Computing and Visualization in Science manuscript No.
(will be inserted by the editor)

Smoothed aggregation multigrid for a Stokes problem

Aleš Janka *

Institute of Analysis and Scientific Computing, Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne

Received: August 11, 2006

Abstract. We discuss advantages of using algebraic multigrid based on smoothed aggregation for solving indefinite linear problems. The ingredients of smoothed aggregation are used to construct a black-box monolithic multigrid method with idefinite coarse problems. Several techniques enforcing inf-sup stability conditions on coarse levels are presented. Numerical experiments are designed to support recent stability results for coupled algebraic multigrid. Comparison of the proposed multigrid preconditioner with other methods shows its robust behaviour even for very elongated geometries, where the pressure mass matrix is no longer a good preconditioner for the pressure Schur complement.

Key words Stokes, multigrid, smoothed aggregation, preconditioner

1 Introduction

This paper investigates the application of the algebraic multigtid by smoothed aggregation [17] for solving indefinite problems in a black-box way. Although we discuss its use for GLS stabilized P1-P1 finite element discretizations of steady-state or evolutive Stokes problems, all the concepts proposed here should transpose also to other schemes.

While multigrid methods for second order positive definite problems are sufficiently mature to be used for industrial problems, multigrid for saddle-point problems is still under development. A straightforward way of applying multigrid to indefinite problems is to use standard segregation schemes like Uzawa [4], block LU for velocity-presure blocks [7,13], or pressure Schur complement method in order to split the saddle-point problem into a set of positive definite subproblems, for which efficient multigrid schemes exist. This approach is described e.g. in [10] or [16]. In such a case, the limiting factor is the convergence speed of the external iteration of the segregation scheme.

In most cases, a good preconditioner [11] for the pressure Schur complement is the crucial point. Althought there exist results on the h-optimal preconditioning of the Schur complement by other sparse operators [13], these approaches are (i) not black-box, since they need a problem-dependent discrete preconditioning operator to be assembled in addition to the original saddle-point problem, and (ii) they usually do not consider the dependence of the preconditioning on the geometry of the computational domain Ω.

In particular, we state in Section 4 that the common pressure mass matrix preconditioner for the steady Stokes problem is, true, h-independent, but its efficiency depends on the shape of Ω and can perform very poorly for elongated domains.

Instead, we opt for a monolithic coupled multigrid method like in [20,1,19], i.e. solving saddle point problems in each multigrid level, while using segregated methods as pre- and post-smoothers. The aim is to build a robust black-box monolithic preconditioner to a Krylov-space method capable of dealing with steady and evolutive saddle-point problems in a unified way.

In Section 3 we give a brief overview of selected segregated and monolithic methods, which are then used either for comparison in numerical tests, or as ingredients of the proposed generalization of the smoothed aggregation multigrid for saddle-point problems.

Section 4 states some stability results of [12] and [19], put in a form which can be experimentally tested.

Section 5 contains numerical results supporting our scheme, as compared to preconditioned Krylov methods for the pressure Schur complement, or monolithic ones using GMRES preconditioned by block LU preconditioners [14].

2 Preliminaries

Let $\Omega \subset \mathbb{R}^d$, $d = \{2, 3\}$ be a bounded domain. We will write the stabilized Stokes problem in its weak form: find

* Supported by the Swiss CTI grant no. 6437.1 IWS-IW, in collaboration with Alcan-Péchiney.

$u \in X \equiv [H_0^1(\Omega)]^d$ and $p \in Y \equiv \{q \in L^2(\Omega) : \int_\Omega q = 0\}$ such that

$$\begin{aligned} a(u,v) + b(v,p) &= (f,v) \qquad && \text{for all } v \in X \\ b(u,q) - c(p,q) &= (g,q) && \text{for all } q \in Y. \end{aligned} \tag{1}$$

Here, we assume that Ω is polygonal. Its triangulation τ_h consists of simplex elements of mesh-size h, with angles bounded from below. Furthermore, $f \in X'$ and $g \in Y'$ are given, and

$$\begin{aligned} a(u,v) &\equiv \int_\Omega \frac{\rho}{\tau} uv + \mu \int_\Omega \nabla u \nabla v, \\ b(v,q) &\equiv -\int_\Omega \operatorname{div} vq, \\ c(p,q) &\equiv \sum_{K\in\tau_h} \frac{\alpha h_K^2}{\mu} \int_K \nabla p \nabla q. \end{aligned} \tag{2}$$

Let $(u_h, p_h) \in X_h \times Y_h$ be the P1 finite element solution of (1), where the spaces X and Y are formally replaced by their P1 finite element approximations X_h and Y_h. For the sake of simplicity, our notation will not distinguish between the finite element spaces X_h, resp. Y_h and the underlying vector spaces of degrees of freedom, i.e. we write $u_h \in X_h$, as well as $\mathbf{u} \in X_h$.

We assume that the finite element formulation (1) is stabilized, i.e. the obtained discrete system is inversible. We search for vectors $(\mathbf{u}, \mathbf{p}) \in (X_h, Y_h)$ such that

$$\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}, \qquad K \equiv \begin{pmatrix} A & B^T \\ B & -C \end{pmatrix}. \tag{3}$$

The matrices A, B, B^T, and C are discretizations of the bilinear forms in (2).

3 Solution methods

Over the last decades, very efficient multigrid methods (algebraic, by aggregation, or geometric) have been proposed and quite successfully analysed for symmetric positive definite problems. Their success invites to apply them also to saddle-point problems. There are two basic options: (i) a straightforward use of known multigrid schemes for SPD systems, applied to a sequence of SPD subproblems obtained by decoupling pressures and velocities in the saddle-point problem [10,16], or (ii) generalizing the multigrid approach to treat saddle-point problems on each multigrid level by extending the notions of smoothing and coarse-grid correction [1,19].

3.1 Multigrid for SPD subproblems of the saddle-point problem

In the approach (i), one generates a sequence of SPD subproblems, by often decoupling (segregating) the pressure degrees of freedom from the velocity ones, like in the Uzawa scheme, the pressure correction, or pressure Schur complement method. While the SPD subproblems might be efficiently solved by existing multigrid methods, the convergence of the overall decoupling process heavily depends on its preconditioning. This is a huge drawback, since the discrete operator to precondition is full, in the ideal case. Let us give the following examples of this class of solvers.

