Martin Stoll Andy Wathen

All-at-once solution of time-dependent Stokes control

Max Planck Institute Magdeburg
Preprints

MPIMD/10-03 June 8, 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

All-at-once solution of time-dependent Stokes control

Martin Stoll[a], Andy Wathen[b]

[a] *Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg Germany, Tel.: +49 391 6110 384, Fax: +49 391 6110 500 (martin.stoll80@gmail.com)*
[b] *Numerical Analysis Group, Mathematical Institute, 24 - 29 St Giles', Oxford, OX1 3LB, United Kingdom, Tel.: +44 1865 615309, Fax: +44 1865 273583 (wathen@maths.ox.ac.uk)*

Abstract

The solution of time-dependent PDE-constrained optimization problems subject to unsteady flow equations presents a challenge to both algorithms and computers. In this paper we present an all-at-once approach where we solve for all time-steps of the discretized unsteady Stokes problem at once. The most desirable feature of this approach is that for all steps of an iterative scheme we only need approximate solutions of the discretized Stokes operator. This leads to an efficient scheme which exhibits mesh-independent behaviour.

Keywords: Saddle point problems, Unsteady Stokes equation, PDE-constrained optimization, Preconditioning

1. Introduction

The solution of complex flow problems is one of the most interesting and demanding problems in applied mathematics and scientific computing. Over the last decades the numerical solution of problems such as Stokes flow has received a lot of attention both from applied scientist and mathematicians alike. The discretization of the Stokes equation via finite elements [13, 1, 10] as well the efficient solution of the corresponding linear systems in saddle point form [13, 47, 40, 3] are well established. In recent years, with the advances of computing power and algorithms, the solution of optimal control problems with partial differential equation (PDE) constraints such as Stokes or Navier-Stokes flow problems have become a topic of great interest [22, 25, 35, 7, 12].

In this paper, we want to address the issue of efficiently solving the linear systems that arise when the optimal control of the time-dependent Stokes problem is considered. We here want to employ the so-called one-shot approach, which is a technique previously used in [23, 24, 5, 43]. In detail, the discretization of the problem is constructed in the space-time domain and then solved for all time-steps at once. We will come back to this later.

The paper is organized as follows, we first discuss the control problem and how it can be discretized. In Section 3 we discuss the choice of the Krylov solver that should be employed. We then discuss the preconditioners for the various

parts of the saddle point problem. This is followed by numerical experiments for two different objective functions in both two and three space dimensions and time.

2. Problem and discretization

In the following we consider the tracking-type functional

$$J(y,u) = \frac{1}{2}\int_0^T \int_{\Omega_1} (y-\bar{y})^2 \, dxdt + \frac{\beta}{2}\int_0^T \int_{\Omega_2} (u)^2 \, dxdt + \frac{\gamma}{2}\int_{\Omega_1} (y(T)-\bar{y}(T))^2 \, dx \tag{1}$$

where $\Omega_{1/2} \subseteq \Omega$ are bounded domains in $\mathbb{R}^d$ with $d = 2, 3$. Additionally, for the state y and the control u the time-dependent Stokes equation has to be satisfied

$$y_t - \nu\triangle y + \nabla p = u \text{ in } [0,T]\times\Omega \tag{2}$$
$$-\nabla \cdot y = 0 \text{ in } [0,T]\times\Omega \tag{3}$$
$$y(t,.) = g(t) \text{ on } \partial\Omega, t \in [0,T] \tag{4}$$
$$y(0,.) = y_0 \text{ in } \Omega, \tag{5}$$

with y the state representing the velocity and p the pressure. Here, $\bar{y}$ is the so-called desired state. The goal of the optimization is to compute the control u in such a way that the state y will be as close as possible to $\bar{y}$. One might impose additional constraints both on the control u and the state y. One of the most common constraints in practice are the so-called box constraints given by

$$u_a \leq u \leq u_b \text{ and } y_a \leq u \leq y_b.$$

We will discuss these later and for now assume that no-such constraints are imposed.

There are two techniques used to solve the above problem. The first is the so-called *Discretize-then-Optimize* approach and the second is the *Optimize-then-Discretize* approach. In the first problem, both the functional and the PDE are discretized before the optimization typically via a Lagrange multiplier approach is employed. Optimize-then-Discretize will first build the KKT system for the infinite dimensional problem. It is desirable to have both approaches lead to the same discrete optimality system.

We begin by considering the first order conditions of the above problem and get the forward problem described in (2) to (5), the relation

$$\beta u + \lambda = 0 \tag{6}$$

referred to as the gradient equation as well as the adjoint PDE

$$-\lambda_t - \nu\triangle\lambda + \nabla\xi = y - \bar{y} \text{ in } [0,T]\times\Omega \tag{7}$$
$$-\nabla \cdot \lambda = 0 \text{ in } [0,T]\times\Omega \tag{8}$$
$$y(t,.) = 0 \text{ on } \partial\Omega, t \in [0,T] \tag{9}$$
$$y(0,.) = \gamma\left(y(T) - \bar{y}(T)\right) \text{ in } \Omega, \tag{10}$$

(for more information see [46, 45, 23, 24]).

The question is now whether we can find a discretization-scheme such that the Discretize-then-Optimize and the Optimize-then-Discretize approach coincide. One central question is how to discretize the above problem. We start by using a backward Euler scheme in time to obtain for the forward Stokes problem

$$\frac{y^k - y^{k-1}}{\tau} - \nu \triangle y^k + \nabla p^k = u^k \tag{11}$$

$$-\nabla \cdot y^k = 0 \tag{12}$$

and similarly for the adjoint PDE we get

$$\frac{\lambda^k - \lambda^{k+1}}{\tau} - \nu \triangle \lambda^k + \nabla \xi^k = y^k - \bar{y}^k \tag{13}$$

$$-\nabla \cdot \lambda^k = 0 \tag{14}$$

and for those two relations, after a finite element space discretization, the matrix representations look like the following

$$\frac{M\mathbf{y}^k - M\mathbf{y}^{k-1}}{\tau} + \nu K \mathbf{y}^k + B^T \mathbf{p}^k = M\mathbf{u}^k \tag{15}$$

$$B\mathbf{y}^k = 0 \tag{16}$$

and

$$\frac{M\boldsymbol{\lambda}^k - M\boldsymbol{\lambda}^{k+1}}{\tau} + \nu K \boldsymbol{\lambda}^k + B^T \boldsymbol{\xi}^k = M\mathbf{y}^k - M\bar{\mathbf{y}}^k \tag{17}$$

$$B\boldsymbol{\lambda}^k = 0, \tag{18}$$

with M being the mass matrix (for the sake of simplicity we assume M to be the lumped mass matrix) and K the finite element stiffness matrix. We will later use M_p for the mass matrix on the pressure space but refrain from adding the index y to the mass matrix on the velocity space.

