbf{A}_-}^{(k)} \\ \mathbf{u}_{\mathbf{A}_I}^{(k)} \\ \lambda^{(k)} \end{bmatrix} = \begin{bmatrix} \mathscr{M}_y\overline{\mathbf{y}} \\ 0 \\ d - \mathscr{N}^{:,\mathbf{A}_+}\overline{u}_b - \mathscr{N}^{:,\mathbf{A}_-}\underline{u}_a \end{bmatrix}. \tag{3.9}$$

Note that this system can be solved similarly to the ones for the unconstrained case (see Stoll & Wathen (2009)) as we implicitly work with the system matrix defined in (3.9) and use a preconditioner for the problem without bound-constraints. The advantage of this approach is that the preconditioner only needs to be constructed once. We will discuss this further in Section 6. For the convergence properties of the active set or equivalently semi-smooth Netwon method we refer to Bergounioux *et al.* (1999); Hintermüller *et al.* (2002). Note that the semi-smooth Newton method converges superlinearly if the initial guess is sufficiently close to the solution of the optimality system (see Hinze *et al.* (2009); Ito & Kunisch (2008) for more details).

4. Choice of Krylov solver

As the dimensionality of the linear system is very large and the applications are likely to be three-dimensional, direct methods based on a factorization of the saddle point system (cf. Davis (2005); Duff (1996)) will not be applicable for realistic scenarios of the above described problem. Therefore, we apply iterative Krylov solvers. These methods build up a so-called Krylov subspace

$$\mathscr{K}_k(\mathscr{A}, r_0) = \text{span}\left\{ r_0, \mathscr{A}r_0, \ldots, \mathscr{A}^{k-1}r_0 \right\}$$

and then construct an approximation to the solution of the linear system based upon some optimality criteria for the current iteration. In the case of a symmetric and positive definite left-upper block conjugate gradient (CG) methods (see Hestenes & Stiefel (1952)) can be applied; typically with a non-standard inner product. There are a number of candidates based upon these non-standard inner product, which usually employ different preconditioners and hence different inner products. The Bramble-Pasciak CG introduced in Bramble & Pasciak (1988) is a very successful method coming from finite element solutions of the Stokes problem and has recently been used for optimal control problems (see Rees & Stoll (2010)). Schöberl and Zulehner proposed another method in Schöberl & Zulehner (2007) that also has been used successfully for optimal control problems by Herzog & Sachs (2010). Our method of choice here will be the minimal residual method (MINRES) Paige & Saunders (1975) which minimizes the residual over the current Krylov space. This method needs a symmetric and positive definite preconditioner, which typically would look like

$$\mathscr{P} = \begin{bmatrix} A_0 & 0 \\ 0 & S_0 \end{bmatrix}, \tag{4.1}$$

where A_0 approximates the left-upper block and S_0 approximates the Schur-complement of the saddle point system. These choices within $\mathscr{P}$ are motivated by a result given in Murphy *et al.* (2000), where it is shown that the choices of A_0 as the unchanged left-upper block and S_0 as the negative Schur-complement lead to three distinct eigenvalues in the preconditioned system. Our goal is hence to find good approximations to both the Schur-complement and the upper left block.

The problem of solving time-periodic PDE problems is not a new one and a variety of methods have been proposed to solve the forward problem, see Vandewalle & Piessens (1992) for a multigrid approach or Bomhof & van der Vorst (2001); Bomhof (2001) for a GMRES technique applicable to cyclic systems. The method given in Vandewalle & Piessens (1992) has been used for the optimal control problem studied in Abbeloos *et al.* (2011). In Ernst (2000) an overview of iterative methods that apply to *p*-cyclic matrices is presented.

5. Preconditioners

As we have seen in the previous section the choice of approximations for the $(1,1)$-block of our system and the Schur-complement

$$\tau^{-1}\mathscr{K}\mathscr{M}_y^{-1}\mathscr{K}^T + \tau\beta^{-1}\mathscr{N}\mathscr{M}_u^{-1}\mathscr{N}^T, \tag{5.1}$$

where we assume that $\mathscr{M}_y$ and $\mathscr{M}_u$ are both invertible, is crucial. Note that this is not the case if $\Omega_1 \subsetneq \Omega$ or a rectangular rule is used for the approximation of the time integral but even in that case we can get good preconditioners that somewhat approximate an equation that resembles (5.1). Note that we initially follow a strategy used in Rees *et al.* (2010b) to drop the second term $\tau\beta^{-1}\mathscr{N}\mathscr{M}_u^{-1}\mathscr{N}^T$ but we will later comment on and introduce alternatives.

5.1 $(1,1)$-*block*

Our goal in this section is to derive effective approximations to the upper left block and the Schur-complement. The left upper block is given by blkdiag$(\tau\mathscr{M}_y, \beta\tau\mathscr{M}_u)$. This leaves us with the problem of efficiently approximating mass matrices. This is a trivial task once the mass matrices are lumped. In the case of consistent mass matrices the Chebyshev semi-iteration (see Algorithm 1) is a viable tool for preconditioning and has been used successfully for optimal control applications (see Rees *et al.* (2010b,a)).