Pressure Schur complement: by eliminating the velocity from (3) we obtain a Schur complement system for pressure

$$S\mathbf{p} = BA^{-1}\mathbf{f} - \mathbf{g}, \tag{4}$$

where $S \equiv BA^{-1}B^T + C$. The system matrix S is SPD, but it is in general full. Thus, we need solution methods for (4) which do not need to assemble S explicitly. One of them is the conjugate gradient (CG) method, which needs only to evaluate iterative residuals $\mathbf{r}_i = \mathbf{r}_0 - S\delta\mathbf{p}_i$. For a given update of pressure $\delta\mathbf{p}_i$ in the i-th iteration, one performs

$$\mathbf{r}_i = \mathbf{r}_0 - B[A^{-1}(B^T\delta\mathbf{p}_i)] - C\delta\mathbf{p}_i,$$

involving a solution of a sparse SPD auxiliary system

$$A\boldsymbol{\eta} = B^T\delta\mathbf{p}_i.$$

For this task, efficient multigrid methods exist. The convergence of the CG method might be enhanced by preconditioning. In this case preconditioners cannot be constructed from the matrix S in an algebraic way, as S is a never assembled full matrix. Instead, one might use an external analytic result, like the one in Lemma 4.1. Even though this strategy might work well for some cases, the lack of algebraic preconditioner to S is an obstacle in the construction of an efficient black-box method. Moreover, we show that it does not lead to robust preconditioning with respect to large elongations $\mathcal{L}$ of the domain Ω, or small timesteps τ.

Block LU preconditioner of a monolithic scheme: Silvester, Elman, Wathen et al. [7,13,14,6] propose a preconditioner for a monolithic Krylov-space method based on the following block LU factorization ($K = L \cdot U$),

$$\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} = \begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix} \begin{pmatrix} A & B^T \\ 0 & -BA^{-1}B^T - C \end{pmatrix}.$$

Thus, using an approximation of U^{-1} as a right preconditioner gives

$$\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} \hat{A} & B^T \\ 0 & -\hat{S} \end{pmatrix}^{-1} \approx \begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix}, \tag{5}$$

where $\hat{A}$ and $\hat{S}$ are suitable preconditioners for A and S. If exact U is used, the Krylov method converges in 2 steps, due to the lower-triangular form of the preconditioned system. The question of good preconditioning for the Schur complement S (matrix $\hat{S}$) is usually treated by using results like the one in Lemma 4.1. For example, for the steady-state Stokes problem we choose $\hat{S} = M$, the pressure-mass matrix. Again, this strategy is not black-box, as it needs an external preconditioning operator for $\hat{S}$, which risks to lack robustness with respect to large elongation $\mathcal{L}$ of the computational domain Ω and to smaller timesteps τ.

Braess-Sarazin method: the principal disadvantage of the pressure Schur complement method is the fact that S is full, which hinders the construction of an algebraic preconditioner for (4).

Let D be a suitable preconditioner for A such that D^{-1} and $(BD^{-1}B^T)$ can be assembled and easily evaluated. The solution of a similar system

$$\begin{pmatrix} D & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{q} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix} \tag{6}$$

is then more easily determined by solving the following auxiliary linear systems,

$$D\mathbf{v}' = \mathbf{f}, \qquad \tilde{S}\mathbf{q} = B\mathbf{v}' - \mathbf{g}, \qquad D\mathbf{v} = \mathbf{f} - B^T\mathbf{q}, \tag{7}$$

where $\tilde{S} \equiv BD^{-1}B^T + C$ is the inexact pressure Schur complement. Please note, that unlike the exact Schur complement $\hat{S} = BA^{-1}B^T + C$, the inexact Schur complement $\tilde{S}$ is sparse and even SPD, if D^{-1} is a sparse SPD matrix. The sparsity of $\tilde{S}$ enables its explicit assembly, which in turn means that an algebraic preconditioner for $\tilde{S}$ can be constructed.

In practice we simply take $D = \chi\,\mathrm{diag}(A)$, ie. we extract the diagonal part of A. Usually we take $\chi \in [0.5, 1.0]$, in function of the timestep τ ($\chi = 0.5$ for $\tau \to \infty$ and $\chi = 1.0$ for $\tau \to 0$).

Most of the time, the Braess-Sarazin method is used as a Richardson iteration, i.e. as a preconditioner for an iterative solution $(\mathbf{u}_j, \mathbf{p}_j)$ of the saddle point problem (3),

$$\begin{pmatrix} \mathbf{u}_{j+1} \\ \mathbf{p}_{j+1} \end{pmatrix} = \begin{pmatrix} \mathbf{u}_j \\ \mathbf{p}_j \end{pmatrix} - \begin{pmatrix} D & B^T \\ B & -C \end{pmatrix}^{-1} \left[\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} \mathbf{u}_j \\ \mathbf{p}_j \end{pmatrix} - \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix} \right], \tag{8}$$

for $j = 0, 1, 2, \ldots$. Provided that $\tilde{S}$ is inversed exactly, the scheme (8) satisfies for any j the incompressibility constraints which are defined for the original problem (1), i.e. $B\mathbf{u}_j - C\mathbf{p}_j = \mathbf{g}$. Thus, it can be formally represented as an iterative scheme on velocities only, with the iteration matrix

$$\mathbf{u} - \mathbf{u}_{j+1} = (I - D^{-1}B^T\tilde{S}^{-1}B)(I - D^{-1}A)(\mathbf{u} - \mathbf{u}_j). \tag{9}$$

If D is chosen so that $\rho(I - D^{-1}A) \leq 1$ and $\tilde{S}$ is inversed exactly, one can show convergence of (8) in the D-norm $\|\cdot\|_D = (D\cdot,\cdot)^{\frac{1}{2}}$,

$$\|\mathbf{u} - \mathbf{u}_{j+1}\|_D \leq \|\mathbf{u} - \mathbf{u}_j\|_D.$$

Also, a very good smoothing property holds: there exists a (pressure) vector $\mathbf{q} \in Y_h$ such that

$$\|A(\mathbf{u} - \mathbf{u}_j) + B^T\mathbf{q}\|_D \leq C\frac{\rho(D)}{j}\|\mathbf{u} - \mathbf{u}_0\|_D.$$

For more details see [2,15]. The advantage of this method is its algebraic construction based only on the Stokes system matrix K. Also note, that unlike the bad preconditioning result of Lemma 4.1 for the limit of small timesteps τ, the convergence of this method is actually improving, because A is tending to a (sparse and SPD) diagonal matrix $D = \mathrm{diag}(A)$ as $\tau \to 0$ ($\chi \to 1$).