The appropriate one-shot form for the forward PDE is now given by

$$\begin{bmatrix} \mathcal{L} & 0 & 0 & 0 & 0 \\ -\mathcal{M}_0 & \mathcal{L} & 0 & 0 & 0 \\ 0 & -\mathcal{M}_0 & \mathcal{L} & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -\mathcal{M}_0 & \mathcal{L} \end{bmatrix} \begin{bmatrix} \mathbf{y}^0 \\ \mathbf{p}^0 \\ \mathbf{y}^1 \\ \mathbf{p}^1 \\ \vdots \\ \mathbf{y}^N \\ \mathbf{p}^N \end{bmatrix} \tag{19}$$

$$-\tau \begin{bmatrix} M & 0 & 0 & \\ 0 & 0 & 0 & 0 \\ 0 & M & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & M \end{bmatrix} \begin{bmatrix} \mathbf{u}^0 \\ \mathbf{u}^1 \\ \vdots \\ \mathbf{u}^N \end{bmatrix} = \begin{bmatrix} L\mathbf{y}^0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \tag{20}$$

$$\mathcal{K}\mathbf{y}^+ - \tau\mathcal{N}\mathbf{u}^+ = d, \tag{21}$$

where $\mathcal{L} = \begin{bmatrix} L & B^T \\ B & 0 \end{bmatrix}$, $L = \tau^{-1}M + K$ and $\mathcal{M}_0 = \text{blkdiag}(\tau^{-1}M, 0)$.

The scheme presented by (15) and (19) represents a discretization of the forward PDE, as already pointed out in [43] the adjoint of (19) will represent the time-evolution described by (17) but the initial condition for the adjoint PDE might make for a difference between the Discretize-then-Optimize and Optimize-then-Discretize approaches.

In more detail, the discretization of the functional $J(y, u)$ (using $\mathbf{y}_i$, $\mathbf{u}_i$, $\bar{\mathbf{y}}_i$ for $i = 0, \ldots, N$ at the different time-steps) via a rectangle rule for the time and finite elements for the space leads to

$$\begin{aligned} J(\mathbf{u}, \mathbf{y}) = &\frac{\tau}{2} \sum_{i=1}^{N} (\mathbf{y}_i - \bar{\mathbf{y}}_i)^T M (\mathbf{y}_i - \bar{\mathbf{y}}_i) + \frac{\beta\tau}{2} \sum_{i=1}^{N} \mathbf{u}_i^T M_u \mathbf{u}_i \\ &+ \frac{\gamma}{2} (\mathbf{y}_N - \bar{\mathbf{y}}_N)^T M_u (\mathbf{y}_N - \bar{\mathbf{y}}_N). \end{aligned} \tag{22}$$

Note that N denotes the number of time-steps. In fact, we are using a slightly different approximation

$$\begin{aligned} J(\mathbf{u}, \mathbf{y}) = &\frac{\tau}{2} \sum_{i=0}^{N} (\mathbf{y}_i - \bar{\mathbf{y}}_i)^T M (\mathbf{y}_i - \bar{\mathbf{y}}_i) + \frac{\beta\tau}{2} \sum_{i=0}^{N} \mathbf{u}_i^T M_u \mathbf{u}_i \\ &+ \frac{\gamma}{2} (\mathbf{y}_N - \bar{\mathbf{y}}_N)^T M_u (\mathbf{y}_N - \bar{\mathbf{y}}_N), \end{aligned} \tag{23}$$

which will later give that the approaches optimize-then-discretize and discretize-then-optimize coincide. Note that this changes the functional $J(y, u)$ only by

constant terms involving the initial values for y_0, $\bar{y}_0$ and u_0 but the location of the minimum will not be changed.

The Lagrangian of the discrete problem can now be written as

$$L(\mathbf{y}^+, \mathbf{u}, \boldsymbol{\lambda}^+) = J(\mathbf{y}, \mathbf{u}) + \left(\boldsymbol{\lambda}^+\right)^T \left(-\mathcal{K}\mathbf{y}^+ + \tau\mathcal{N}\mathbf{u} + d\right) \tag{24}$$

where we use $\mathbf{y}^+ = [\mathbf{y}_0, \mathbf{p}_0, \ldots, \mathbf{y}_N, \mathbf{p}_N]$ and similarly for $\boldsymbol{\lambda}^+ = [\boldsymbol{\lambda}_0, \boldsymbol{\xi}_0, \ldots, \boldsymbol{\lambda}_N, \boldsymbol{\xi}_N]$. The first order or KKT conditions for $L(\mathbf{y}^+, \mathbf{u}, \boldsymbol{\lambda}^+)$ are now given by the following system

$$\begin{bmatrix} \tau\mathcal{M} & 0 & -\mathcal{K}^T \\ 0 & \beta\tau\mathcal{M}_u & \tau\mathcal{N}^T \\ -\mathcal{K} & \tau\mathcal{N} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}^+ \\ \mathbf{u}^+ \\ \boldsymbol{\lambda}^+ \end{bmatrix} = \begin{bmatrix} \mathcal{M}\bar{\mathbf{y}}^+ \\ 0 \\ d \end{bmatrix}. \tag{25}$$

We will discuss appropriate solvers and possible preconditioners for system (25) in Section 3.

Our aim now is to discuss the Optimize-then-Discretize approach and how to ensure it coincides with the Discretize-then-Optimize. Here we follow the results presented in [23] for the Stokes equation (see [24] for Navier-Stokes). Hinze *et al.* start with the infinite-dimensional KKT system. A straightforward discretization of the infinite dimensional problems will in general not result in agreement of both optimization-discretization approaches. The technique introduced in [23] starts with a time-discretization of the forward problem as done in (11) and adjusting the initial condition to be

$$y_0 - \tau\triangle y_0 = y_0 - \tau\triangle y_0.$$

Writing down the Lagrangian for the semi-discretized problem we obtain the following system

$$\frac{y^k - y^{k-1}}{\tau} - \nu\triangle y^k + \nabla p^k = u^k \tag{26}$$

$$-\nabla \cdot y^k = 0 \tag{27}$$

$$y_0 - \tau\triangle y_0 = y_0 - \tau\triangle y_0 \tag{28}$$

$$\beta u_k + \lambda_k = 0 \tag{29}$$

$$\frac{\lambda^k - \lambda^{k+1}}{\tau} - \nu\triangle\lambda^k + \nabla\xi^k = y^k - \bar{y}^k \tag{30}$$

$$-\nabla \cdot \lambda^k = 0 \tag{31}$$

$$\lambda_N - \tau\triangle\lambda_N = (\tau + \gamma)(y_N - \bar{y}_N). \tag{32}$$

Using standard mixed finite elements to discretize in space we obtain the same discrete first order system for the Optimize-then-Discretize approach as for the Discretize-then-Optimize procedure. Note that with the changes we made earlier to the discretization of the cost functional $J(y, u)$, we get agreement of the initial values of the Optimize-then-Discretize approach and the Discretize-then-Optimize procedure.