Set $D = diag(M)$
Set relaxation parameter ω
Compute $g = \omega D^{-1}\hat{b}$
Set $S = (I - \omega D^{-1}M)$ (this can be used implicitly)
Set $z_{k-1} = 0$ and $z_k = Sz_{k-1} + g$
$c_{k-1} = 2$ and $c_k = \omega$
for $k = 2, \ldots, l$ **do**
    $c_{k+1} = \omega c_k - \frac{1}{4}c_{k-1}$
    $\vartheta_{k+1} = \omega \frac{c_k}{c_{k+1}}$
    $z_{k+1} = \vartheta_{k+1}(Sz_k + g - z_{k-1}) + z_{k-1}$
end for

Algorithm 1: Chebyshev semi-iterative method for a number of l steps

5.2 *Schur-complement: Richardson iteration*

In Stoll & Wathen (2010) we studied all-at-once approaches for the heat equation. In contrast to our previous results where the matrix representing the one-shot discretization was a lower block-triangular matrix, we now have an additional term in the upper right corner of $\mathscr{K}$ coming from the periodicity condition. Our goal is to derive preconditioners that deal with the Schur-complement approximation $\widehat{S} = \mathscr{K}\widehat{\mathscr{M}}^{-1}\mathscr{K}^T$, where $\widehat{\mathscr{M}}$ represents a symmetric positive definite approximation to $\tau\mathscr{M}_y$, e.g., in the case of the trapezoidal rule and $\Omega_1 = \Omega$ this will simply be $\tau\mathscr{M}_y$. We now approximate $\widehat{S}^{-1}$ by approximating $\mathscr{K}^{-1}$ and $\mathscr{K}^{-T}$ using the Richardson iteration. The idea of a Richardson iteration is rather simple as we can use a trivial identity

$$\mathscr{K}x = Ix + (\mathscr{K} - I)x = b$$

and rearranging the last part we get

$$x = (I - \mathscr{K})x + b.$$

We can now turn this into an iterative method in the following way

$$x^{(k+1)} = (I - \mathscr{K})x^{(k)} + b.$$

It is well known Saad (2003) that this method converges if the eigenvalues of the matrix $\mathscr{K}$ lie within the unit disk. To improve the convergence of this approach a preconditioner $\mathscr{P}$ can be introduced as follows

$$\mathscr{P}^{-1}\mathscr{K}x = Ix + (\mathscr{P}^{-1}\mathscr{K} - I)x = \mathscr{P}^{-1}b$$

resulting in the following iteration

$$x^{(k+1)} = (I - \mathscr{P}^{-1}K)x^{(k)} + \mathscr{P}^{-1}b$$

or equivalently

$$x^{(k+1)} = x^{(k)} - \mathscr{P}^{-1}r_k$$

with $r_k = \mathscr{K} x_k - b$ the residual. Many well known methods fit into this scheme and we refer to Saad (2003) for details. For our problem we decide to use the following preconditioner

$$P = \begin{bmatrix} \widehat{L} & & & & \\ -M & \widehat{L} & & & \\ & -M & \widehat{L} & & \\ & & \ddots & \ddots & \\ & & & -M & \widehat{L} \end{bmatrix}$$

for the forward PDE, where $\widehat{L}$ is an approximation to the matrix $L = \tau K + M$. We will use a fixed number of algebraic multigrid cycles (AMG) as $\hat{L}^{-1}$. This approach is feasible as we are not interested in the solution of the PDE problem but only in an approximation as part of the preconditioner. Note that for $\widehat{L} = L$ this simply is the Gauss-Seidel method. Similarly, we proceed for the adjoint PDE represented by $\mathscr{K}^T$ with the preconditioner

$$P = \begin{bmatrix} \widehat{L} & -M & & & \\ & \widehat{L} & -M & & \\ & & \widehat{L} & \ddots & \\ & & & \ddots & -M \\ & & & & \widehat{L} \end{bmatrix}.$$

In our experiments we performed two steps of the Richardson iteration scheme given in for both forward and adjoint problem as an approximation to the Schur-complement.

Note that it would also be possible to use other preconditioners such as the one used in Jacobi's method, i.e., a block diagonal with the blocks given by $\widehat{L}$.

5.3 *Schur-complement: Circulant approach*

We will now focus on a different approximation of the Schur-complement. For this we study the structure of the discretized forward problem defined by the one-shot operator

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

which can be written as

$$\mathscr{K} = I \otimes \tau K + C \otimes M \tag{5.2}$$

with

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

a circulant matrix. It is well known from Chen (1987) that the matrix C can be diagonalized using the Fourier matrix F, i.e.,

$$C = F\,\mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{N_T})F^H.$$

Note that if C is non-singular we have

$$C^{-1} = F\,\mathrm{diag}(\lambda_1^{-1}, \lambda_2^{-1}, \ldots, \lambda_{N_T}^{-1})F^H.$$

If we apply the FFT to the matrix $\mathscr{K}y = c$ we get

$$\left(F^H \otimes I_{N_T}\right)\mathscr{K}\left(F \otimes I_{N_T}\right)\left(F^H \otimes I_{N_T}\right)y = \left(F^H \otimes I_{N_T}\right)g$$

and using the definition of $\mathscr{K}$ this becomes

$$\begin{aligned}\left(F^H \otimes I_{N_T}\right)\mathscr{K}\left(F \otimes I_{N_T}\right) &= \left(F^H \otimes I_{N_T}\right)\left(I \otimes \tau K + C \otimes M\right)\left(F \otimes I_{N_T}\right) &(5.3)\\ &= F^H F \otimes \tau K + F^H C F \otimes M &(5.4)\\ &= I \otimes \tau K + \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{N_T}) \otimes M. &(5.5)\end{aligned}$$

The eigenvalues λ_j can be determined via Chen (1987) as

$$\lambda_j = c_0 + [c_1 + c_{n-1}]\cos\left(\frac{(j-1)2\pi}{k}\right) + i[c_1 - c_{n-1}]\sin\left(\frac{(j-1)2\pi}{k}\right)$$

for $j = 1, \ldots, N_T$. In our case we get $c_1 = 0$, $c_0 = 1$ and $c_{n-1} = -1$ and hence

$$\lambda_j = 1 - \cos\left(\frac{(j-1)2\pi}{k}\right) + i\sin\left(\frac{(j-1)2\pi}{k}\right)$$

All of this results in a block-diagonal matrix with the diagonal elements in the following form

$$W_j = \tau K + \lambda_j M = \tau K + (1 - \cos\left(\frac{(j-1)2\pi}{k}\right))M + i\sin\left(\frac{(j-1)2\pi}{k}\right)M.$$