3.2 Generalization of multigrid approach for saddle-point problems

Apart from the approaches of Section 3.1, one can think of applying the idea of multiple resolution scales directly to the whole problem (3). In this case, however, special care is needed when addressing the complementarity of the coarse grid correction and of the smoother. In return, however, the scheme might have independent behaviour regardless of the shape of Ω. At least, this has been observed for multigrids for SPD systems [3].

We are proposing a monolithic multigrid V-cycle as a preconditioner for K. We denote the finest multigrid level by an index J, the corresponding finite element spaces $X_J = X_h$, $Y_J = Y_h$, with vectors $\mathbf{u}_J = \mathbf{u}$, $\mathbf{p}_J = \mathbf{p}$, and operators $A_J = A$, $B_J = B$, $C_J = C$. The discrete problem on the finest level reads

$$\begin{pmatrix} A_J & B_J^T \\ B_J & -C_J \end{pmatrix} \begin{pmatrix} \mathbf{u}_J \\ \mathbf{p}_J \end{pmatrix} = \begin{pmatrix} \mathbf{f}_J \\ \mathbf{g}_J \end{pmatrix}, \quad K_J \equiv \begin{pmatrix} A_J & B_J^T \\ B_J & -C_J \end{pmatrix}. \tag{10}$$

Let us precondition (10) by the following multigrid V-cycle algorithm $\mathcal{V}(\nu_1, \nu_2)$.

Algorithm 1 *Set $\mathbf{u}_k = 0$ and $\mathbf{p}_k = 0$ and perform one multigrid V-cycle $\mathcal{V}(\nu_1, \nu_2)$ on each level k:*

1. *Braess-Sarazin pre-smoother: repeat ν_1 times:*
$$\begin{pmatrix} \mathbf{u}_k \\ \mathbf{p}_k \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{u}_k \\ \mathbf{p}_k \end{pmatrix} - \begin{pmatrix} D_k & B_k^T \\ B_k & -C_k \end{pmatrix}^{-1} \left[\begin{pmatrix} A_k & B_k^T \\ B_k & -C_k \end{pmatrix} \begin{pmatrix} \mathbf{u}_k \\ \mathbf{p}_k \end{pmatrix} - \begin{pmatrix} \mathbf{f}_k \\ \mathbf{g}_k \end{pmatrix} \right].$$
2. *Coarse-grid correction: if $k > 1$,*
   - (a) *Restrict the residual on the level k to the get right-hand side on the coarse level $(k-1)$:*
     $\mathbf{f}_{k-1} \leftarrow (I_k^X)^T \left[\mathbf{f}_k - A_k\mathbf{u}_k - B_k^T\mathbf{p}_k\right],$
     $\mathbf{g}_{k-1} \leftarrow (I_k^Y)^T \left[\mathbf{g}_k - B_k\mathbf{u}_k + C_k^T\mathbf{p}_k\right].$
   - (b) *Solve for $(\mathbf{u}_{k-1}, \mathbf{p}_{k-1})$ on the coarse-level*
     $$\begin{pmatrix} A_{k-1} & B_{k-1}^T \\ B_{k-1} & -C_{k-1} \end{pmatrix} \begin{pmatrix} \mathbf{u}_{k-1} \\ \mathbf{p}_{k-1} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_{k-1} \\ \mathbf{g}_{k-1} \end{pmatrix} \tag{11}$$
     by recursive application of this algorithm.
   - (c) *Prolongation: correct $(\mathbf{u}_k, \mathbf{p}_k)$ by*
     $\mathbf{u}_k \leftarrow \mathbf{u}_k + I_k^X\mathbf{u}_{k-1},$
     $\mathbf{p}_k \leftarrow \mathbf{p}_k + I_k^Y\mathbf{p}_{k-1}.$
3. *Braess-Sarazin post-smoother: repeat ν_2 times:*
$$\begin{pmatrix} \mathbf{u}_k \\ \mathbf{p}_k \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{u}_k \\ \mathbf{p}_k \end{pmatrix} - \begin{pmatrix} D_k & B_k^T \\ B_k & -C_k \end{pmatrix}^{-1} \left[\begin{pmatrix} A_k & B_k^T \\ B_k & -C_k \end{pmatrix} \begin{pmatrix} \mathbf{u}_k \\ \mathbf{p}_k \end{pmatrix} - \begin{pmatrix} \mathbf{f}_k \\ \mathbf{g}_k \end{pmatrix} \right].$$

Algorithm 1 uses two prolongation operators $I_k^X : X_{k-1} \mapsto X_k$ and $I_k^Y : Y_{k-1} \mapsto Y_k$ for the transfer of solution updates to finer level and restriction of residuals to coarse level. They work separately in either velocity or pressure spaces X_k, resp. Y_k.

Coarse-level spaces by smoothed aggregation: The velocity and pressure transfer operators are constructed by the technique of smoothed aggregation, as described and analysed in [17] for SPD problems. The P1-P1 finite element basis functions of spaces (X_h, Y_h) share a common support – the mesh. To keep the coarse space basis functions for velocities and pressures also colocated, both prolongation operators I_k^X and I_k^Y are based on the same coloring of nodal aggregates.

On the current level k, we create a disjoint covering of all n_k nodes by a set of n_{k-1} nodal aggregates, see Fig. 1. We denote $\mathcal{A}_k(p)$, $p = 1, \ldots, n_{k-1}$ the sets of finer nodes contained in the aggregate p. The aggregation is done based on the connectivity of nodes of the mesh. This information is extracted from the sparse structure of the original discrete operator $K_J = K$. Neighbouring nodes are agglomerated by a simple greedy algorithm to create aggregates with approximately 3 nodes in diameter of the graph of the mesh, ie. the coarsening ratio is $R = 3$.