In addition to the above considered problem, we will also discuss the numerical solution of a PDE-constrained optimization problem that has an added pressure term in the functional $J(y,u)$, i.e.,

$$\begin{aligned} J(y,u) = \frac{1}{2}\int_0^T \int_{\Omega_1} (y-\bar{y})^2 \, dxdt + \frac{\nu}{2}\int_0^T \int_{\Omega_1} (p-\bar{p})^2 \, dxdt \\ + \frac{\beta}{2}\int_0^T \int_{\Omega} (u)^2 \, dxdt + \frac{\gamma}{2}\int_{\Omega_1} (y(T)-\bar{y}(T))^2 \, dx \end{aligned} \tag{33}$$

subject to the unsteady Stokes equation as shown above. Here p is the pressure and $\bar{p}$ is the desired pressure. The discretization follows the above procedure and the first order conditions $L(\mathbf{y}^+, \mathbf{u}, \boldsymbol{\lambda}^+)$ are now given by the following system

$$\begin{bmatrix} \tau\mathcal{M} & 0 & -\mathcal{K}^T \\ 0 & \beta\tau\mathcal{M}_u & \tau\mathcal{N}^T \\ -\mathcal{K} & \tau\mathcal{N} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{y}^+ \\ \mathbf{u}^+ \\ \boldsymbol{\lambda}^+ \end{bmatrix} = \begin{bmatrix} \mathcal{M}\bar{\mathbf{y}}^+ \\ 0 \\ d \end{bmatrix}, \tag{34}$$

where the only difference to the system given in (25) is the matrix $\mathcal{M} = \text{blkdiag}(M, \ldots, M, M_p, ..., M_p)$. Note that for reasons of convenience we will always use $\mathcal{M}$ as a notation for the $(1,1)$ block whether positive definite or not. We will specify when it is important to consider the two cases separately.

3. Krylov solver and preconditioning

After having derived the linear system corresponding to the solution of the optimal control problem, we now want to discuss how to solve this system efficiently. For a reasonable sized spatial discretization even in two dimensions a direct solver might run out of memory fairly quickly as the dimensionality of the overall system crucially depends on the temporal discretization. Hence, we dismiss the possibility of using a direct solver for the overall system and rather decide to employ Krylov-subspace solvers. Because of the nature of the problem, $\mathcal{A}$ being symmetric and indefinite, we will use MINRES [31] as it is often the method of choice for saddle point problems. In more detail, MINRES (and also other Krylov subspace solvers) will build up a Krylov subspace

$$\text{span}\left\{r_0, \mathcal{A}r_0, \mathcal{A}^2 r_0, \ldots, \mathcal{A}^{k-1} r_0\right\}$$

by multiplying with the system matrix at each step. The approximation to the solution of the linear system will then be computed such that the 2-norm (in the unpreconditioned case) of the residual $\|r_k\|_2$ is minimized over the current Krylov subspace. Naturally, MINRES will only be used with a preconditioner and we refer to [13] for implementation details. In order for the preconditioned system to maintain the symmetric and indefinite nature of the problem, we need the preconditioner to be symmetric and positive definite. Hence, our choice will be a symmetric block-diagonal preconditioner. Before we mention the details of the preconditioner we will discuss alternative choices for the iterative scheme.

In case the upper-left block (blkdiag($\tau\mathcal{M}, \beta\tau\mathcal{M}_u$) is positive-definite, as is the case for the added pressure term or the forward Stokes problem, we could employ a non-standard CG method also known as the Bramble-Pasciak CG [8], which also has been successfully used for optimal control problems [42, 34]. It is also possible to use the projected CG method [19] with the so-called constraint preconditioners [26], which was demonstrated to also work well for control problems [32, 39, 20, 49]. For the use of indefinite preconditioners we would have to use non-symmetric methods such as GMRES [38], BICG [14] or SQMR [15] but we will refrain from using these methods in the course of this paper. Benzi *et al.* use Krylov methods within the preconditioner, which means that as an outer method a flexible method such as FGMRES [37] has to be employed.

We will now discuss the choice of preconditioner best suited to be used with MINRES. Our choice is a block-diagonal preconditioner of the following form

$$\mathcal{P} = \begin{bmatrix} A_0 & 0 & 0 \\ 0 & A_1 & 0 \\ 0 & 0 & S_0 \end{bmatrix}, \tag{35}$$

where A_0 is an approximation to $\tau\mathcal{M}$, A_1 approximates the $(2,2)$-block of the saddle point system, which we can afford to invert in case the mass matrices are lumped, and S_0 is a Schur-complement approximation. First, we want to comment on the blocks involving mass matrices. If we decide to use a consistent mass matrix, good preconditioners are available; Namely, the Chebyshev semi-iteration [17, 18], which is an easy-to-use but nevertheless very efficient method for systems involving the mass matrix as illustrated in [48]. The blocks corresponding to the zero-bocks in $\tau\mathcal{M}$, can be approximated by αI with $\alpha > 0$ as was done in [5].

The choice of the Schur-complement approximation is more tricky as the $(1,1)$-block of $\mathcal{A}$ is semi-definite. Assuming for now that $\tau\mathcal{M}$ is definite, the Schur-complement of the system matrix would look like the following

$$\tau^{-1}\mathcal{K}\mathcal{M}^{-1}\mathcal{K}^T + \tau^{-1}\beta^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T. \tag{36}$$

We will now use an approach presented in [32, 33, 34, 35] where we drop the second term ($\tau^{-1}\beta^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T$) in (36). Hence, our approximation to the Schur-complement at the moment is given by

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

We will discuss this now in more detail.

The Schur-complement approximation

Steady Stokes control

In this section we want to put a special emphasis on the approximation of the Schur-complement of the saddle point system (25). We emphasize the fact that the expression

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

involves two block-triangular matrices, which will be important as a preconditioned Krylov subspace solver will be using the inverse of the Schur-complement approximation $\mathcal{K}^{-T}\mathcal{M}\mathcal{K}^{-1}$. Both expressions $\mathcal{K}^{-T}$ and $\mathcal{K}^{-1}$ are relatively easy to evaluate as we can simply apply a backward and a forward substitution, respectively. Recalling the structure of $\mathcal{K}$ (see (19)) this means to "simply" solve an unsteady Stokes-system, which in itself poses a significant computational task. For the simpler problem of the state equation being the heat equation the authors suggest in [43] that one can replace the solution with the discretized PDE operator by an appropriately chosen algebraic multigrid (AMG) preconditioner. We also want to do this for the Stokes problem but as already pointed out in [35] the approximation of the Schur-complement in case of the Stokes problem is more involved than for the simpler heat equation. In [35] the authors show that a preconditioner for the Schur-complement, namely the block-diagonal preconditioner $\mathcal{P} = \text{blkdiag}(A_0, M_p)$, is a good preconditioner for the forward Stokes equation but in the case of Stokes control where a fourth-order operator has to be approximated, the contraction of the block-diagonal preconditioner is not sufficient for the Schur-complement approximation of the control problem. Hence, Rees and Wathen suggest the use of an inexact Uzawa method using a block-triangular preconditioner