W_j represents one of the blocks of the block-diagonal matrix that we have to solve for. First, we have to point out that the application of the Fourier transform will in general result in complex valued systems. In more detail, the diagonal blocks mentioned above represent N_T complex valued linear systems, i.e.,

$$\left(\tau K + (1 - \cos\left(\frac{(j-1)2\pi}{k}\right))M + i\sin\left(\frac{(j-1)2\pi}{k}\right)M\right)(y_r + iy_c) = (g_r + ig_c)\forall j \qquad (5.6)$$

or equivalently

$$\begin{bmatrix} U & -V \\ V & U \end{bmatrix}\begin{bmatrix} y_r \\ y_c \end{bmatrix} = \begin{bmatrix} g_r \\ g_c \end{bmatrix} \qquad (5.7)$$

using

$$U = \tau K + (1 - \cos\left(\frac{(j-1)2\pi}{k}\right))M \text{ and } V = \sin\left(\frac{(j-1)2\pi}{k}\right)M.$$

The linear system can also be written in symmetric form to give

$$\begin{bmatrix} U & -V \\ -V & -U \end{bmatrix}\begin{bmatrix} y_r \\ y_c \end{bmatrix} = \begin{bmatrix} g_r \\ -g_c \end{bmatrix}. \qquad (5.8)$$

Once, the solution to the system (5.11) is computed we have to transform the solution back using the Fourier transform F. Note that the same has to be done for the adjoint PDE as we have to approximate the solution of both forward and adjoint PDE in order to approximate the Schur-complement. Note that the one-shot discretization of the adjoint PDE is characterized by

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

which can be written as

$$\mathscr{K} = I \otimes \tau K + \tilde{C} \otimes M \tag{5.9}$$

with

$$\tilde{C} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & \ddots & \\ 0 & 0 & 0 & \ddots & -1 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Similar to the forward PDE we see that $\tilde{C}$ is a circulant matrix, which means we can diagonalize it using the Fourier matrix to get

$$I \otimes \tau K + \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{N_T}) \otimes M$$

where the eigenvalues λ are determined from

$$\lambda_j = c_0 + [c_1 + c_{n-1}] \cos\left(\frac{(j-1)2\pi}{k}\right) + i[c_1 - c_{n-1}] \sin\left(\frac{(j-1)2\pi}{k}\right)$$

with $c_0 = 1$, $c_1 = -1$, and $c_{n-1} = 0$ to give

$$\lambda_j = 1 - \cos\left(\frac{(j-1)2\pi}{k}\right) - i \sin\left(\frac{(j-1)2\pi}{k}\right).$$

These are simply the complex conjugates of the eigenvalues of the forward circulant matrix. Again, we have to solve a complex linear system and we use the above presented approach to get

$$\left(\tau K + (1 - \cos\left(\frac{(j-1)2\pi}{k}\right))M - i \sin\left(\frac{(j-1)2\pi}{k}\right) M\right)(y_r + i y_c) = (g_r + i g_c) \tag{5.10}$$

or equivalently

$$\begin{bmatrix} U & V \\ V & -U \end{bmatrix} \begin{bmatrix} y_r \\ y_c \end{bmatrix} = \begin{bmatrix} g_r \\ -g_c \end{bmatrix}. \tag{5.11}$$

Again, the solution to the complex linear system has to be transformed back using F via the FFT. We will now discuss how to solve the linear systems associated with the complex valued system. Note that preconditioning a matrix of block-circulant type was also recently studied for the solution of a forward time-periodic PDE (see Greidanus (2010)).

5.4 *Solving the complex linear system*

As we have already seen in the previous section, the circulant approach to both the forward and the adjoint problem leads to a complex-valued linear system. We want to solve the complex systems in their real form shown in (5.8) and (5.11). As these systems arise within an outer MINRES iteration we need the iterative solver for both systems to represent a linear operator. This would not be achieved in case a Krylov solver is used due to its nonlinearity. Instead we propose to use a fixed number of steps of an inexact Uzawa-type method as was already proposed for Stokes control in both the steady (cf. Rees & Wathen (2010)) and the unsteady (cf. Stoll & Wathen (2011)) case. The main iteration of the inexact Uzawa method can be cast in the following form

$$x_{k+1} = x_k + \omega \mathscr{P}^{-1} r_k,$$

which means we need to multiply with the system matrices

$$\begin{bmatrix} U & V \\ V & -U \end{bmatrix} \text{ and } \begin{bmatrix} U & -V \\ -V & -U \end{bmatrix}, \tag{5.12}$$

with $U = \tau K + (1-c)M$ and $V = sM$ where s and c are abbreviations for the sine and cosine values used before. Matrices that resemble the ones used in (5.12) can be found in the numerical solution of the bidomain equations (see Pennacchio & Simoncini (2011, 2009)). There we have a two by two block matrix that has mass matrix plus stiffness matrix terms as diagonal blocks and the off-diagonals are mass matrices. We want to use a preconditioner $\mathscr{P} = \text{blkdiag}(A_0, A_1)$, where A_0 approximates the $(1,1)$ block of (5.8) or (5.11) and A_1 the corresponding Schur-complement. For the bidomain equations the choice of A_1 approximating the $(2,2)$-block also gives good results as the Schur-complement is rather complicated and one way could be to use $U + V\text{diag}(U)V^T$ as an approximation. Here we will stay with the choice of an algebraic multigrid method for both the $(1,1)$ and the $(2,2)$ block. In fact, in our computations we simply use the algebraic multigrid that we need for the $L = \tau K + M$ block as an approximation of U. This is due to the fact that otherwise we would need N_T different preconditioners for one space-time solve as all the diagonal blocks in the circulant approach are different, which would be infeasible.

