

Efficient smoothers for all-at-once multigrid methods for Poisson and Stokes control problems

Stefan Takacs

stefan.takacs@numa.uni-linz.ac.at,

WWW home page: <http://www.numa.uni-linz.ac.at/~stefant/J3362/>

Mathematical Institute, University of Oxford, United Kingdom

The research was funded by the Austrian Science Fund (FWF): J3362-N25.

Abstract. In the present paper we concentrate on an important issue in constructing a good multigrid solver: the choice of an efficient smoother. We will introduce all-at-once multigrid solvers for optimal control problems which show robust convergence in the grid size and in the regularization parameter. We will refer to recent publications that guarantee such a convergence behavior. These publications do not pay much attention to the construction of the smoother and suggest to use a normal equation smoother. We will see that using a Gauss Seidel like variant of this smoother, the overall multigrid solver can be speeded up by a factor of about two with no additional work. The author will give a proof which indicates that also the Gauss Seidel like variant of the smoother is covered by the convergence theory. Numerical experiments suggest that the proposed methods are competitive with Vanka type methods.

Keywords: PDE-constrained optimization, All-at-once multigrid, Gauss Seidel

1 Introduction

In the present paper we discuss the construction of the all-at-once multigrid solvers for two model problems. The first model problem is a standard *Poisson control problem*: Find a state $y \in H^1(\Omega)$ and a control $u \in L^2(\Omega)$ such that they minimize the cost functional

$$J(y, u) := \frac{1}{2} \|y - y_D\|_{L^2(\Omega)}^2 + \frac{\alpha}{2} \|u\|_{L^2(\Omega)}^2,$$

subject to the elliptic boundary value problem (BVP)

$$-\Delta y + y = u \text{ in } \Omega \quad \text{and} \quad \frac{\partial y}{\partial n} = 0 \text{ on } \partial\Omega.$$

The desired state y_D and the regularization parameter $\alpha > 0$ are assumed to be given. Here and in what follows, $\Omega \subseteq \mathbb{R}^2$ is a polygonal domain. We want to solve the finite element discretization of this problem using a fast linear solver which shows robust convergence behavior in the grid size and the regularization

parameter. For solving this problem, we use the method of Lagrange multipliers, cf. [5, 6]. We obtain a linear system in the state y , the control u and the Lagrange multiplier λ . In this linear system we eliminate the control as this has been done in [6, 9]. We discretize the resulting system using the Courant element and obtain a linear system:

$$\underbrace{\begin{pmatrix} M_k & K_k \\ K_k & -\alpha^{-1}M_k \end{pmatrix}}_{\mathcal{A}_k :=} \underbrace{\begin{pmatrix} \underline{y}_k \\ \underline{\lambda}_k \end{pmatrix}}_{\underline{x}_k :=} = \underbrace{\begin{pmatrix} \underline{f}_k \\ 0 \end{pmatrix}}_{\underline{f}_k :=}. \quad (1)$$

Here, M_k and K_k are the standard mass and stiffness matrices, respectively. The control can be recovered using the following simple relation from the Lagrange multiplier: $\underline{u}_k = \alpha^{-1}\underline{\lambda}_k$, cf. [6]. In [6, 12] it was shown that there are constants $\underline{C} > 0$ and $\overline{C} > 0$ (independent of the grid size h_k and the choice of α) such that the stability estimate

$$\|\mathcal{Q}_k^{-1/2} \mathcal{A}_k \mathcal{Q}_k^{-1/2}\| \leq \overline{C} \quad \text{and} \quad \|\mathcal{Q}_k^{1/2} \mathcal{A}_k^{-1} \mathcal{Q}_k^{1/2}\| \leq \underline{C}^{-1} \quad (2)$$

holds for the symmetric and positive definite matrix

$$\mathcal{Q}_k := \begin{pmatrix} M_k + \alpha^{1/2}K_k & \\ & \alpha^{-1}M_k + \alpha^{-1/2}K_k \end{pmatrix}.$$

The second model problem is a standard *Stokes control problem* (velocity tracking problem): Find a velocity field $v \in [H^1(\Omega)]^d$, a pressure distribution $p \in L^2(\Omega)$ and a control $u \in [L^2(\Omega)]^d$ such that

$$J(v, p, u) = \frac{1}{2}\|v - v_D\|_{L^2(\Omega)}^2 + \frac{\alpha}{2}\|u\|_{L^2(\Omega)}^2$$

is minimized subject to the Stokes equations

$$-\Delta v + \nabla p = u \text{ in } \Omega, \quad \nabla \cdot v = 0 \text{ in } \Omega, \quad v = 0 \text{ on } \partial\Omega.$$

The regularization parameter $\alpha > 0$ and the desired state (desired velocity field) $v_D \in [L^2(\Omega)]^d$ are assumed to be given. To enforce uniqueness of the solution, we additionally require $\int_{\Omega} p \, dx = 0$.