$$\mathcal{P} = \begin{bmatrix} A_0 & 0 \\ B & -M_p \end{bmatrix},$$

where A_0 is an approximation to the discretized Laplacian, in general a multigrid operator and M_p the mass matrix on the pressure space (see [13]). A fixed number of Uzawa steps to approximate the discrete Stokes operator represents a linear operator and provides a good enough contraction rate such that the approximation to the Schur-complement will be sufficient to guarantee convergence of the overall outer MINRES iteration. For the unsteady case this was shown in [35]. Algorithm 1 shows a version of the inexact Uzawa method. Note that in the case of enclosed flow the Stokes-system matrix will be singular due to the hydrostatic pressure [13] but a consistent right-hand-side still enables the use of iterative solvers. As we need to apply a forward and a backward solve with the inexact Uzawa method a scaling[1] to make the right-hand-side sufficiently close to a consistent right-hand-side always worked very well in our numerical experiments. In the case the Stokes system is invertible these issues do not arise.

Time-dependent Stokes control

We will now describe how to approximate the Schur-complement in the case of time-dependent Stokes control. We note that the Schur-complement approximation

$$\hat{\mathcal{K}}\mathcal{M}\hat{\mathcal{K}^T}$$

[1] In Matlab notation: Scaling b such that Pb is close to a consistent right hand side, with $\text{P} = \text{speye(n)} - \frac{\alpha}{\text{n}}\text{ones(n, n)}$ with α close to one, e.g. 0.9.

1: Select x_0.
2: **for** $k = 1, 2, \ldots$ **do**
3: $\quad x_{k+1} = x_k + \mathcal{P}^{-1}(b - \mathcal{A}x_k)$
4: **end for**

Algorithm 1: Inexact Uzawa method

involves a forward and backward substitution where we have to approximate the inverse of the matrix

$$\begin{bmatrix} \tau^{-1}M + K & B^T \\ B & 0 \end{bmatrix} \tag{37}$$

for the evaluation of $\hat{\mathcal{K}}$ and $\hat{\mathcal{K}}^T$ at each time-step. We propose to use the inexact Uzawa algorithm (see Algorithm 1) for the matrix (37) with a block-triangular preconditioner defined as

$$\mathcal{P} = \begin{bmatrix} \hat{A} & 0 \\ B & -\hat{S} \end{bmatrix}.$$

We can now simply use an algebraic or geometric multigrid for the preconditioner $\hat{A}$ approximating $M + \tau K$ but the choice of $\hat{S}$ is not so straightforward. In the case of steady Stokes problem the pressure mass matrix will allow for a suitable approximation to the Schur-complement. In our case, we have a different $(1,1)$-block to the steady case and we derive a suitable preconditioner using a technique for the steady Navier-Stokes equation. We follow [13] by looking at the commutator

$$\mathcal{E} = (\mathcal{L})\nabla - \nabla(\mathcal{L})_p$$

where $\mathcal{L} = I + \tau\triangle$. Using the finite element method we obtain the discretization of the discrete operators (see [13, Chapter 8.2]) and put this into the discretized version of the above to get

$$\mathcal{E}_h = (M^{-1}L)M^{-1}B^T - M^{-1}B^T(M_p^{-1}L_p)$$

where $L = M + \tau K$. We now pre-multiply the last equation by $BL^{-1}M$ and post-multiply by $L_p^{-1}M_p$ to get

$$BM^{-1}B^TL_p^{-1}M_p - BL^{-1}B^T. \tag{38}$$

Under the assumption that the commutator is small the last equation gives

$$BM^{-1}B^TL_p^{-1}M_p \approx BL^{-1}B^T, \tag{39}$$

which will allow us to create a suitable Schur-complement approximation. In fact, $BM^{-1}B^TL_p^{-1}M_p$ will be used. The first term $BM^{-1}B^T$ cannot be used as it is but we rather use the fact that it is spectrally equivalent to the Laplacian formed on the pressure space to give

$$\hat{S} = K_pL_p^{-1}M_p. \tag{40}$$

Note that as we are only interested in the application of $\hat{S}^{-1}$ we can further obtain

$$\hat{S}^{-1} = M_p^{-1} L_p K_p^{-1} = M_p^{-1}(M_p + \tau K_p)K_p^{-1} = K_p^{-1} + \tau M_p^{-1}. \tag{41}$$

We are now able to provide efficient preconditioners for the solution of the time-dependent Stokes problem within the Uzawa method. The preconditioner $\hat{S}^{-1}$ was first derived in [11] by Cahouet and Chabard and is hence often referred to as the Cahouet-Chabard preconditioner. It was extensively used, analyzed and extended to for example the Navier-Stokes case (more information can be found in [4, 27, 9, 6, 28]). $\hat{A}$ will in our case be an algebraic multigrid method applied to $\tau^{-1}M + K$ and for $\hat{S}^{-1}$ we need the approximation to K_p^{-1}, which can be done using an algebraic multigrid as well, and additionally we need to approximate M_p^{-1}, which can be efficiently approximated using the Chebyshev semi-iteration [17, 18, 48] (see Algorithm 2).

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

Algorithm 2: Chebyshev semi-iterative method for a number of l steps

3.1. Eigenvalue analysis

In this section, we will try and give bounds for the eigenvalues of the preconditioned matrix. We closely follow an earlier analysis presented in [43].

We now want to give bounds on the eigenvalues of the preconditioned case for a somewhat idealized case. We assume that the preconditioner is given by

$$\mathcal{P} = \begin{bmatrix} \hat{\mathcal{M}} & 0 & 0 \\ 0 & \beta\tau\mathcal{M}_u & 0 \\ 0 & 0 & S_0 \end{bmatrix}$$

with $S = \mathcal{K}\hat{\mathcal{M}}_2\mathcal{K}^T$. A similarity transformation $\mathcal{P}^{-1/2}\mathcal{A}\mathcal{P}^{-1/2}$, now reveals the following matrix

$$\mathcal{P}^{-1/2}\mathcal{A}\mathcal{P}^{-1/2} = \begin{bmatrix} D & 0 & B_1^T \\ 0 & I & B_2^T \\ B_1 & B_2 & 0 \end{bmatrix} \tag{42}$$

where with $B_1 = S_0^{-1/2}\mathcal{K}\hat{\mathcal{M}}^{-1/2}$, and $B_2 = \tau^{-1/2}\beta^{-1/2}S_0^{-1/2}\mathcal{N}\mathcal{M}_u^{-1/2}$. We will switch to the notation $A = \text{blkdiag}(D, I)$ and $B = [B_1 B_2]$ as for the classical saddle point problem. It is a well-known result [36] that for such a saddle point problem with symmetric and positive-definite $(1,1)$-block the eigenvalues of $\mathcal{P}^{-1/2}\mathcal{A}\mathcal{P}^{-1/2}$ lie in the intervals

$$\mathcal{I}^- = \left[\frac{1}{2}\left(\lambda_{\min}^{(A)} - \sqrt{\left(\lambda_{\min}^{(A)}\right)^2 + \sigma_{\max}^2}\right), \frac{1}{2}\left(\lambda_{\max}^{(A)} - \sqrt{\left(\lambda_{\max}^{(A)}\right)^2 + \sigma_{\max}^2}\right)\right]$$

and

$$\mathcal{I}^+ = \left[\lambda_{\min}^{A}, \frac{1}{2}\left(\lambda_{\max}^{(A)} + \sqrt{\left(\lambda_{\max}^{(A)}\right)^2 + \sigma_{\max}^2}\right)\right].$$