5.5 *Singular constraints*

As we also consider boundary control, it has to be noted that this means for the stiffness matrix of the Laplace part to be only positive semi-definite. The operator and hence the matrix K have a one-dimensional kernel, that for the matrix K is written as $\mathbf{1}$, the vector of all ones of the appropriately chosen dimension, i.e., $K\mathbf{1} = 0$. It can easily be seen that the vector $\left[\mathbf{1}^T, \mathbf{1}^T, \ldots, \mathbf{1}^T\right]^T$ is in the nullspace of the one-shot discretization $\mathscr{K}$ of the time-dependent PDE (cf. (2.2) (2.4)). Problems of similar type occur in applications such as the treatment of the hydrostatic pressure in the solution of Stokes flow (see Elman *et al.* (2005)). Hence, for the singular problems we refer to Section 2.3 in Elman *et al.* (2005), where it is stated that iterative methods will be able to handle the singularity as any Richardson method will converge as long as all non-zero eigenvalues of the iteration matrix are inside the unit disk.

In the case of the circulant approach we note that the first of the diagonal blocks will become a pure Neumann problem for both the real and the complex part. Hence, the system we want to solve is given by

$$\begin{bmatrix} \tau K & 0 \\ 0 & \tau K \end{bmatrix} \begin{bmatrix} y_r \\ y_c \end{bmatrix} = \begin{bmatrix} g_r \\ g_c \end{bmatrix}, \tag{5.13}$$

with K a Neumann Laplacian. As a consequence we have to solve two uncoupled pure Neumann problems. The solution of pure Neumann problems is a fundamental problem in many application and has to be treated carefully as the system matrix τK has a one dimensional kernel spanned by **1**. Bochev and Lehoucg present a review of techniques to overcome this dilemma in Bochev & Lehoucq (2005). We now solve the system

$$\Pi^T K \Pi x = \Pi^T b$$

where Π is the following projection operator

$$\Pi = I - \frac{wc^T}{c^T w}$$

with c being in the span of **1** and $w \in \mathbb{R}^n$ is chosen such that $c^T w > 0$. For more details we refer to Bochev & Lehoucq (2005).

5.6 *Dependence on the regularization parameter*

There has recently been a surge in the development of preconditioners that show not only mesh-independent convergence behaviour but also have independence with respect to the regularization parameter β. Zulehner *et al.* (cf. Takacs & Zulehner (2011); Schöberl & Zulehner (2007)) introduce a symmetric positive definite preconditioner/inner product matrix that incorporates the regularization parameter β and the resulting iteration numbers are independent of β. In the case of distributed control, the dependence on β for many values of the parameter could be observed to be very benign for the Schur-complement approximation given by

$$\hat{S} = \tau^{-1} \hat{\mathscr{K}} \mathscr{M} \hat{\mathscr{K}}^T.$$

For boundary control this observation is no longer true.

Pearson and Wathen (see Pearson & Wathen (2010)) recently introduced a different approximation of the Schur-complement of a non-time-dependent, distributed control problem by

$$\widehat{S} = \left(K + \frac{1}{\sqrt{\beta}} M \right) M^{-1} \left(K + \frac{1}{\sqrt{\beta}} M \right), \tag{5.14}$$

which can be obtained by dropping the term $-\frac{2}{\sqrt{\beta}} K$ from the Schur-complement

$$S = \left(K + \frac{1}{\sqrt{\beta}} M \right) M^{-1} \left(K + \frac{1}{\sqrt{\beta}} M \right) - \frac{2}{\sqrt{\beta}} K.$$

Note that for our problem given in (2.8) we cannot simply use this technique but rather use an extension to the time-dependent, boundary control case that will be analyzed in a different publication (cf. Pearson *et al.* (2011)) and hence we only motivate it here. The Schur complement of the time-dependent problem is given by

$$S = \tau^{-1} \mathscr{K} \mathscr{M}_y^{-1} \mathscr{K}^T + \tau \beta^{-1} \mathscr{N} \mathscr{M}_u^{-1} \mathscr{N}^T$$

and this will now be approximated by

$$\hat{S} = \tau^{-1} \left(\mathscr{K} + \hat{\mathscr{M}} \right) \mathscr{M}_y^{-1} \left(\mathscr{K}^T + \hat{\mathscr{M}} \right), \tag{5.15}$$

where $\hat{\mathscr{M}} = \text{blkdiag}(0, \tau\sqrt{\frac{h}{\beta}}\mathscr{M}_u)$ if we assume that the degrees of freedoms corresponding to nodes on the boundary are ordered so they appear in the last components. Note that h is the mesh-size and this scaling has to be introduced to compensate for the different order of the boundary mass matrix and the mass matrix over the whole domain. We remember at this stage that $\mathscr{N}\mathscr{M}_u^{-1}\mathscr{N}^T$ will be a diagonal matrix with $\mathscr{M}_u$ a block-diagonal matrix consisting of lumped boundary mass matrices. A second alternative that can also be found in Pearson *et al.* (2011) is given by

$$\hat{S} = \tau^{-1}\left(\mathscr{K} + \hat{\mathscr{M}}\right)\check{\mathscr{M}}^{-1}\left(\mathscr{K}^T + \hat{\mathscr{M}}\right) \tag{5.16}$$

where $\hat{\mathscr{M}}$ as defined before and $\check{\mathscr{M}} = h\text{blkdiag}(hI, \mathscr{M}_u)$ in $2D$ and $\check{\mathscr{M}} = h\text{blkdiag}(h^2I, \mathscr{M}_u)$ in $2D$ again assuming the degrees of freedom would be such that the boundary nodes correspond to the right lower entries in the matrix $\check{\mathscr{M}}$.

One important fact about these new Schur-complement approximations is that the preconditioning strategies mentioned earlier, i.e., Richardson iteration and circulant approach, apply with almost no changes to this case. Hence, we will not discuss how to implement these new approximations efficiently.