Similar as above, we can set up the optimality system and eliminate the control, cf. [12, 7]. The discretization can be done using the Taylor-Hood element. After these steps, we end up with the following linear system:

$$\underbrace{\begin{pmatrix} M_k & K_k & D_k^T \\ 0 & D_k & \\ K_k & D_k^T & -\alpha^{-1}M_k \\ D_k & & 0 \end{pmatrix}}_{\mathcal{A}_k :=} \underbrace{\begin{pmatrix} \underline{v}_k \\ \underline{p}_k \\ \underline{\lambda}_k \\ \underline{\mu}_k \end{pmatrix}}_{\underline{x}_k :=} = \underbrace{\begin{pmatrix} \underline{f}_k \\ 0 \\ 0 \\ 0 \end{pmatrix}}_{\underline{f}_k :=}. \quad (3)$$

where M_k and K_k are standard mass and stiffness matrices and D_k^T is the discretization of the gradient operator, see, e.g., [12, 7]. Again, we are interested in

a fast solver which is robust in the regularization parameter and the grid size. As in the previous example, the control \underline{u}_k can be recovered from the Lagrange multiplier: $\underline{u}_k = \alpha^{-1} \underline{\lambda}_k$. In [12] it was shown that stability estimate (2) is satisfied for

$$\mathcal{Q}_k = \text{block-diag} \left(W_k, \alpha D_k W_k^{-1} D_k^T, \alpha^{-1} W_k, D_k W_k^{-1} D_k^T \right),$$

where $W_k := M_k + \alpha^{1/2} K_k$.

2 An all-at-once multigrid method

The linear systems (1) and (3) shall be solved by a multigrid method, which reads as follows. Starting from an initial approximation $\underline{x}_k^{(0)}$, one iterate of the multigrid method is given by the following two steps:

– *Smoothing procedure:* Compute

$$\underline{x}_k^{(0,m)} := \underline{x}_k^{(0,m-1)} + \hat{\mathcal{A}}_k^{-1} \left(\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,m-1)} \right) \quad \text{for } m = 1, \dots, \nu$$

with $\underline{x}_k^{(0,0)} = \underline{x}_k^{(0)}$. The choice of the smoother (or, in other words, of the matrix $\hat{\mathcal{A}}_k^{-1}$) will be discussed below.

– *Coarse-grid correction:*

- Compute the defect $\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,\nu)}$ and restrict it to grid level $k-1$ using an restriction matrix I_k^{k-1} : $\underline{r}_{k-1}^{(1)} := I_k^{k-1} \left(\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,\nu)} \right)$.
- Solve the following coarse-grid problem approximatively:

$$\mathcal{A}_{k-1} \underline{p}_{k-1}^{(1)} = \underline{r}_{k-1}^{(1)} \tag{4}$$

- Prolongate $\underline{p}_{k-1}^{(1)}$ to the grid level k using an prolongation matrix I_{k-1}^k and add the result to the previous iterate: $\underline{x}_k^{(1)} := \underline{x}_k^{(0,\nu)} + I_{k-1}^k \underline{p}_{k-1}^{(1)}$.

As we have assumed to have nested spaces, the intergrid-transfer matrices can be chosen in a canonical way: I_{k-1}^k is the canonical embedding and the restriction I_k^{k-1} is its (properly scaled) transpose. If the problem (4) is solved exactly, we obtain the two-grid method. In practice, the problem (4) is approximatively solved by applying one step (V-cycle) or two steps (W-cycle) of the multigrid method, recursively. Only the coarsest grid level, (4) is solved exactly.