This is true for both problems presented here. In the case of the pressure included in the objective function (33) the resulting saddle point system (34) has a positive definite $(1,1)$ block, which will lead to $\lambda_{\min}^{A} > 0$. We will now discuss the bounds for $\mathcal{I}^+$ and $\mathcal{I}^-$ in more detail. In both cases, we need bounds for eigenvalues of A. The structure of A reveals that we have an identity block and the matrix $D = \text{blkdiag}(I, 0, I, 0, \ldots, I, 0)$ for the objective function (1) and $D = \text{blkdiag}(I, I_p, I, I_p, \ldots, I, I_p)$ for (33). It is now easy to read off the eigenvalues of D. The estimation of the singular values of B is a bit more involved and we use the fact that the eigenvalues of BB^T are the square of the singular values of B. The structure of B now gives

$$BB^T = B_1B_1^T + B_2B_2^T = S_0^{-1/2}\mathcal{K}\hat{\mathcal{M}}^{-1}\mathcal{K}^T S_0^{-1/2} + \tau^{-1}\beta^{-1}S_0^{-1/2}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T S_0^{-1/2}$$

and note that the last matrix is similar to

$$S_0^{-1}\left(\mathcal{K}\hat{\mathcal{M}}^{-1}\mathcal{K}^T + \tau^{-1}\beta^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T\right).$$

This indicates that if S_0^{-1} is chosen to be $\mathcal{K}^{-T}\hat{\mathcal{M}}\mathcal{K}^{-1}$ the above takes the following form

$$\left(I + \tau^{-1}\beta^{-1}\mathcal{K}^{-T}\hat{\mathcal{M}}\mathcal{K}^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T\right). \tag{43}$$

Similar equations to (43) have be analyzed before for non-time-dependent problems [32, 44] where we now have to show that for a more refined mesh, smaller mesh parameter h, the eigenvalues of $\mathcal{K}^{-T}\hat{\mathcal{M}}\mathcal{K}^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T$ do not change. We are at this stage not able to prove the mesh-independence of the term $\tau^{-1}\beta^{-1}\mathcal{K}^{-T}\hat{\mathcal{M}}\mathcal{K}^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T$. In general the term $\hat{\mathcal{M}}$ will include a multiplication by τ which removes the dependency of the eigenvalue bounds on τ since no other matrix involves τ. In Figure 1 we show the largest 100 eigenvalues computed by the MATLAB `eigs` command of the matrix $\mathcal{K}^{-T}\hat{\mathcal{M}}\mathcal{K}^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T$ where $\hat{\mathcal{M}} = \text{blkdiag}(M, 1e-6I, \ldots, M, 1e-6I)$ and the Stokes problem is for simplicity chosen with a Neumann boundary at the bottom and Dirichlet on the remaining sides of the domain to have an invertible Stokes matrix[2]. It can be

[2]Note that in the enclosed flow case we have a one-dimensional kernel and this cannot be used for the illustration in Figure 1.

seen that for two relatively small meshes with the DoF for one Stokes system given by $n = 578, m = 81$ and for the second Stokes system $n = 162, m = 25$, where n is the number of discrete velocity variables and m the number of discrete pressure variables. Note that these are the degrees of freedom for one instance of the unsteady problem. We chose a fixed number of time-steps $N = 10$ and see that the eigenvalues for these problems do not depend on h; we expect this behaviour to continue for smaller h as in our numerical experiments (see Section 4) we do not observe mesh-dependent behaviour.

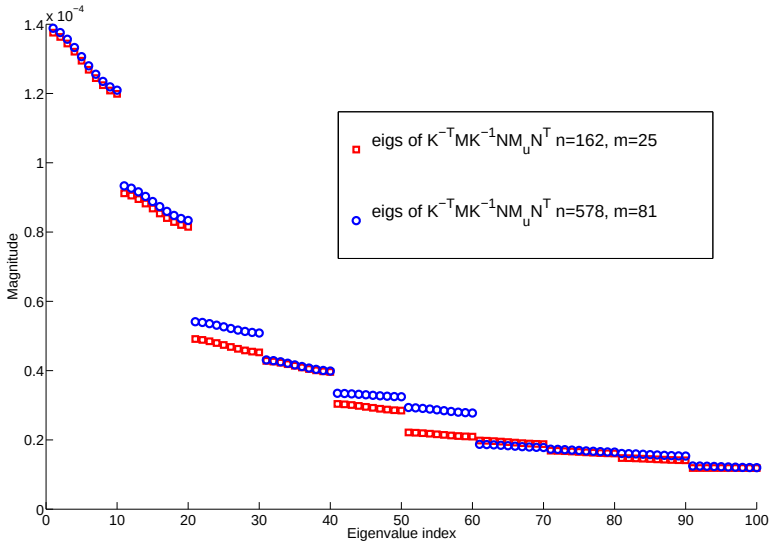


Figure 1: Largest 100 eigenvalues of $\mathcal{K}^{-T}\hat{\mathcal{M}}\mathcal{K}^{-1}\mathcal{N}\mathcal{M}_u^{-1}\mathcal{N}^T$ for two small problems.