6. Numerical Experiments

6.1 *Setup and implementation details*

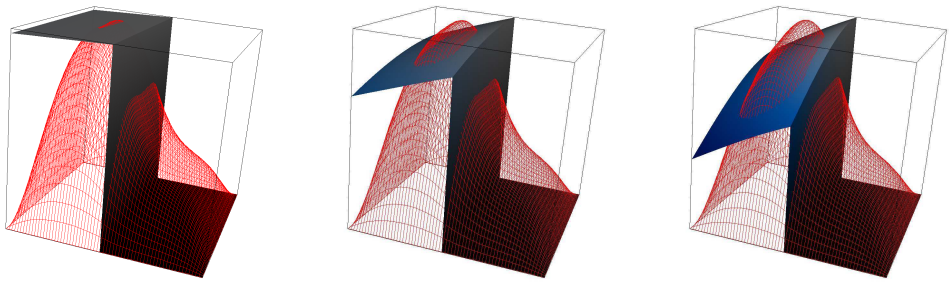
In this section we provide numerical experiments for the methods presented above. For the discretization we used the deal.II library (cf. Bangerth *et al.* (2007)), which is implemented in C++ using quadrilateral elements. As deal.II provides easy access to the Trilinos ML AMG package our multigrid approximations were performed using Trilinos' smoothed aggregation preconditioners (see Gee *et al.* (2006)). We approximated the blocks involving L as the sum of a mass matrix and stiffness matrix, also the matrices possibly involving scalar factors in front of M or K, by 2 steps of an AMG V-cycle and 10 steps of a Chebyshev smoother. The Richardson iteration based preconditioner uses 2 steps for both the adjoint and the forward problem. The circulant based preconditioner uses 3 steps of the inexact Uzawa method for every complex linear system. The application of the FFT needed for the circulant approach was provided by employing FFTW (see Frigo & Johnson (1998, 2005)). We use a relative tolerance of 10^{-4} for the pseudo-residual and $\tau = 0.05$ with $T = 1$, i.e., $N_T = 20$ unless mentioned otherwise. All experiments are performed on a Centos Linux machine with Intel(R) Xeon(R) CPU X5650 @ 2.67GHz CPUs and 48GB of RAM. Note that no parallelism is exploited in the implementation of our algorithms. If we mention the degrees of freedom in the Tables further on, they usually refer to the degrees of freedom needed for one grid point in time. So the dimensionality for the overall system is much larger. To give an example, Table 5 mentions 274625 degrees of freedom; this number needs to be multiplied by 3 and the number of grid points in time (20) to give the dimension of the system that is solved implicitly 16 477 500.

6.2 *Distributed control*

The first example is a distributed control example with zero Dirichlet condition. The desired state is given by

$$\bar{\mathbf{y}} = \begin{cases} 0.5\left(2.0 + \sin(0.5t\pi x_0) + \cos(0.5t\pi(1-x_1))\right) & x_0 < 0.5 \\ 0.5 & x_0 \geqslant 0.5, \end{cases}$$

which together with the computed state is depicted in Figure 1 for different time-steps. The computations in Figure 1 are done with $\beta = 10^{-4}$. Note that the Dirichlet boundary condition $y = 0$ on $\partial\Omega$



(a) $\bar{y}_0$ and y_0 (b) $\bar{y}_{10}$ and y_{10} (c) $\bar{y}_{19}$ and y_{19}

Figure 1: Desired (solid) and computed (mesh) state at different points in time.

forces the state to differ drastically from the desired state on the boundary. A remedy could be to set the desired state to zero on the boundary or to the Dirichlet boundary to match the desired state. These choices clearly depend on the underlying problem and the requirements coming from an application.

In Table 1 we show the FFT preconditioned method for two different values of β to illustrate that the dependence on the regularization parameter β is rather benign.

Table 2 shows the comparison of the Richardson iteration vs. the circulant approach for a three-dimensional example. The desired state in this case is given by

$$\bar{\mathbf{y}}(t) = 2^{10t} x_0 x_2 x_1 (x_0 - 1)(x_1 - 1)(x_2 - 1),$$

which for $\bar{\mathbf{y}}_{10}$ is depicted in Figure 2a. Figure 2b shows a spherical slice of the computed state that approximates $\bar{\mathbf{y}}_{10}$. The rather significant difference in timings might be due to the fact that we perform 2 Richardson iterations vs. 3 inexact Uzawa for every complex system of which we have to solve N_T with the each system of system dimension $2n$.

	$\beta = 1e-2$	$\beta = 1e-4$
DoF	MINRES(T)	MINRES(T)
4225	12(20)	38(61)
16641	12(74)	38(216)
66049	12(313)	38(889)
263169	12(1390)	38(4180)

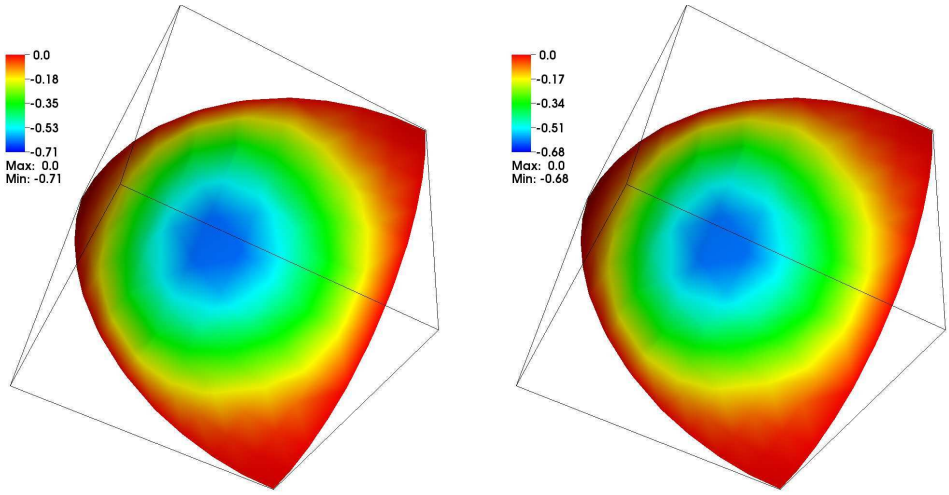
Table 1: Different values for β: Number of MINRES iterations (CPU time).