The only part of the multigrid algorithm that has not been specified yet, is the smoother. For the choice of the smoother, we make use of the convergence theory. We develop a convergence theory based on Hackbusch's splitting of the analysis into smoothing property and approximation property:

– *Smoothing property:*

$$\sup_{\tilde{\underline{x}}_k \in X_k} \frac{\left(\mathcal{A}_k(\underline{x}_k^{(0,\nu)} - \underline{x}_k^*), \tilde{\underline{x}}_k \right)_{\ell^2}}{\|\tilde{\underline{x}}_k\|_{\mathcal{L}_k}} \leq \eta(\nu) \|\underline{x}_k^{(0)} - \underline{x}_k^*\|_{\mathcal{L}_k} \tag{5}$$

should hold for some function $\eta(\nu)$ with $\lim_{\nu \rightarrow \infty} \eta(\nu) = 0$. Here and in what follows, $\underline{x}_k^* := \mathcal{A}_k^{-1} \underline{f}_k$ is the exact solution, $\|\cdot\|_{\mathcal{L}_k} := (\cdot, \cdot)_{\mathcal{L}_k}^{1/2} := (\mathcal{L}_k \cdot, \cdot)_{\ell^2}^{1/2}$ for some symmetric positive definite matrix \mathcal{L}_k and $(\cdot, \cdot)_{\ell^2}$ is the standard Euclidean scalar product.

– *Approximation property:*

$$\|\underline{x}_k^{(1)} - \underline{x}_k^*\|_{\mathcal{L}_k} \leq C_A \sup_{\tilde{\underline{x}}_k \in \tilde{X}_k} \frac{\left(\mathcal{A}_k(\underline{x}_k^{(0,\nu)} - \underline{x}_k^*), \tilde{\underline{x}}_k \right)_{\ell^2}}{\|\tilde{\underline{x}}_k\|_{\mathcal{L}_k}}$$

should hold for some constant $C_A > 0$.

It is easy to see that, if we combine both conditions, we see that the two-grid method converges in the norm $\|\cdot\|_{\mathcal{L}_k}$ for ν large enough. The convergence of the W-cycle multigrid method can be shown under mild assumptions, see e.g. [3].

For the smoothing analysis, it is convenient to rewrite the smoothing property in pure matrix notation: (5) is equivalent to

$$\|\mathcal{L}_k^{-1/2} \mathcal{A}_k (I - \hat{\mathcal{A}}_k^{-1} \mathcal{A}_k)^\nu \mathcal{L}_k^{-1/2}\| \leq \eta(\nu). \quad (6)$$

For the Poisson control problem, it was shown in [6], that the approximation property is satisfied for the following choice of the matrix \mathcal{L}_k (note that this matrix represents the norm $\|\cdot\|_X$ used in the mentioned paper)

$$\mathcal{L}_k = \begin{pmatrix} \text{diag}(M_k + \alpha^{1/2} K_k) & \\ & \text{diag}(\alpha^{-1} M_k + \alpha^{-1/2} K_k) \end{pmatrix},$$

i.e., $\mathcal{L}_k = \text{diag}(\mathcal{Q}_k)$. Here and in what follows, $\text{diag}(M)$ is the diagonal of a matrix M . For the Stokes control problem it was shown in [7], that the approximation property is satisfied for the following choice of \mathcal{L}_k :

$$\mathcal{L}_k = \begin{pmatrix} \hat{W}_k & & & \\ & \hat{P}_k & & \\ & & \alpha^{-1} \hat{W}_k & \\ & & & \alpha^{-1} \hat{P}_k \end{pmatrix},$$

where $\hat{W}_k := \text{diag}(M_k + \alpha^{1/2} K_k)$ and $\hat{P}_k := \alpha \text{diag}(D_k \hat{W}_k^{-1} D_k^T)$.

Still, we have not specified the choice of the smoother, which now can be done using the convergence theory. We have seen for which choices of \mathcal{L}_k the approximation property is satisfied. We are interested in a smoother such that the smoothing property is satisfied for the same choice of \mathcal{L}_k .

In [9, 7] a *normal equation smoother* was proposed. This approach is applicable to a quite general class of problems, cf. [2] and others. In our notation, the normal equation smoother reads as follows:

$$\underline{x}_k^{(0,m)} := \underline{x}_k^{(0,m-1)} + \underbrace{\tau \mathcal{L}_k^{-1} \mathcal{A}_k^T \mathcal{L}_k^{-1}}_{\hat{\mathcal{A}}_k^{-1} :=} \left(\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,m-1)} \right) \quad \text{for } m = 1, \dots, \nu.$$

Here, a fixed $\tau > 0$ has to be chosen such that the spectral radius $\rho(\tau \hat{\mathcal{A}}_k^{-1} \mathcal{A}_k)$ is bounded away from 2 on all grid levels k and for all choices of the parameters. It was shown that it is possible to find such a uniform τ for the Poisson control problem, e.g., in [9] and for the Stokes control problem, e.g., in [7]. For the normal equation smoother, the smoothing property can be shown using a simple eigenvalue analysis, cf. [2].