Based on the agglomeration of nodes, we define tentative pair of coarser spaces $(\tilde{X}_{k-1}, \tilde{Y}_{k-1})$, $\tilde{X}_{k-1} \subset X_k$, $\tilde{Y}_{k-1} \subset Y_k$ of functions aggregate-wise constant (on each type of degree of freedom separately). The tentative spaces are simple, but they do not satisfy the energy stability properties required for discretization of second-order partial differential operators. That is why, in the second step, we perform the aggregation smoothing.

First, we define tentative zero-one discrete injection operators (matrices) $\tilde{I}_k^X : \tilde{X}_{k-1} \mapsto X_k$ and $\tilde{I}_k^Y : \tilde{Y}_{k-1} \mapsto Y_k$. Hence, we can also say that $\tilde{X}_{k-1} \equiv \text{Range}\, \tilde{I}_k^X$ and $\tilde{Y}_{k-1} \equiv \text{Range}\, \tilde{I}_k^Y$.

Second, we smooth the tentative prolongation matrices to create the prolongation operators I_k^X and I_k^Y by

$$\begin{aligned} I_k^X &= \left(I - \tfrac{\omega}{\rho(A_k)} A_k\right) \tilde{I}_k^X\,, \\ I_k^Y &= \left(I - \tfrac{\omega}{\rho(C_k)} C_k\right) \tilde{I}_k^Y\,, \end{aligned} \tag{12}$$

where A_k and C_k are the velocity-velocity and pressure-pressure blocks of the coarse-level operator, respectively. Please note, that C_k is supposed symmetric positive semi-definite. The resulting coarse space of the level $(k-1)$ is then formed by $X_{k-1} \equiv \text{Range}\, I_k^X \subset X_k$ and $Y_{k-1} \equiv \text{Range}\, I_k^Y \subset Y_k$.

Assembly of coarse-level problems: Corresponding to the prolongation and restriction operators, we would like to construct the coarse grid systems K_{k-1} by a Ritz-Galerkin procedure

$$\begin{pmatrix} A_{k-1} & B_{k-1}^T \\ B_{k-1} & -C_{k-1} \end{pmatrix} = \begin{pmatrix} (I_k^X)^T & 0 \\ 0 & (I_k^Y)^T \end{pmatrix} \begin{pmatrix} A_k & B_k^T \\ B_k & -C_k \end{pmatrix} \begin{pmatrix} I_k^X & 0 \\ 0 & I_k^Y \end{pmatrix},$$

or, in a block-wise notation,

$$A_{k-1} = (I_k^X)^T A_k I_k^X \quad , \quad B_{k-1} = (I_k^Y)^T B_k I_k^X \tag{13}$$

and

$$C_{k-1} = (I_k^Y)^T C_k I_k^Y\,. \tag{14}$$

However, caution must be taken [19] about the inf-sup stability condition of the pair of coarse-level discretization spaces (X_k, Y_k). Thus, the formulation (14) is known to gradually lose its stability properties with coarsening.

Following recent results of Wabro [19], we make a small deviation from the Ritz-Galerkin framework for C_{k-1}. We keep Galerkin definitions (13) of A_{k-1} and B_{k-1}, and we consider, besides the formulation (14), also the following formulas,

$$C_{k-1} = \sigma (I_k^Y)^T C_k I_k^Y\,, \quad \text{or} \tag{15}$$

$$C_{k-1} = \sigma (\tilde{I}_k^Y)^T C_k \tilde{I}_k^Y\,. \tag{16}$$

Wabro provides in [19] stability theory for the choice (15) with $\sigma \approx R^2$, the square of the coarsening ratio. We will conduct numerical experiments with the different formulations at the beginning of Section 5, in order to validate the theoretical stability results, and to see what is the influence of the coarse-level inf-sup constants on the multigrid convergence.

4 Pressure Schur complement preconditioner

Recent results [12] show that also for the P1-P1 finite element GLS scheme the mass-matrix in the pressure space Y_h is an h-independent preconditioner for the pressure Schur complement S for the steady-state Stokes problem. Let us restate this result, with a special attention not only to dependencies of constants on h, but also on the shape and form of Ω.

Lemma 4.1. *Let A, B and C be matrices introduced in (3), M is the pressure mass-matrix, $M_{ij} = (\varphi_i, \varphi_j)_{0,\Omega}$, and α denotes the GLS stabilization parameter. There exist two constants $c_1, c_2 > 0$, depending on the mesh smallest angle, the shape and elongation $\mathcal{L}$ of the domain Ω, such that for all $\mathbf{q} \in Y_h$, $\mathbf{q} \neq 0$ we have*

$$\frac{c_1}{\mu\left(1 + \frac{1}{\alpha} + \frac{\rho}{\mu\tau}\right)} \leq \frac{(\mathbf{q}, (BA^{-1}B^T + C)\mathbf{q})}{(\mathbf{q}, M\mathbf{q})} \leq c_2 \frac{1+\alpha}{\mu}. \tag{17}$$

Proof. For full proof please refer to [12]. Let us only briefly discuss its main ingredients with respect to their dependencies on the geometry of the computational domain Ω.

The upper bound is obtained by applying the following inverse inequality: for all $q_h \in Y_h$ there is

$$C_1 \sum_{K \in \tau_h} h_K^2 \|\nabla q_h\|_{0,K}^2 \leq \|q_h\|_{0,\Omega}^2, \tag{18}$$

with C_1 depending only on the mesh smallest angle.

To obtain the lower bound, one needs to use Poincaré's inequality and the Verfürth's trick [18], which both contain assumptions on the shape of the domain. Lemma 3.3 in [9] states that there are two positive constants C_2, C_3 independent of h, but dependent on the characteristics of the bounded domain Ω (star-shape and elongation $\mathcal{L}$) such that for all $q_h \in Y_h$ with $\int_\Omega q_h = 0$ we have

$$\sup_{v_h \neq 0} \frac{\int_\Omega q_h \operatorname{div} v_h}{\|\nabla v_h\|_{0,\Omega}} \geq C_2 \|q_h\|_{0,\Omega} - C_3 \|q_h\|_h, \tag{19}$$

where $\|q_h\|_h^2 = \sum_{K \in \tau_h} h_K^2 \|\nabla q_h\|_{0,K}^2$.