6.3 *Boundary control*

Our next task is to illustrate the performance of the preconditioners proposed for the boundary control problem presented earlier. In the case of boundary control we noticed that the approximation for the

	FFT	RI
DoF	MINRES(T)	MINRES(T)
729	10(7)	6(2)
4913	10(35)	6(8)
35937	10(267)	6(58)
274625	10(2780)	6(681)

Table 2: Different preconditioners: FFT (left) vs. Richardson (right) and number of MINRES iterations (CPU time) for $\beta = 1e-4$.



(a) $\bar{y}_{10}$ (b) y_{10}

Figure 2: Desired and computed state at different points in time.

Schur-complement given by

$$\tau^{-1}\mathcal{K}\mathcal{M}^{-1}\mathcal{K}^T$$

did not seem to suffice to guarantee convergence within a reasonable number of iterations. Hence, we developed the preconditioners presented in Section 5.6 that can use all the techniques presented for the original approximation as both the structure for the Richardson iteration as well as the circulant structure remain untouched. We will now illustrate the performance of the new Schur-complement approximation. We first compare the two different approximations for S given in (5.15) and (5.16) for a two-dimensional example defined by the following desired state

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

Table 3 shows the results for (5.15) and different mesh-size as well as different values for β. Table 4 shows the results for a similar setup with the approximation to the Schur-complement coming from

(5.16). We can see that both methods perform very similar with slightly fewer iterations for the second approximations.

	$\beta = 1e-2$	$\beta = 1e-4$	$\beta = 1e-6$
DoF	MINRES(T)	MINRES(T)	MINRES(T)
289	24(3)	32(3)	34(4)
1089	30(10)	42(14)	46(15)
4225	32(25)	54(41)	60(45)
16641	34(92)	70(186)	76(202)
66049	38(406)	88(921)	100(1065)

Table 3: Different values for β with approximation $\hat{S}$ defined in (5.15)

	$\beta = 1e-2$	$\beta = 1e-4$	$\beta = 1e-6$
DoF	MINRES(T)	MINRES(T)	MINRES(T)
289	22(3)	30(3)	32(3)
1089	24(9)	38(13)	42(14)
4225	30(24)	50(38)	58(44)
16641	32(90)	62(167)	72(193)
66049	36(400)	82(868)	96(1008)

Table 4: Different values for β with approximation $\hat{S}$ defined in (5.16).

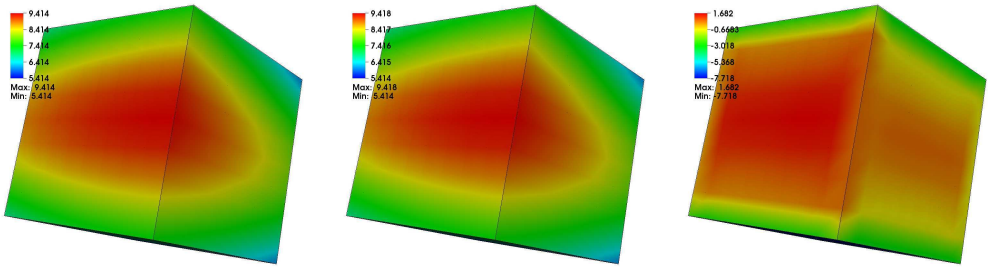
We next want to illustrate the performance of our approach for a three-dimensional boundary control problem. The desired state is given by

$$\overline{\mathbf{y}} = (2 + \cos(x_0 t\pi/2) + \cos((1-x_1)t\pi/2.0) + \exp(x_2\pi))/(t+0.05).$$

Figure 3a shows the desired state $\overline{\mathbf{y}}_{10}$, Figure 3b and 3c shows the computed state $\mathbf{y}_{10}$ and control $\mathbf{u}_{10}$, respectively. The results for this example are shown in Table 5 for $\beta = 1e-4$ and different mesh-sizes.

	$\beta = 1e-4$
DoF	MINRES(T)
729	32(18)
4913	44(130)
35937	56(1378)
274625	74(19560)

Table 5: 3D results for different mesh-sizes.



(a) $\bar{\mathbf{y}}_{10}$ (b) $\mathbf{y}_{10}$ (c) $\mathbf{u}_{10}$

Figure 3: Desired state, computed state, and computed control.

6.4 *Boundary control with box-constraints*

We finally want to present results for the case of boundary control in the presence of box constraints. The desired state $\bar{\mathbf{y}}$is defined by

$$\begin{cases} \sin(t(1-t)) + x_0x_1x_2 & \text{for } x_0 > 0.5 \text{ and } x_1 < 0.5 \\ 1 & \text{otherwise.} \end{cases} \tag{6.1}$$

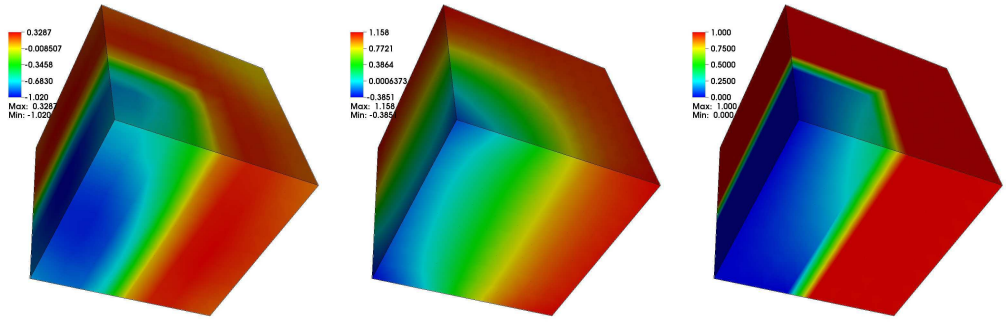
In Figure 4c we show the computed results for this problem without any box constraints on the control. In Figure 5 the results are shown in the presence of the upper bound set to 0.15. The results for $\beta = 1e-2$ and the bound $u_b = 0.15$ are shown in Table 6. It can be seen that the growth in iteration numbers is rather benign similar to the case when no box constraints are present. Nevertheless, the individual iteration numbers for one Newton system are slightly higher than in the unconstrained case, which might be due to the fact that the same preconditioner is used in every iteration of the active set method not taking the structure of (3.9) into account. It would be possible to incorporate the structure of the Newton system into the preconditioner by construction a new preconditioner for every Newton step. This should be done in the future for more realistic scenarios.