Numerical experiments show that the normal equation smoother works rather well for the mentioned model problems. However, there are smoothers such that the overall multigrid method converges much faster. Note that the normal equation smoother is basically a Richardson iteration scheme, applied to the normal equation. It is well-known for elliptic problems that Gauss Seidel iteration schemes are typically much better smoothers than Richardson iteration schemes. In the context of saddle point problems, the idea of Gauss Seidel smoothers has been applied, e.g., in the context of collective smoothers, see below. However, in the context of normal equation smoothers the idea of Gauss Seidel smoothers has not gained much attention. The setup of such an approach is straight forward: In compact notation such an approach, which we call *least squares Gauss Seidel* (LSGS) approach, reads as follows:

$$\underline{x}_k^{(0,m)} := \underline{x}_k^{(0,m-1)} + \underbrace{\text{trig}(\mathcal{N}_k)^{-1} \mathcal{A}_k^T \mathcal{L}_k^{-1}}_{\hat{\mathcal{A}}_k :=} \left(\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,m-1)} \right) \quad \text{for } m = 1, \dots, \nu,$$

where $\mathcal{N}_k := \mathcal{A}_k^T \mathcal{L}_k^{-1} \mathcal{A}_k$ and $\text{trig}(M)$ is the left-lower triangular part (including the diagonal) of a matrix M . The author provides a possible realization of that approach as Algorithm 2 to convince the reader that the computational complexity of the LSGS approach is equal to the computational complexity of the normal equation smoother, where a possible realization is given as Algorithm 1.

We will see below that the LSGS approach works very well in the numerical experiments. However, there is no proof of the smoothing property known to the author. This is due to the fact that the matrix $\hat{\mathcal{A}}_k$ is not symmetric. One possibility to overcome this difficulty is to consider the symmetric version (symmetric least squares Gauss Seidel approach, sLSGS approach). This is analogous to the case of elliptic problems: For elliptic problems the smoothing property for the symmetric Gauss Seidel iteration can be shown for general cases but for the standard Gauss Seidel iteration the analysis is restricted to special cases, cf. Section 6.2.4 in [3].

One step of the sLSGS iteration consists of one step of the LSGS iteration, followed by one step of the LSGS iteration with reversed order of the variables. (So the computational complexity of of step of the sLSGS iteration is equal to the computational complexity of two steps of the standard LSGS iteration.) One step of the sLSGS iteration reads as follows in compact notation:

$$\underline{x}_k^{(0,m)} := \underline{x}_k^{(0,m-1)} + \hat{\mathcal{N}}_k^{-1} \mathcal{A}_k^T \mathcal{L}_k^{-1} \left(\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,m-1)} \right) \quad \text{for } m = 1, \dots, \nu,$$

where $\hat{\mathcal{N}}_k := \text{trig}(\mathcal{N}_k) \text{diag}(\mathcal{N}_k)^{-1} \text{trig}(\mathcal{N}_k)^T$. (7)

For our needs, the following convergence lemma is sufficient.

Given: Iterate $(\mathbf{x}_i)_{i=1}^N = \underline{\mathbf{x}}^{(0,m-1)}$ and corresp. residual $(\mathbf{r}_i)_{i=1}^N = \underline{\mathbf{f}} - \mathcal{A}\underline{\mathbf{x}}^{(0,m-1)}$;
Result: Iterate $(\mathbf{x}_i)_{i=1}^N = \underline{\mathbf{x}}^{(0,m)}$ and corresp. residual $(\mathbf{r}_i)_{i=1}^N = \underline{\mathbf{f}} - \mathcal{A}\underline{\mathbf{x}}^{(0,m)}$;
for $i = 1, \dots, N$ **do**
 $\mathbf{q} := 0$;
 for all j **such that** $\mathcal{A}_{i,j} \neq 0$ **do** $\mathbf{q} := \mathbf{q} + \mathcal{A}_{i,j}/\mathcal{L}_{j,j} * \mathbf{r}_j$;
 $\mathbf{p}_i := \tau * \mathbf{q}/\mathcal{L}_{i,i}$;
end
for $i = 1, \dots, N$ **do**
 $\mathbf{x}_i := \mathbf{x}_i + \mathbf{p}_i$;
 for all j **such that** $\mathcal{A}_{j,i} \neq 0$ **do** $\mathbf{r}_j := \mathbf{r}_j - \mathcal{A}_{j,i} * \mathbf{p}_i$;
end