Please note, that for the limit without GLS stabilization $\alpha \to 0$, the lower bound in (17) vanishes – the Schur complement matrix becomes singular. For all our tests, we take $\alpha = 0.01$.

As $\mathrm{cond}(S, M) = \frac{c_2}{c_1}(1+\alpha)\left(1 + \frac{1}{\alpha} + \frac{\rho}{\mu\tau}\right)$, it might appear, that the pressure mass matrix is an excellent preconditioner for the pressure Schur complement, at least for the limit $\tau \to \infty$. Indeed, this consideration is exploited in many solution methods. Nevertheless, for elongated domains, there are several shortcomings, even in the limit case of the steady-state Stokes problem.

We have noticed that proof of Lemma 4.1 uses the Poincaré's inequality and the Verfürth's trick, whose constants depend on the shape of Ω and, most importantly, on its elongation $\mathcal{L}$. Verfürth's trick compares the inf-sup constant of the actual finite element spaces (X_h, Y_h) to a reference configuration on which inf-sup condition is verified. However, the resulting estimates cannot be Ω-independent, since even for the original non-discretized spaces (X, Y) the continuous inf-sup (LBB) constant

$$0 < C_{LBB} = \inf_{q \in Y(\Omega)} \sup_{v \in X(\Omega)} \frac{\int_\Omega q \operatorname{div} v}{\|q\|_{0,\Omega}\|v\|_{1,\Omega}}$$

depend on elongation $\mathcal{L}$. The analysis of the C_{LBB} constant for several types of domains in [5] etimates that $C_{LBB} \approx \mathcal{O}(\mathcal{L}^{-1})$.

Let us now turn our attention to inf-sup stability properties of coarse operators. Using the stability result in [19], we can state the equivalent of Lemma 4.1 for coarse spaces.

Lemma 4.2. *Let A_k, B_k and C_k be matrices introduced in (13) and (15) with $\sigma = R^2$, where R is the coarsening ratio. Let M_k be the coarse-level pressure mass-matrix introduced by the recurrence $M_{\ell-1} = (I_\ell^Y)^Y M_\ell I_\ell^Y$, $\ell = \{J, J-1, \ldots, k+1\}$, $M_J = M$. Let α denote the GLS stabilization parameter. There exist two constants $c_1, c_2 > 0$, independent of the multigrid level k, such that for all $\mathbf{q} \in Y_k$, $\mathbf{q} \neq 0$ we have*

$$\frac{c_1}{\mu\left(1 + \frac{1}{\alpha} + \frac{\rho}{\mu\tau}\right)} \le \frac{\left(\mathbf{q}, (B_k A_k^{-1} B_k^T + C_k)\mathbf{q}\right)}{(\mathbf{q}, M_k \mathbf{q})} \le c_2 \frac{1+\alpha}{\mu}. \quad (20)$$

Proof. The proof is formally the same as the proof of Lemma 4.1, in which the fine-level inverse inequality (18) is replaced by

$$C_1 \sum_{\substack{p=1 \\ \bar{\mathcal{A}}_k(p)}}^{n_k} \mathrm{diam}(\bar{\mathcal{A}}_k(p))^2 \|\nabla q_h\|^2_{0,\bar{\mathcal{A}}_k(p)} \le \|q_h\|^2_{0,\Omega},$$

for $q_h \in Y_k$. Here, $\bar{\mathcal{A}}_k(p) = \{K \in \tau_h : \exists i \in K, i \in \mathcal{A}_k(p)\}$, is a set of fine-level finite elements covering the nodes of the p-th nodal aggregate, on the multigrid level k. Its characteristic diameter is denoted by

$$\mathrm{diam}(\bar{\mathcal{A}}_k(p)) = R^{J-k} \max_{K \in \bar{\mathcal{A}}_k(p) \cap \tau_h} h_K.$$

The norm $\|\cdot\|_h$ is generalized on the level k by

$$\|q_h\|_h^2 = \sum_{K \in \tau_h} R^{2(J-k)} h_K^2 \|\nabla q_h\|^2_{0,K} = \frac{\mu}{\alpha}\mathbf{q}^T C_k \mathbf{q},$$

for any function $q_h \in Y_k \subset Y_J$, whose discrete counterpart is $\mathbf{q} \in Y_k$

The estimate (19) is still valid for $q \in Y_k$ and $v_h \in X_k$, by virtue of Lemma 1 in [19].

5 Numerical experiments

Let us perform series of tests on a simple 2D Poiseuille flow problem. First, we will concentrate on the stability properties of the coarse-grid operators, depending on the choice of assembly formula for C_{k-1}. Then we perform a study of performance of the multigrid preconditioner with respect to different time-steps and elongated domains. Then we pass on a 3D flow example.

If not stated otherwise, we are using a $\mathcal{V}(3,3)$-cycle multigrid preconditioner of Algorithm 1 with 3 steps of Braess-Sarazin pre- and post-smoothing, with $D = 0.5 \cdot \mathrm{diag}(A)$, and with the inexact Schur complement $\tilde{S}$ approximatively inversed by one backward substitution of the ILU(0) method inside each iteration of the Braess-Sarazin method (7). Prolongation smoothers (12) are tuned by $\omega = 4/3$, like in [17]. All methods are considered converged when the relative residual euclidean norm drops by a factor of 10^{-10}.