DoF	AS	Total number MINRES(Total time)
729	5	346(194)
4913	5	407(1225)
35937	6	584(16232)

Table 6: 3D results for active set method with $\beta = 1e-2$ and different mesh-sizes.

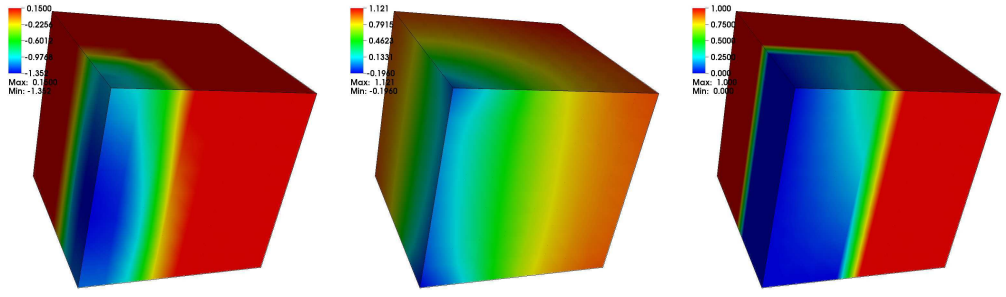
7. Conclusions

In this paper we presented a monolithic solver for the space-time discretization of an optimal control problem subject to the time-periodic heat equation. We formulated a one-shot approach that resulted in a huge linear system in saddle point form. With our choice of MINRES as the Krylov subspace solver we



(a) $\mathbf{u}_{15}$ (b) $\mathbf{y}_{15}$ (c) $\bar{\mathbf{y}}_{15}$

Figure 4: Desired state, computed state, and computed control.



(a) $\mathbf{u}_{15}$ (b) $\mathbf{y}_{15}$ (c) $\bar{\mathbf{y}}_{15}$

Figure 5: Desired state, computed state, and computed control in the presence of bound constraints.

were focusing on the task of devising good preconditioners for Schur-complement of the saddle point matrix. We proposed two techniques, one based on a Richardson iteration and the other one circulant formulation that allowed the use of the FFT. Both methods performed competitive for distributed control. In the case of boundary control, we introduced a new Schur-complement approximation that allowed more flexibility with respect to the regularization parameter β. The efficient solution of the control problem also enabled the fast solution of the minimization when control constraints are present. We showed the results for boundary control with box constraints on the control and illustrated the flexibility of our approach.

Acknowledgments

The author would like to thank Andy Wathen for his comments and Daniel Kressner for pointing him to some references.

REFERENCES

ABBELOOS, D., DIEHL, M., HINZE, M. & VANDEWALLE, S. (2011) Nested multigrid methods for time-periodic, parabolic optimal control problems. *accepted Computing and Visualization in Science*.

BANGERTH, W., HARTMANN, R. & KANSCHAT, G. (2007) deal.II—a general-purpose object-oriented finite element library. *ACM Trans. Math. Software*, **33**, Art. 24, 27.

BENZI, M., HABER, E. & TARALLI, L. (2010) A preconditioning technique for a class of pde-constrained optimization problems. *To appear in Advances in Computational Mathematics*.

BERGOUNIOUX, M., ITO, K. & KUNISCH, K. (1999) Primal-dual strategy for constrained optimal control problems. *SIAM J. Control Optim.*, **37**, 1176–1194.

BOCHEV, P. & LEHOUCQ, R. (2005) On the finite element solution of the pure neumann problem. *SIAM review*, **47**, 50–66.

BOMHOF, C. (2001) Iterative and parallel methods for linear systems, with applications in circuit simulation. *Ph.D. thesis*, Universiteit Utrecht.

BOMHOF, W. & VAN DER VORST, H. (2001) A parallelizable gmres-type method for p-cyclic matrices, with applications in circuit simulation. *Scientific computing in electrical engineering: proceedings of the 3rd international workshop, August 20-23, 2000, Warnemünde, Germany*. Springer Verlag, Springer Verlag, p. 293.

BRAMBLE, J. H. & PASCIAK, J. E. (1988) A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems. *Math. Comp*, **50**, 1–17.

CHEN, M. (1987) On the solution of circulant linear systems. *SIAM J. Numer. Anal.*, **24**, 668–683.

DAVIS, T. (2005) Umfpack version 4.4 user guide. *Technical Report*. Dept. of Computer and Information Science and Engineering Univ. of Florida, Gainesville, FL.

DUFF, I. (1996) Sparse numerical linear algebra: direct methods and preconditioning. *Technical Report* TR/PA/96/22. CERFACS, Toulouse, France. Also RAL Report RAL 96-047.

ELMAN, H. C., SILVESTER, D. J. & WATHEN, A. J. (2005) *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*. Numerical Mathematics and Scientific Computation. New York: Oxford University Press, pp. xiv+400.

ERNST, O. (2000) Equivalent iterative methods for p-cyclic matrices. *Numerical Algorithms*, **25**, 161–180.

FRIGO, M. & JOHNSON, S. G. (1998) FFTW: An adaptive software architecture for the FFT. *Proc. 1998 IEEE Intl. Conf. Acoustics Speech and Signal Processing*, vol. 3. IEEE, pp. 1381–1384.