4. Numerical experiments

The numerical tests are all performed using deal.II [2] with algebraic multigrid ML Trilinos preconditioner for the appropriate blocks [16]. We use a **Q2/Q1** discretization of the Stokes problem. We use 4 steps of the Uzawa method for each approximation of the PDE operator. The various blocks are approximated as follows the pressure Laplacian is approximated by 2 steps of an AMG V-cycle and 20 steps of a Chebyshev smoother, the Laplacian plus mass matrix block is approximated by 20 steps of a Chebyshev smoother and 3 steps of an AMG V-cycle. In general we use a relative tolerance of 10^{-4} for the pseudo-residual and mention explicitly if any other tolerance is used. All experiments are performed on a Centos Linux machine with Intel(R) Xeon(R) CPU X5650 @ 2.67GHz CPUs and 48GB of RAM. The example we look at in

this section is taken from the paper by Hinze *et al.* [23]. The spatial domain is defined as $\Omega = [0,1]^d$ and the time domain is given as $[0,1]$. As we have not used special multigrid methods devised for parameter-dependent problems it has been observed in [29] that general purpose preconditioners might loose the independence with respect to τ. This behaviour could not be observed if τ scaled with the mesh-parameter and hence we are often choosing $\tau \approx h$. We will also present results for a fixed $\tau = 0.05$. We begin describing the target flow $\bar{y}$. The target flow is the solution for the unsteady Stokes equation with Dirichlet boundary conditions, i.e. $y = (1,0)$ when the second spatial component $x_2 = 1$ and $y = (0,0)$ on the remaining boundary for the two-dimensional case and $y = (1,0,0)$ similar to before in the three-dimensional case. The viscosity was chosen to be $\nu = 1$. Figure 2 shows the desired state at $t = 0.5$. For the control problem we now consider the following boundary conditions. For the top-boundary where $x_2 = 1$ we get $y = (1 + \frac{1}{2}\cos(4\pi t - \pi), 0)$ and zero elsewhere in two space dimensions. For this example the viscosity is set to $1/100$. We also take $\gamma = 0$. Figure 3 shows both the computed controlled state and the uncontrolled state for the above system at $t = 0.5$. For the choice of the scaling parameter α dealing with the zero-blocks in the $(1,1)$ block of the saddle point system we follow a strategy proposed in [41] where $\alpha = \|\beta\tau\mathcal{M}_u\|$. To illustrate the performance of our preconditioner it is imperative to consider three-dimensional results and we choose for $x_2 = 1$ we get $y = (t + sin(0.1x_1), t + cos(0.5x_2), 0)$, and $nu = \frac{1}{100}$, which is a somewhat arbitrary choice but nevertheless exhibits all the complications expected in a realistic problem.

4.1. Without pressure term

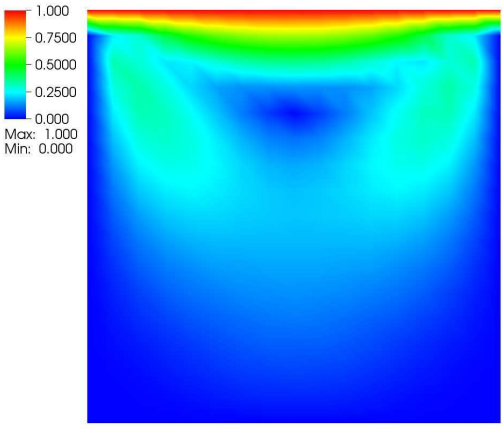
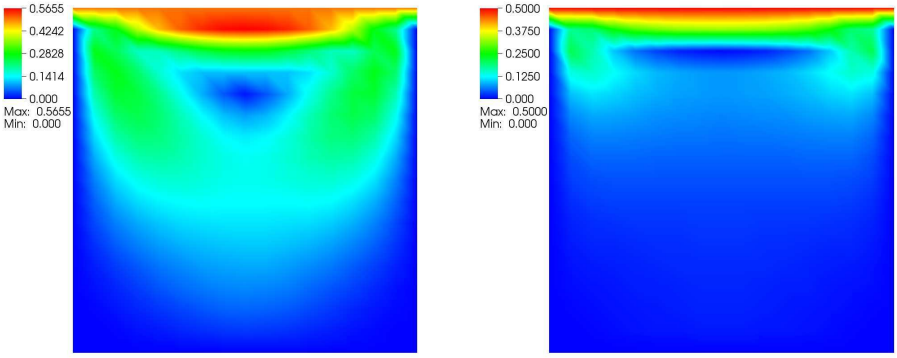


Figure 2: Desired state at $t = 0.5$.



(a) State at $t = 0.5$ (b) Uncontrolled state at $t = 0.5$

Figure 3: Uncontrolled vs. controlled state

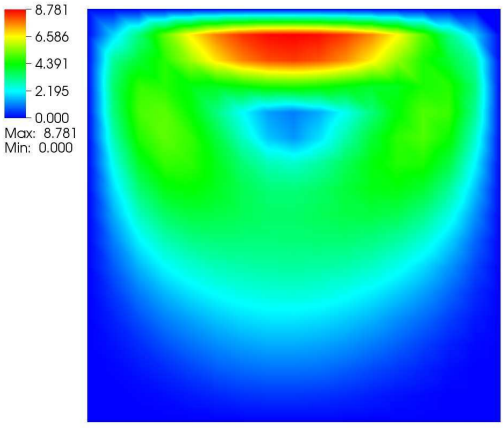


Figure 4: Control at $t = 0.5$.

We begin our numerical experiments by computing the approximate solution to the above problems on a variety of meshes. Figure 3 shows the controlled and the uncontrolled state at the time $t = 0.5$. The control for this time is shown in Figure 4. We denote by DoF the degrees of freedom used for the discretization of the PDE at one time-step, similar to a steady problem, then we show the number of time-steps N. This means that for the smallest mesh we are implicitly solving a linear system of dimension $3 * 37507 * 129 \approx 14$ million unknowns. Timings are given in seconds. Table 1 shows the results for a relative tolerance of 10^{-4} for the case when $\tau \approx h$ and in Table 2 we show the results for the same setup just the number of time-steps is now fixed. As can be seen from the results in Table 1 the iteration numbers do not increase with refinement of both t and h. The reduction of the iteration numbers might be explained by the fact that as τ scales like h, this means that the small eigenvalues of the mass matrix contribute less and the small eigenvalues look more similar to the small eigenvalues of the steady problem. The results in Table 2 show that for a fixed number of time-steps the iteration numbers are more or less constant although higher than in the case of a time-step that scales with the mesh-parameter.

DoF	N	MINRES	Time
2467	33	37	251
9539	65	14	765
37507	129	11	4970

Table 1: Number of MINRES steps with CPU-time $\tau \approx h$.

DoF	N	MINRES	Time
2467	21	42	179
9539	21	27	449
37507	21	32	2565

Table 2: Number of MINRES steps with CPU-time $\tau = 0.05$.

We noted that for the same number of Uzawa steps (chosen as 4-steps) the convergence to a smaller tolerance was sometimes slower than expected. This might be due to the fact that the Schur-complement approximation is not of good enough quality, which can lead to stagnation of MINRES (see [30]). One possibility would be to use deflated MINRES as proposed in [30], which would need approximations of some eigenvalues of the Schur-complement. For our examples it was sufficient to increase the number of Uzawa steps to 6 and the results are shown in Table 3. The results shown in Table 4 are computed for a tolerance of $1e-4$ and the $\tau \approx h$ with the setup as in the above extended to three dimensions. Again, we see a very moderate number of MINRES iterations for this case. Also here we used an increased number of inexact Uzawa steps, i.e., 8 steps seemed sufficient for all our tests. We show results for the

DoF	N	MINRES	Time
2467	33	34	340
9539	65	19	1500
37507	129	15	10368

Table 3: Number of MINRES steps with CPU-time $\tau \approx h$ and $tol = 1e-6$.

three-dimensional problem in Figure 5 shows the computed state, Figure 6 the uncontrolled state and Figure 7 the desired state.