Algorithm 1: Normal equation iteration scheme

Given: Iterate $(\mathbf{x}_i)_{i=1}^N = \underline{\mathbf{x}}^{(0,m-1)}$ and corresp. residual $(\mathbf{r}_i)_{i=1}^N = \underline{\mathbf{f}} - \mathcal{A}\underline{\mathbf{x}}^{(0,m-1)}$;
Result: Iterate $(\mathbf{x}_i)_{i=1}^N = \underline{\mathbf{x}}^{(0,m)}$ and corresp. residual $(\mathbf{r}_i)_{i=1}^N = \underline{\mathbf{f}} - \mathcal{A}\underline{\mathbf{x}}^{(0,m)}$;
Prepare once: $\mathcal{N}_{i,i} := \sum_{j=1}^N \mathcal{A}_{i,j}^2/\mathcal{L}_{j,j}$ for all $i = 1, \dots, N$;
for $i = 1, \dots, N$ **do**
 $\mathbf{q} := 0$;
 for all j **such that** $\mathcal{A}_{i,j} \neq 0$ **do** $\mathbf{q} := \mathbf{q} + \mathcal{A}_{i,j}/\mathcal{L}_{j,j} * \mathbf{r}_j$;
 $\mathbf{p} := \mathbf{q}/\mathcal{N}_{i,i}$;
 $\mathbf{x}_i := \mathbf{x}_i + \mathbf{p}$;
 for all j **such that** $\mathcal{A}_{j,i} \neq 0$ **do** $\mathbf{r}_j := \mathbf{r}_j - \mathcal{A}_{j,i} * \mathbf{p}$;
end

Algorithm 2: LSGS iteration scheme

Lemma 1. Assume that \mathcal{A}_k is sparse, (2) is satisfied and let \mathcal{L}_k be a positive definite diagonal matrix such that

$$\|\mathcal{Q}_k^{1/2} \underline{\mathbf{x}}_k\| \leq \|\mathcal{L}_k^{1/2} \underline{\mathbf{x}}_k\| \quad \text{for all } \underline{\mathbf{x}}_k. \quad (8)$$

Then the sLSGS approach satisfies the smoothing property (6), i.e.,

$$\|\mathcal{L}_k^{-1/2} \mathcal{A}_k (I - \hat{\mathcal{N}}_k^{-1} \mathcal{N}_k)^\nu \mathcal{L}_k^{-1/2}\| \leq \frac{2^{-1/2} \bar{C} \text{nnz}(\mathcal{A}_k)^{5/2}}{\sqrt{\nu}},$$

where $\text{nnz}(M)$ is the maximum number of non-zero entries per row of M .

Note that (8) is a standard inverse inequality, which is satisfied for both model problems, cf. [6, 9, 7]. Note moreover that this assumption also has to be satisfied to show the smoothing property for the normal equation smoother, cf. [9, 7].

Proof of Lemma 1. The combination of (2) and (8) yields $\|\mathcal{L}_k^{-1/2} \mathcal{A}_k \mathcal{L}_k^{-1/2}\| \leq \bar{C}$. Prop. 6.2.27 in [3] states that for any symmetric positive definite matrix \mathcal{N}_k

$$\|\hat{\mathcal{N}}_k^{-1/2} \mathcal{N}_k (I - \hat{\mathcal{N}}_k^{-1} \mathcal{N}_k)^\nu \hat{\mathcal{N}}_k^{-1/2}\| \leq \nu^{-1} \quad (9)$$

holds, where $\hat{\mathcal{N}}_k$ is as in (7). Using $\mathcal{D}_k := \text{diag}(\mathcal{N}_k)$, we obtain

$$\begin{aligned} \|\mathcal{L}_k^{-1/2} \hat{\mathcal{N}}_k^{1/2}\|^2 &= \rho(\mathcal{L}_k^{-1/2} \hat{\mathcal{N}}_k \mathcal{L}_k^{-1/2}) \leq \|\mathcal{L}_k^{-1/2} \text{trig}(\mathcal{N}_k) \mathcal{D}_k^{-1/2}\|^2 \\ &\leq \|\mathcal{L}_k^{-1/2} \mathcal{D}_k^{1/2}\|^2 \|\mathcal{D}_k^{-1/2} \text{trig}(\mathcal{N}_k) \mathcal{L}_k^{-1/2}\|^2 \end{aligned}$$