FRIGO, M. & JOHNSON, S. G. (2005) The design and implementation of FFTW3. *Proceedings of the IEEE*, **93**, 216–231. Special issue on "Program Generation, Optimization, and Platform Adaptation".

GEE, M., SIEFERT, C., HU, J., TUMINARO, R. & SALA, M. (2006) ML 5.0 smoothed aggregation user's guide. *Technical Report* SAND2006-2649. Sandia National Laboratories.

GREIDANUS, J. W. (2010) *Efficient Computation of Periodic Orbits in Space-Time Discretized Nonlinear Dynamical Systems. Master's thesis*, University of Groningen.

HERZOG, R. & SACHS, E. W. (2010) Preconditioned conjugate gradient method for optimal control problems with control and state constraints. *SIAM J. Matrix Anal. Appl.*, **31**, 2291–2317.

HESTENES, M. R. & STIEFEL, E. (1952) Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand*, **49**, 409–436 (1953).

HINTERMÜLLER, M., ITO, K. & KUNISCH, K. (2002) The primal-dual active set strategy as a semismooth Newton method. *SIAM J. Optim.*, **13**, 865–888.

HINZE, M. (2005) A variational discretization concept in control constrained optimization: the linear-quadratic case. *Comput. Optim. Appl.*, **30**, 45–61.

HINZE, M., KÖSTER, M. & TUREK, S. (2008) A Hierarchical Space-Time Solver for Distributed Control of the Stokes Equation. *Technical Report*. SPP1253-16-01.

HINZE, M., PINNAU, R., ULBRICH, M. & ULBRICH, S. (2009) *Optimization with PDE Constraints*. Mathematical Modelling: Theory and Applications. Springer-Verlag, New York.

ITO, K. & KUNISCH, K. (2008) *Lagrange multiplier approach to variational problems and applications*. Advances in Design and Control, vol. 15. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), pp. xviii+341.

KAWAJIRI, Y. & BIEGLER, L. (2006) Optimization strategies for simulated moving bed and powerfeed processes. *AIChE journal*, **52**, 1343–1350.

KELLEY, C. T. (1999) *Iterative methods for optimization*. Frontiers in Applied Mathematics, vol. 18. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), pp. xvi+180.

MURPHY, M. F., GOLUB, G. H. & WATHEN, A. J. (2000) A note on preconditioning for indefinite linear systems. *SIAM J. Sci. Comput*, **21**, 1969–1972.

PAIGE, C. C. & SAUNDERS, M. A. (1975) Solutions of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal*, **12**, 617–629.

PEARSON, J. W., STOLL, M. & WATHEN, A. J. (2011) Regularization-robust preconditioners for time-dependent PDE-constrained optimization problems. *Submitted*.

PEARSON, J. W. & WATHEN, A. J. (2010) A New Approximation of the Schur Complement in Preconditioners for PDE Constrained Optimization. *Submitted*.

PENNACCHIO, M. & SIMONCINI, V. (2009) Algebraic multigrid preconditioners for the bidomain reaction-diffusion system. *Applied numerical mathematics*, **59**, 3033–3050.

PENNACCHIO, M. & SIMONCINI, V. (2011) Fast structured AMG preconditioning for the bidomain model in electrocardiology. *SIAM Journal on Scientific Computing*, **33**, 721.

POTSCHKA, A., MOMMER, M., SCHLÖDER, J. & BOCK, H. (2010) A Newton-Picard approach for efficient numerical solution of time-periodic parabolic PDE constrained optimization problems. *Interdisciplinary Center for Scientific Computing, Heidelberg University*.

QI, L. Q. & SUN, J. (1993) A nonsmooth version of Newton's method. *Math. Programming*, **58**, 353–367.

REES, T., STOLL, M. & WATHEN, A. (2010a) All-at-once preconditioners for PDE-constrained optimization. *Kybernetika*, **46**, 341–360.

REES, T., DOLLAR, H. S. & WATHEN, A. J. (2010b) Optimal solvers for PDE-constrained optimization. *SIAM Journal on Scientific Computing*, **32**, 271–298.

REES, T. & STOLL, M. (2010) Block-triangular preconditioners for pde-constrained optimization. *Numerical Linear Algebra with Applications*, **17**, 977–996.

REES, T. & WATHEN, A. (2010) Preconditioning iterative methods for the optimal control of the Stokes equation. *Submitted*.

SAAD, Y. (2003) *Iterative methods for sparse linear systems*. Philadelphia, PA: Society for Industrial and Applied Mathematics, pp. xviii+528.

SCHÖBERL, J. & ZULEHNER, W. (2007) Symmetric indefinite preconditioners for saddle point problems with applications to pde-constrained optimization problems. *SIAM J. Matrix Anal. Appl*, **29**, 752–773.

SIMONCINI, V. (2011) Reduced order solution of structured linear systems arising in certain PDE-constrained optimization problems. *submitted*.

STOLL, M. & WATHEN, A. (2009) Preconditioning for PDE-constrained optimization with control constraints. *Submitted*.

STOLL, M. & WATHEN, A. (2010) All-at-once solution of time-dependent PDE-constrained optimization problems. *Submitted*.

STOLL, M. & WATHEN, A. (2011) All-at-once solution of time-dependent stokes control. *In preparation*.

TAKACS, S. & ZULEHNER, W. (2011) Convergence analysis of multigrid methods with collective point smoothers for optimal control problems. *Preprint JKU Linz*.

TRÖLTZSCH, F. (2005) *Optimale Steuerung partieller Differentialgleichungen: Theorie, Verfahren und Anwendungen*. Vieweg Verlag, Wiesbaden.

TRÖLTZSCH, F. (2010) *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*. Amer Mathematical Society.

ULBRICH, M. (2011) *Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems*. SIAM Philadelphia.

VANDEWALLE, S. & PIESSENS, R. (1992) Efficient parallel algorithms for solving initial-boundary value and time-periodic parabolic partial differential equations. *SIAM J. Sci. Statist. Comput*, **13**, 1330.