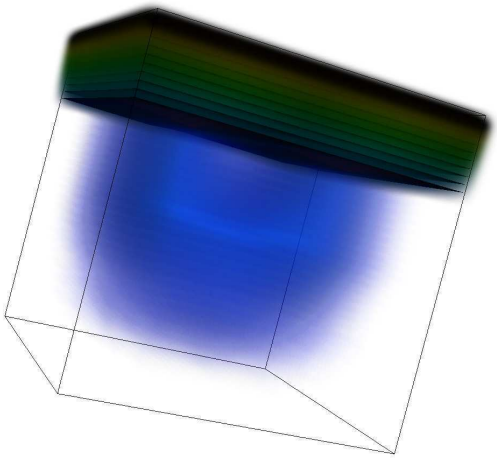


Figure 5: Computed state at $t = 0.5$.

DoF	N	MINRES	Time
2312	9	58	403
15468	17	33	4289

Table 4: Number of MINRES steps with CPU-time $\tau \approx h$ in $3D$.

Finally, in Table 5 we compute a $2D$ solution with tolerance $1e-4$ and the above setup and only change the viscosity ν to be equal to one. Again, the iteration numbers are very low.

4.2. With pressure term

In this section we are going to show results for the problem including a pressure term in the objective function. The desired state and the desired pressure

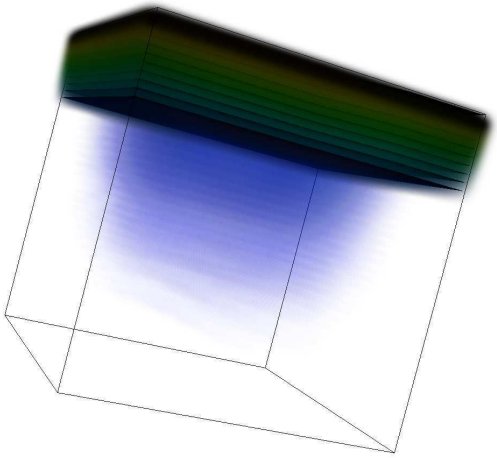


Figure 6: Uncontrolled state at $t = 0.5$.

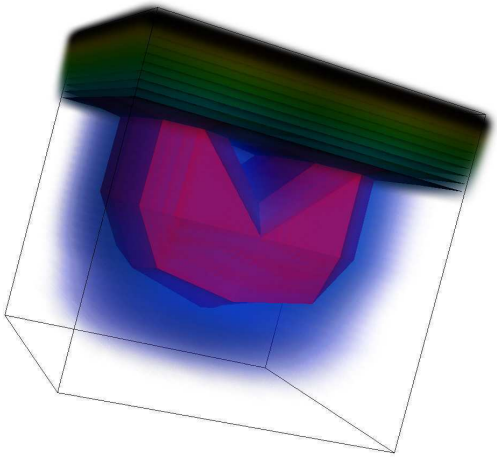


Figure 7: Desired state at $t = 0.5$.

DoF	N	MINRES	Time
2467	33	8	89
9539	65	10	1161
37507	129	15	15699

Table 5: Number of MINRES steps with CPU-time $\tau \approx h$ and $\nu = 1$.

are obtained from solving the previously mentioned unsteady flow problem. In our case we now simply invert the mass matrix coming from the velocity space (as it is lumped) and use the Chebyshev semi-iteration for the mass matrix on the pressure space that corresponds to the pressure terms in the objective function, i.e., 20 steps of this method are typically employed. Table 6 shows the results for a three-dimensional setup with tolerance at $1e-4$. The two-dimensional results are shown in Table 7. In both cases, we have chosen τ to scale like the mesh-parameter h and similar to before we observe lower iteration numbers for finer meshes.

DoF	N	MINRES	Time
2312	9	91	408
15468	17	72	5952
112724	33	38	63291

Table 6: Number of MINRES steps with CPU-time $\tau \approx h$ in $3D$.

DoF	N	MINRES	Time
2467	33	25	188
9539	65	14	997
37507	129	12	4134

Table 7: Number of MINRES steps with CPU-time $\tau \approx h$.

5. Conclusions and future work

We have shown that the discretization of the PDE-constrained optimal control problem involving unsteady Stokes flow as a PDE constraint can be efficiently cast using a Lagrangian technique into a all-at-once saddle point problem. As the dimension of these type of problems are vast the use of iterative solvers is imperative. We proposed to use MINRES as the outer solver and proposed block preconditioners. The approximation of the Schur-complement can efficiently be approximated using an inexact Uzawa method for which we have shown that the well-known Cahouet-Chabard preconditioner can be used

here. The iteration numbers for the outer MINRES method are always very low and in some cases even show a benign mesh-dependence, i.e., iteration numbers were smaller for discrete problems of larger dimension.

We believe that the results for the computation of the Stokes control problem will be very similar to the ones presented here if control constraints are present. In that case an outer Newton-type [21] method can be used and the linear systems that have to be solved at each step of the active set iteration are similar in nature to the ones for the problem with no bound constraints [42]. It might also be good to apply a nested approach where the solution is first approximated on a coarse mesh and then transfered to a fine discretization (see [20]). Of course, these problems should also be analyzed and tested numerically in the future.

Acknowledgement

The first author would like to thank Michael Köster for his advice on the discretization of the control problem. He would also like to thank Michele Benzi and Kent-Andre Mardall for pointing out references for the Cahouet-Chabard preconditioner.

References

[1] D. Arnold, F. Brezzi, M. Fortin, A stable finite element for the Stokes equations, Calcolo 21 (1984) 337–344.

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

[3] R. Bank, B. Welfert, H. Yserentant, A class of iterative methods for solving saddle point problems, Numerische Mathematik 56 (1989) 645–666.

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

[5] M. Benzi, E. Haber, L. Taralli, A preconditioning technique for a class of pde-constrained optimization problems, To appear in Advances in Computational Mathematics (2010).

[6] M. Benzi, M.A. Olshanskii, Z. Wang, Modified augmented Lagrangian preconditioners for the incompressible NavierStokes equations , Int. J. Numer. Meth. Fluids (2010).

[7] A. Borzi, K. Ito, K. Kunisch, An optimal control approach to optical flow computation, Int. J. Numer. Meth. Fluids 40 (2002).

[8] J.H. Bramble, J.E. Pasciak, A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems, Math. Comp 50 (1988) 1–17.

[9] J.H. Bramble, J.E. Pasciak, Iterative techniques for time dependent Stokes problems, Computers & Mathematics with Applications 33 (1997) 13–30.