Let $\mathcal{A}_k = (\mathcal{A}_{i,j})_{i,j=1}^N$, $\mathcal{N}_k = (\mathcal{N}_{i,j})_{i,j=1}^N$, $\mathcal{L}_k = (\mathcal{L}_{i,j})_{i,j=1}^N$ and $\psi(i) := \{j \in \mathbb{N} : \mathcal{N}_{i,j} \neq 0\}$. We obtain using Gerschgorin's theorem, the fact that the infinity norm is monotone in the matrix entries, and using the symmetry of \mathcal{N}_k and \mathcal{A}_k and Cauchy-Schwarz inequality:

$$\begin{aligned}
& \|\mathcal{D}_k^{-1/2} \text{trig}(\mathcal{N}_k) \mathcal{D}_k^{-1/2}\| \\
& \leq \|\mathcal{D}_k^{-1/2} \text{trig}(\mathcal{N}_k) \mathcal{D}_k^{-1/2}\|_\infty^{1/2} \|\mathcal{D}_k^{-1/2} \text{trig}(\mathcal{N}_k)^T \mathcal{D}_k^{-1/2}\|_\infty^{1/2} \leq \|\mathcal{D}_k^{-1/2} \mathcal{N}_k \mathcal{D}_k^{-1/2}\|_\infty \\
& = \max_{i=1,\dots,N} \sum_{k \in \psi(i)} \left(\sum_{n=1}^N \frac{\mathcal{A}_{i,n}^2}{\mathcal{L}_{n,n}} \right)^{-1/2} \left(\sum_{j=1}^N \frac{\mathcal{A}_{i,j} \mathcal{A}_{j,k}}{\mathcal{L}_{j,j}} \right) \left(\sum_{n=1}^N \frac{\mathcal{A}_{k,n}^2}{\mathcal{L}_{n,n}} \right)^{-1/2} \\
& \leq \max_{i=1,\dots,N} \sum_{k \in \psi(i)} 1 = \text{nnz}(\mathcal{N}_k) \leq \text{nnz}(\mathcal{A}_k)^2. \tag{10}
\end{aligned}$$

Further, we obtain

$$\begin{aligned}
\|\mathcal{L}_k^{-1/2} \mathcal{D}_k^{1/2}\|^2 &= \|\mathcal{L}_k^{-1/2} \mathcal{D}_k^{1/2}\|_\infty^2 = \|\mathcal{L}_k^{-1/2} \mathcal{D}_k \mathcal{L}_k^{-1/2}\|_\infty = \max_{i=1,\dots,N} \sum_{j=1}^N \frac{\mathcal{A}_{i,j}^2}{\mathcal{L}_{i,i} \mathcal{L}_{j,j}} \\
&\leq \text{nnz}(\mathcal{A}_k) \max_{i,j=1,\dots,N} \frac{\mathcal{A}_{i,j}^2}{\mathcal{L}_{i,i} \mathcal{L}_{j,j}} = \text{nnz}(\mathcal{A}_k) \|\mathcal{L}^{-1/2} \mathcal{A} \mathcal{L}^{-1/2}\|^2 \leq \text{nnz}(\mathcal{A}_k) \bar{C}^2. \tag{11}
\end{aligned}$$

By combining (9), (10) and (11), we obtain

$$\begin{aligned}
& \|\mathcal{L}_k^{-1/2} \mathcal{A}_k (I - \hat{\mathcal{N}}_k^{-1} \mathcal{N}_k)^\nu \mathcal{L}_k^{-1/2}\|^2 \\
& \leq \|\mathcal{L}_k^{-1/2} (I - \mathcal{N}_k \hat{\mathcal{N}}_k^{-1})^\nu \mathcal{A}_k \mathcal{L}_k^{-1} \mathcal{A}_k (I - \hat{\mathcal{N}}_k^{-1} \mathcal{N}_k)^\nu \mathcal{L}_k^{-1/2}\| \\
& = \|\mathcal{L}_k^{-1/2} \mathcal{N}_k (I - \hat{\mathcal{N}}_k^{-1} \mathcal{N}_k)^{2\nu} \mathcal{L}_k^{-1/2}\| \leq \frac{\bar{C}^2 \text{nnz}(\mathcal{A}_k)^5}{2\nu},
\end{aligned}$$

which finishes the proof. \square

We went to compare the numerical behavior of the LSGS approach with the behavior of a standard smoother. One class of standard smoothers for saddle point problems is the class of *Vanka type smoothers*, which has been originally introduced for Stokes problems, cf. [11]. Such smoothers have also gained interest for optimal control problems, see, e.g., [10, 1, 8].