Poiseuille flow on elongated domains in 2D: We pose the Poiseuille problem successively for different computational domains $\Omega \equiv (-\mathcal{L}, \mathcal{L}) \times (-1, -1)$ with different elongations $\mathcal{L}$, different timesteps τ and different isotropic mesh sizes h. In each case, we are using the same P1-P1 finite element scheme with Brezzi-Pitkäranta-like stabilization (2) on an isotropic mesh, with the stabilization parameter $\alpha = 0.01$, $\mu = 1$ and $\rho = 1$. We are looking for $(u, p) \in (X_h, Y_h)$ such that

$$\begin{aligned} \frac{u}{\tau} - \operatorname{div}\nabla u + \nabla p &= 0 \quad \text{in } \Omega, \\ \operatorname{div} u &= 0 \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \Gamma_D, \\ \left(\frac{\partial u_i}{\partial x_j} - \delta_{ij} p\right) n_j &= F_i \quad \text{on } \Gamma_F, \end{aligned}$$

with the domain geometry Ω as in Fig. 1, for $\mathcal{L} = 2$ and isotropic mesh parameter $h = 1/16$. A horizontal driving force is prescribed on the left boundary of the pipe, and zero velocity conditions are imposed on the horizontal pipe walls. On the right side, horizontal outflow is required. The nodal aggregates for the first 3 multigrid levels are depicted in Fig. 1.

Stability of coarse-grid operators: First, we assemble the coarse problems on all multigrid levels by formulas (13) and (15) or (16) and we validate numerically the stability results of Lemma 4.2. We use a particular fine mesh with $\mathcal{L} = 1$ and $h = 1/364$, for which we generate six multigrid levels ($k = J = 6$ is the finest one) with coarsening ratio $R = 3$. We take $\tau = \infty$, $\mu = 1$, $\alpha = 0.01$. Table 1 gives the estimates of spectral bounds and conditioning of $M_k^{-1} S_k$ for different σ's

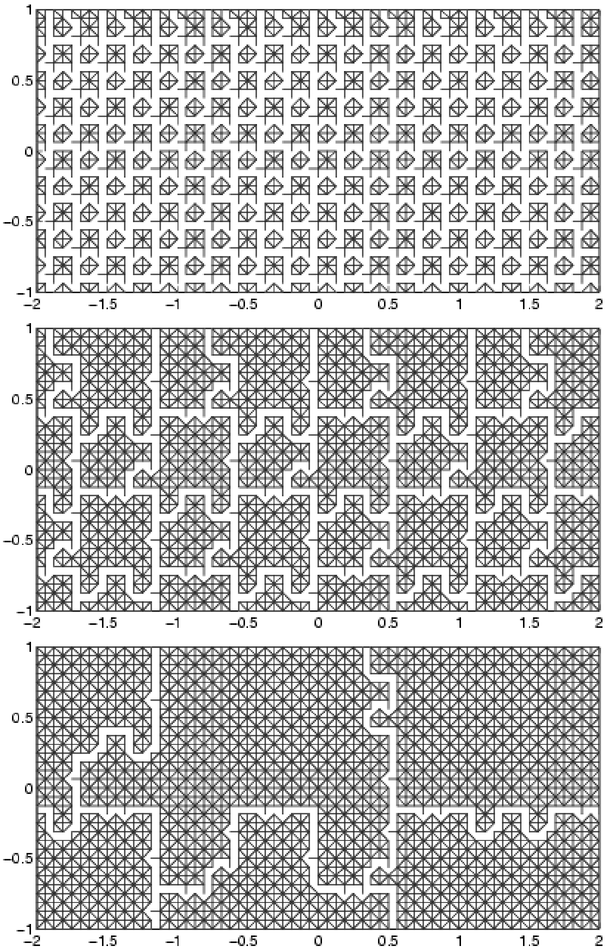

Fig. 1. Poiseuille 2D flow: nodal aggregates on 3 levels, elongation $\mathcal{L} = 2$, $h = 1/16$.

in (15) or (16). We clearly see, that the formulation with $C_{k-1} = R^2 \cdot (I_k^Y)^T C_k I_k^Y$, proposed by Wabro [19], is uniformly stable (i.e. the condition number $\mathrm{cond}(S_k, M_k)$ is bounded independently of k). So is the choice $C_{k-1} = R \cdot (\tilde{I}_k^Y)^T C_k \tilde{I}_k^Y$.

Next, we would like to see the effect of coarse-level stabilization of formulas (15) and (16) on the multigrid convergence speed. We observe the convergence history of 6-level multigrid $\mathcal{V}(1,1)$- and $\mathcal{V}(3,3)$-cycles, used as simple iterations without outer Krylov loop. The results for a steady-state problem ($\tau = \infty$) and an evolutive one with a small timestep ($\tau = 10^{-8}$) are presented in Figs. 2 and 3. We clearly see, that the choices $C_{k-1} = R^2 \cdot (I_k^Y)^T C_k I_k^Y$ and $C_{k-1} = R \cdot (\tilde{I}_k^Y)^T C_k \tilde{I}_k^Y$, which are uniformly stable on all levels, need more smoothing, otherwise they perform quite badly (the $\mathcal{V}(1,1)$-cycle diverges for both problems). At the same time, we observe that the best choice as to convergence speed would be the purely Galerkin formulation (14). Therefore, we opt for this choice in the sequel.