[10] F. Brezzi, M. Fortin, Mixed and hybrid finite element methods, Springer Series in Computational Mathematics, Springer-Verlag, New York, 1991.

[11] J. Cahouet, J. Chabard, Some fast 3D finite element solvers for the generalized Stokes problem, International Journal for Numerical Methods in Fluids 8 (1988) 869–895.

[12] H. Choi, M. Hinze, K. Kunisch, Instantaneous control of backward-facing step flows, Applied Numerical Mathematics 31 (1999) 133–158.

[13] H.C. Elman, D.J. Silvester, A.J. Wathen, Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics, Numerical Mathematics and Scientific Computation, Oxford University Press, New York, 2005.

[14] R. Fletcher, Conjugate gradient methods for indefinite systems, in: Numerical analysis (Proc 6th Biennial Dundee Conf., Univ. Dundee, Dundee, 1975), Springer, Berlin, 1976, pp. 73–89. Lecture Notes in Math., Vol. 506.

[15] R.W. Freund, N.M. Nachtigal, A New Krylov-Subspace Method for Symmetric Indefinite Linear Systems, in: E.W.F. Ames (Ed.), Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics, IMACS, pp. 1253–1256.

[16] M. Gee, C. Siefert, J. Hu, R. Tuminaro, M. Sala, ML 5.0 Smoothed Aggregation User's Guide, Technical Report SAND2006-2649, Sandia National Laboratories, 2006.

[17] G.H. Golub, R.S. Varga, Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second order Richardson iterative methods. I, Numer. Math. 3 (1961) 147–156.

[18] G.H. Golub, R.S. Varga, Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second order Richardson iterative methods. II, Numer. Math. 3 (1961) 157–168.

[19] N.I.M. Gould, M.E. Hribar, J. Nocedal, On the solution of equality constrained quadratic programming problems arising in optimization, SIAM J. Sci. Comput 23 (2001) 1376–1395.

[20] R. Herzog, E.W. Sachs, Preconditioned conjugate gradient method for optimal control problems with control and state constraints, SIAM J. Matrix Anal. Appl. 31 (2010) 2291–2317.

[21] M. Hintermüller, K. Ito, K. Kunisch, The primal-dual active set strategy as a semismooth Newton method, SIAM J. Optim. 13 (2002) 865–888.

[22] M. Hinze, Optimal and instantaneous control of the instationary Navier-Stokes equations, Habilitation, TU Berlin, 2000.

[23] M. Hinze, M. Köster, S. Turek, A Hierarchical Space-Time Solver for Distributed Control of the Stokes Equation, Technical Report, SPP1253-16-01, 2008.

[24] M. Hinze, M. Köster, S. Turek, A Space-Time Multigrid Solver for Distributed Control of the Time-Dependent Navier-Stokes System, Technical Report, SPP1253-16-02, 2008.

[25] M. Hinze, R. Pinnau, M. Ulbrich, S. Ulbrich, Optimization with PDE Constraints, Mathematical Modelling: Theory and Applications, Springer-Verlag, New York, 2009.

[26] C. Keller, N. Gould, A. Wathen, Constraint Preconditioning for Indefinite Linear Systems, SIAM J. Matrix Anal. Appl 21 (2000) 1300–1317.

[27] Y. Maday, D. Meiron, A. Patera, E. Rønquist, Analysis of iterative methods for the steady and unsteady Stokes problem: Application to spectral element discretizations, SIAM Journal on Scientific Computing 14 (1993) 310.

[28] K. Mardal, R. Winther, Preconditioning discretizations of systems of partial differential equations, Numerical Linear Algebra with Applications 18 (2011) 1–40.

[29] K.A. Mardal, R. Winther, Construction of preconditioners by mapping properties, Construction of preconditioners by mapping properties, Bentham Science Publishers, 2010, pp. X–XX.

[30] M. Olshanskii, V. Simoncini, Acquired clustering properties and solution of certain saddle point systems., SIAM J. Matrix Anal. Appl. 31 (2010) 2754–2768.

[31] C.C. Paige, M.A. Saunders, Solutions of sparse indefinite systems of linear equations, SIAM J. Numer. Anal 12 (1975) 617–629.

[32] T. Rees, H.S. Dollar, A.J. Wathen, Optimal solvers for PDE-constrained optimization, SIAM Journal on Scientific Computing 32 (2010) 271–298.

[33] T. Rees, M. Stoll, Block-triangular preconditioners for pde-constrained optimization, Numerical Linear Algebra with Applications 17 (2010) 977–996.

[34] T. Rees, M. Stoll, A. Wathen, All-at-once preconditioners for PDE-constrained optimization, Kybernetika 46 (2010) 341–360.

[35] T. Rees, A. Wathen, Preconditioning iterative methods for the optimal control of the Stokes equation, Submitted (2010).

[36] T. Rusten, R. Winther, A preconditioned iterative method for saddlepoint problems, SIAM Journal on Matrix Analysis and Applications 13 (1992) 887.

[37] Y. Saad, A flexible inner-outer preconditioned GMRES algorithm, SIAM Journal on Scientific Computing 14 (1993) 461–461.

[38] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Statist. Comput 7 (1986) 856–869.

[39] J. Schöberl, W. Zulehner, Symmetric indefinite preconditioners for saddle point problems with applications to pde-constrained optimization problems, SIAM J. Matrix Anal. Appl 29 (2007) 752–773.

[40] D. Silvester, A. Wathen, Fast iterative solution of stabilised Stokes systems. II. Using general block preconditioners, SIAM J. Numer. Anal 31 (1994) 1352–1367.

[41] V. Simoncini, Reduced order solution of structured linear systems arising in certain PDE-constrained optimization problems, submitted (2011).

[42] M. Stoll, A. Wathen, Preconditioning for PDE-constrained optimization with control constraints, Submitted (2009).

[43] M. Stoll, A. Wathen, All-at-once solution of time-dependent PDE-constrained optimization problems, Submitted (2010).

[44] M. Stoll, A. Wathen, Preconditioners for state constrained optimal control problems with Moreau-Yosida penalty function, Submitted (2010).

[45] F. Tröltzsch, Optimale Steuerung partieller Differentialgleichungen: Theorie, Verfahren und Anwendungen, Vieweg Verlag, Wiesbaden, 2005.

[46] F. Tröltzsch, Optimal Control of Partial Differential Equations: Theory, Methods and Applications, Amer Mathematical Society, 2010.

[47] A. Wathen, D. Silvester, Fast iterative solution of stabilised Stokes systems. I. Using simple diagonal preconditioners, SIAM J. Numer. Anal 30 (1993) 630–649.

[48] A.J. Wathen, T. Rees, Chebyshev semi-iteration in preconditioning for problems including the mass matrix, Electronic Transactions in Numerical Analysis 34 (2008-2009) 125–135.

[49] W. Zulehner, Non-standard norms and robust estimates for saddle point problems, Submitted (2010).