The idea of Vanka type smoothers is to compute updates in subspaces directly for the whole saddle point problem and to combine these updates in an additive or a multiplicative way to compute the next update. Here, the variables are not grouped based on the block-structure of \mathcal{A}_k , but the grouping is done based on the location of the corresponding degrees of freedom in the domain Ω . The easiest of such ideas for the Poisson control problems is to do the grouping point-wise, which leads to the idea of *point smoothing*. Here, we group for each node δ_i of the discretization (each degree of freedom of the Courant element) the value y_i of the state and the value λ_i of the Lagrange multiplier and compute

an update in the corresponding subspace. The multiplicative variant of such a smoother is a *collective Gauss Seidel* (CGS) smoother:

$$\underline{x}_k^{(0,m,i)} := \underline{x}_k^{(0,m,i-1)} + \mathcal{P}_k^{(i)} \left(\mathcal{P}_k^{(i)T} \mathcal{A}_k \mathcal{P}_k^{(i)} \right)^{-1} \mathcal{P}_k^{(i)T} \left(\underline{f}_k - \mathcal{A}_k \underline{x}_k^{(0,m,i-1)} \right),$$

where $\underline{x}_k^{(0,m,0)} := \underline{x}_k^{(0,m-1)}$ and $\underline{x}_k^{(0,m)} := \underline{x}_k^{(0,m,N_k)}$. For each $i = 1, \dots, N_k$, the matrix $\mathcal{P}_k^{(i)} \in \mathbb{R}^{2N_k \times 2}$ takes the value 1 on the positions $(i, 1)$ and $(i + N_k, 2)$ and the value 0 elsewhere. For the Poisson control problem, we obtain

$$\mathcal{P}_k^{(i)T} \mathcal{A}_k \mathcal{P}_k^{(i)} = \begin{pmatrix} M_{i,i} & K_{i,i} \\ K_{i,i} & -\alpha^{-1} M_{i,i} \end{pmatrix},$$

where $M_{i,i}$ and $K_{i,i}$ are the entries of the matrices M_k and K_k .

For the Stokes control problem, it is not reasonable to use exactly the same approach. This is basically due to the fact that the degrees of freedom for v and λ are not located on the same positions as the degrees of freedom for p and μ . However, we can introduce an approach based on patches: so, for each vertex of the triangulation, we consider subspaces that consist of the degrees of freedoms located on the vertex itself and the degrees of freedom located on all edges which have one end at the chosen vertex, cf. Fig. 1. Note that here the subspaces are

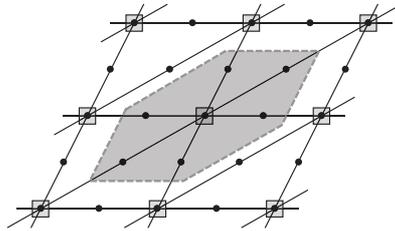


Fig. 1. Patches for the Vanka-type smoother applied to a Taylor Hood discretization. The dots are the degrees of freedom of v and λ , the rectangles are the degrees of freedom of p and μ

much larger than the subspaces chosen in the case of the CGS approach for the Poisson control problem (which was just 2). This increases the computational cost of applying the method significantly. For Vanka type smoothers there are only a few convergence results known, cf. [1] for a Fourier Analysis and an analysis based compactness argument and [8] for a proof based on Hackbusch's splitting of the analysis into smoothing property and approximation property which shows the convergence in case of a collective Richardson smoother.

3 Numerical results

In this section we give numerical results to illustrate quantitatively the convergence behavior of the proposed methods. The number of iterations was measured as follows: We start with a random initial guess and iterate until the relative error

in the norm $\|\cdot\|_{\mathcal{L}_k}$ was reduced by a factor of 10^{-6} . Without loss of generality, the right-hand side was chosen to be 0. For both model problems, the normal equation smoother, the LSGS smoother, the sLSGS smoother and a Vanka type smoother have been applied. For the smoothers 2 pre- and 2 post-smoothing steps have been applied. Only for the sLSGS smoother, just 1 pre- and 1 post-smoothing step has been applied. This is due to the fact that one step of the symmetric version is basically the same computational cost as two steps of the standard version. The normal equation smoother was damped with $\tau = 0.4$ for the Poisson control problem and $\tau = 0.35$ for the Stokes control problem, cf. [9, 7]. For the Gauss Seidel-like approaches, damping was not used.

In Table 1, we give the results for the standard Poisson control problem. Here, we see that all smoothers lead to convergence rates that are well bounded for a wide range of h_k and α . Compared to the normal equation smoother, the LSGS smoother leads to a speedup be a factor of about two without any additional work. The symmetric version (sLSGS) is a bit slower than the LSGS method. For the first model problem, the (popular) CGS method is significantly faster. However, for this method no convergence theory is known.