MG-lvl k	$\lambda_{\min}(M_k^{-1} S_k)$	$\lambda_{\max}(M_k^{-1} S_k)$	$\mathrm{cond}(S_k, M_k)$
$C_{k-1} = \left(I_k^Y\right)^T C_k I_k^Y$			
fine 6	$8.351 \cdot 10^{-2}$	$9.999 \cdot 10^{-1}$	$1.197 \cdot 10^{1}$
5	$1.247 \cdot 10^{-2}$	$8.656 \cdot 10^{-1}$	$6.939 \cdot 10^{1}$
4	$2.212 \cdot 10^{-3}$	$7.162 \cdot 10^{-1}$	$3.236 \cdot 10^{2}$
3	$1.901 \cdot 10^{-3}$	$6.990 \cdot 10^{-1}$	$3.677 \cdot 10^{2}$
2	$1.227 \cdot 10^{-2}$	$6.760 \cdot 10^{-1}$	$5.509 \cdot 10^{1}$
coarse 1	$5.773 \cdot 10^{-2}$	$4.116 \cdot 10^{-1}$	$7.129 \cdot 10^{0}$
$C_{k-1} = 3 \cdot \left(I_k^Y\right)^T C_k I_k^Y$			
fine 6	$8.351 \cdot 10^{-2}$	$9.999 \cdot 10^{-1}$	$1.197 \cdot 10^{1}$
5	$3.620 \cdot 10^{-2}$	$8.684 \cdot 10^{-1}$	$2.398 \cdot 10^{1}$
4	$1.778 \cdot 10^{-2}$	$7.165 \cdot 10^{-1}$	$4.029 \cdot 10^{1}$
3	$7.577 \cdot 10^{-3}$	$6.990 \cdot 10^{-1}$	$9.226 \cdot 10^{1}$
2	$1.380 \cdot 10^{-2}$	$6.762 \cdot 10^{-1}$	$4.897 \cdot 10^{1}$
coarse 1	$5.797 \cdot 10^{-2}$	$4.120 \cdot 10^{-1}$	$7.107 \cdot 10^{0}$
$C_{k-1} = 9 \cdot \left(I_k^Y\right)^T C_k I_k^Y$			
fine 6	$8.351 \cdot 10^{-2}$	$9.999 \cdot 10^{-1}$	$1.197 \cdot 10^{1}$
5	$9.933 \cdot 10^{-2}$	$8.764 \cdot 10^{-1}$	$8.823 \cdot 10^{0}$
4	$1.060 \cdot 10^{-1}$	$7.197 \cdot 10^{-1}$	$6.787 \cdot 10^{0}$
3	$1.057 \cdot 10^{-1}$	$7.002 \cdot 10^{-1}$	$6.624 \cdot 10^{0}$
2	$1.039 \cdot 10^{-1}$	$6.854 \cdot 10^{-1}$	$6.594 \cdot 10^{0}$
coarse 1	$9.172 \cdot 10^{-2}$	$5.212 \cdot 10^{-1}$	$5.682 \cdot 10^{0}$
$C_{k-1} = \left(\tilde{I}_k^Y\right)^T C_k \tilde{I}_k^Y$			
fine 6	$8.351 \cdot 10^{-2}$	$9.999 \cdot 10^{-1}$	$1.197 \cdot 10^{1}$
5	$2.569 \cdot 10^{-2}$	$8.671 \cdot 10^{-1}$	$3.375 \cdot 10^{1}$
4	$8.298 \cdot 10^{-3}$	$7.113 \cdot 10^{-1}$	$8.571 \cdot 10^{1}$
3	$3.646 \cdot 10^{-3}$	$6.968 \cdot 10^{-1}$	$1.910 \cdot 10^{2}$
2	$9.091 \cdot 10^{-3}$	$6.593 \cdot 10^{-1}$	$7.253 \cdot 10^{1}$
coarse 1	$4.791 \cdot 10^{-2}$	$3.367 \cdot 10^{-1}$	$7.027 \cdot 10^{0}$
$C_{k-1} = 3 \cdot \left(\tilde{I}_k^Y\right)^T C_k \tilde{I}_k^Y$			
fine 6	$8.351 \cdot 10^{-2}$	$9.999 \cdot 10^{-1}$	$1.197 \cdot 10^{1}$
5	$7.221 \cdot 10^{-2}$	$8.729 \cdot 10^{-1}$	$1.208 \cdot 10^{1}$
4	$6.552 \cdot 10^{-2}$	$7.124 \cdot 10^{-1}$	$1.087 \cdot 10^{1}$
3	$6.359 \cdot 10^{-2}$	$6.975 \cdot 10^{-1}$	$1.096 \cdot 10^{1}$
2	$6.339 \cdot 10^{-2}$	$6.656 \cdot 10^{-1}$	$1.050 \cdot 10^{1}$
coarse 1	$7.788 \cdot 10^{-2}$	$4.022 \cdot 10^{-1}$	$5.164 \cdot 10^{0}$

Table 1. Stability properties on multigrid levels k ($k = 6$ finest, $k = 1$ coarsest), measured by $\mathrm{cond}(S_k, M_k)$ for different formulations of coarse-problems. Poiseuille flow in 2D on a square $\Omega = (-1, 1) \times (-1, 1)$ with isotropic fine mesh ($h = 1/364$, 531441 nodes), timestep $\tau = \infty$ (steady-state problem).

Robustness with respect to $\mathcal{L}$ and τ: Let us test now the performance of one $\mathcal{V}(3,3)$-cycle as a preconditioner for GMRES. In particular, we would like to experiment with the robustness of the preconditioner with respect to the mesh-size h, the timestep τ and the geometry elongation $\mathcal{L}$.

We are comparing 3 methods: (i) a GMRES method preconditioned by one multigrid $\mathcal{V}(3,3)$-cycle of Algorithm 1, denoted further (MG), (ii) a GMRES method preconditioned by a block LU preconditioner (5), as in [7], denoted further (LU), and (iii) a preconditioned conjugate gradient method for the system (4), i.e. for

$$S\mathbf{p} = BA^{-1}\mathbf{f} - \mathbf{g},$$

with $S \equiv BA^{-1}B^T + C$. This method is further denoted by (CG). For the (LU) and (CG) methods, we use preconditioning of the pressure Schur complement S by $\hat{S} = M$, the pressure mass-matrix. Calculating the

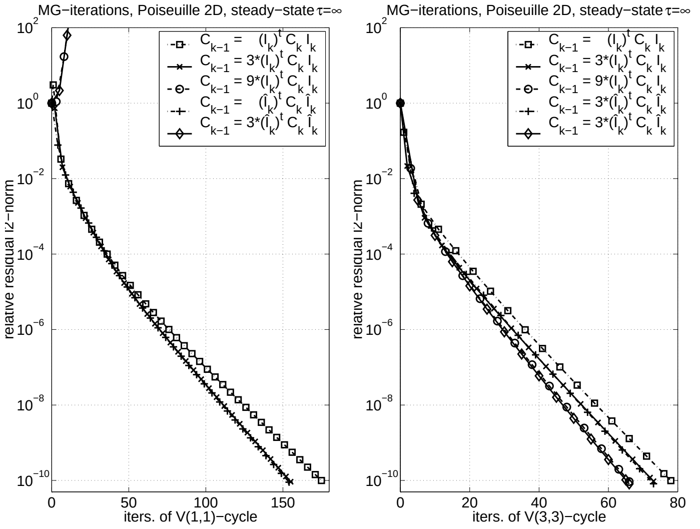

Fig. 2. Convergence history for MG-iterations of $\mathcal{V}(1,1)$- and $\mathcal{V}(3,3)$-cycles for different formulations of coarse-problems. Poiseuille flow in 2D on a square $\Omega = (-1,1) \times (-1,1)$ with isotropic fine mesh ($h = 1/364$, 531441 nodes), timestep $\tau = \infty$ (steady-state problem).