	Normal equation			LSGS			sLSGS			CGS		
$\alpha =$	10^0	10^{-6}	10^{-12}	10^0	10^{-6}	10^{-12}	10^0	10^{-6}	10^{-12}	10^0	10^{-6}	10^{-12}
$k = 5$	26	31	28	11	9	7	14	12	14	5	5	3
$k = 6$	27	28	29	11	11	7	14	14	13	5	5	3
$k = 7$	27	28	31	11	11	6	14	14	12	5	5	3
$k = 8$	27	27	25	11	11	3	14	14	7	5	5	4

Table 1. Number of iterations for the *Poisson control model problem*

In Table 2, we give the convergence results for the Stokes control problem. Also here we observe that the LSGS and the sLSGS approach lead to a speedup of a factor of about two compared to the normal equation smoother. Here, the Vanka type smoother shows slightly smaller iteration numbers than the LSGS approach. In terms of computational costs, the LSGS smoother seems to be much better than the patch-based Vanka type smoother because there relatively large subproblems have to be solved to compute the updates. This is different the case of the CGS smoother, where the subproblems are just 2-by-2 linear systems. Numerical experiments have shown that the undamped version of the patch-based Vanka type method does not lead to a convergent multigrid method. So, this smoother was damped with $\tau = 0.4$. Due to lack of convergence theory, the author cannot explain why this approach – although it is a multiplicative approach – needs damping.

For completeness, the author wants to mention that for cases, where a (closed form of a) matrix \mathcal{Q}_k satisfying (2) robustly is not known, the normal equation smoother does not show as good results as methods where such an information is not needed, like Vanka type methods. This was discussed in [8] for a bound-

	Normal equation			LSGS			sLSGS			Vanka type		
$\alpha =$	10^0	10^{-6}	10^{-12}	10^0	10^{-6}	10^{-12}	10^0	10^{-6}	10^{-12}	10^0	10^{-6}	10^{-12}
$k = 4$	31	31	60	13	12	14	17	16	22	11	10	7
$k = 5$	32	30	55	14	13	12	18	16	19	11	10	7
$k = 6$	32	31	44	14	13	9	18	17	12	11	11	7
$k = 7$	32	31	37	14	14	6	18	17	9	11	11	9

Table 2. Number of iterations for the *Stokes control model problem*

ary control problem, but it is also true for the linearization of optimal control problems with inequality constraints as discussed in [4] and others. The same is true for the Gauss Seidel like variants of the normal equation smoother.

Concluding, we have observed that accelerating the idea of normal equation smoothing with a Gauss Seidel approach, leads to a speedup of a factor of about two without any further work. The fact that convergence theory is known for the sLSGS approach, helps also for the numerical practise (unlike the case of Vanka type smoothers).

References

1. A. Borzi, K. Kunisch, and D.Y. Kwak, *Accuracy and convergence properties of the finite difference multigrid solution of an optimal control optimality system*, SIAM J. on Control and Optimization **41(5)** (2003), 1477 – 1497.
2. S.C. Brenner, *Multigrid methods for parameter dependent problems*, RAIRO, Modélisation Math. Anal. Numér **30** (1996), 265 – 297.
3. W. Hackbusch, *Multi-Grid Methods and Applications*, Springer, Berlin, 1985.
4. R. Herzog and E. Sachs, *Preconditioned Conjugate Gradient Method for Optimal Control Problems with Control and State Constraints*, SIAM J. on Matrix Anal. & Appl **31** (2010), no. 5, 2291 – 2317.
5. J.L. Lions, *Optimal control of systems governed by partial differential equations*, Berlin-Heidelberg-New York: Springer-Verlag, 1971.
6. J. Schöberl, R. Simon, and W. Zulehner, *A Robust Multigrid Method for Elliptic Optimal Control Problems*, SIAM J. on Numerical Analysis **49** (2011), 1482 – 1503.
7. S. Takacs, *A robust all-at-once multigrid method for the Stokes control problem*, 2013, submitted.
8. S. Takacs and W. Zulehner, *Convergence Analysis of Multigrid Methods with Collective Point Smoothers for Optimal Control Problems*, Computing and Visualization in Science **14** (2011), no. 3, 131–141.
9. ———, *Convergence analysis of all-at-once multigrid methods for elliptic control problems under partial elliptic regularity*, SIAM Journal on Numerical Analysis **51** (2013), no. 3, 1853–1874.
10. U. Trottenberg, C. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, London, 2001.
11. S. P. Vanka, *Block-implicit multigrid solution of Navier-Stokes equations in primitive variables*, Math. Comp. **65** (1986), 138 – 158.
12. W. Zulehner, *Non-standard Norms and Robust Estimates for Saddle Point Problems*, SIAM J. on Matrix Anal. & Appl **32** (2011), 536 – 560.