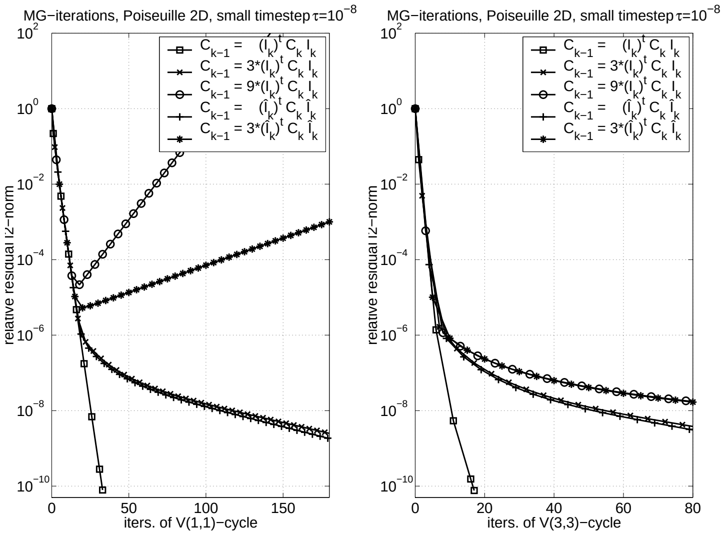

Fig. 3. Convergence history for MG-iterations of $\mathcal{V}(1,1)$- and $\mathcal{V}(3,3)$-cycles for different formulations of coarse-problems. Poiseuille flow in 2D on a square $\Omega = (-1,1) \times (-1,1)$ with isotropic fine mesh ($h = 1/364$, 531441 nodes), timestep $\tau = 10^{-8}$ (small).

residual in the (CG) method involves the multiplication $S\mathbf{p}$ containing the evaluation of A^{-1}. For this sake we make a call to UMFPACK direct solver, although a few V-cycles of a SPD multigrid method would be sufficient. This is to avoid supplementary effects of the incomplete solves for A and rather concentrate on problems with the pressure Schur complement preconditioning. At the same time, the (CG) method generates data to Lanczos estimation of eigenvalues $\lambda_{\min}(S,M)$ and $\lambda_{\max}(S,M)$ of the generalized eigenvalue problem $S\mathbf{q} = \lambda M\mathbf{q}$.

We observe the relative condition number of the exact pressure Schur complement S with respect to the pressure mass matrix M in function of the mesh size $h \in \{1/8, 1/16, 1/32\}$, the elongation $\mathcal{L} \in \{1,2,4,8,64\}$ of the domain Ω and timestep values $\tau \in \{\infty, 1, 10^{-2}, 10^{-4}\}$. We can see in Tabs. 2, 3, 4, 5 the estimations of $\operatorname{cond}(S,M)$. The iteration count denotes the number of iterations to achieve convergence. Depending on the method, it is the number of overall iterations of the preconditioned conjugate gradient method (CG), or the number of exterior iterations of the GMRES method preconditioned either by a block LU preconditioner (LU), or by one $\mathcal{V}(3,3)$-cycle of the aggregation multigrid (MG).

We clearly see the independence on the mesh-size h, and the dramatic dependence of the two mass-matrix based preconditioners, in the methods (LU) and (CG), on the geometry elongation $\mathcal{L}$ and the timestep τ. On the other hand, the aggregation multigrid (MG) performs quite well in all cases.

Elongated geometry in 3D: The last test shows an application of the multigrid preconditioner for a 3D testcase with elongated geometry, for which mass-matrix preconditioning does not work well, even for the steady problem $\tau \to \infty$.

Six pipes with prescribed forces on their ends are crossed, creating quite complicated velocity and pressure fields, see Fig. 4. The employed mesh is of isotropic Delaunay type, with 104383 nodes ($h = 0.25$) and 723558 nodes ($h = 0.125$). The aggregates on the first 4 levels of the 104383 node mesh are in Fig. 6. The convergence histories for the mass-matrix preconditioned CG metod for S, the block LU preconditioner for the monolithic GMRES and the algebraic multigrid preconditioner for the monolithic GMRES are in Fig. 5. Clearly, mass-matrix is not a good preconditioner for pressure Schur complement. Indeed, the estimation of $\operatorname{cond}(S,M)$ for the 104383 node mesh is of about $1.7 \cdot 10^3$.

6 Conclusions

We have tested a generalization of the smoothed aggregation multigrid [17] for saddle-point problems. Similar methods have been introduced and applied to engineering problems in [1] and their stability properties were analysed in [19].

We have made experiments with the inf-sup stability on coarse levels. It seems, according to our numerical tests, that uniform inf-sup stability of coarse-level operators is not strictly necessary for obtaining a successful preconditioner.

We have also pointed out, that the smoothed aggregation multigrid V-cycle is a suitable preconditioner of GMRES, quite robust for different timesteps and for high elongations of the computational domain. These two characteristics are reputed obstacles to black-box behaviour for other methods, e.g. [7]. Thanks to the algebraic way of constructing coarse levels and smoothers, the method can be used as a black-box solver.

							iterations		
τ	$\mathcal{L}$	h	nodes	$\lambda_{\min}(S,M)$	$\lambda_{\max}(S,M)$	$\mathrm{cond}(S,M)$	(CG)	(LU)	(MG)
∞	1	1/8	289	$1.234 \cdot 10^{-1}$	$9.865 \cdot 10^{-1}$	$7.997 \cdot 10^{0}$	34	25	–
∞	1	1/16	1089	$1.209 \cdot 10^{-1}$	$9.957 \cdot 10^{-1}$	$8.238 \cdot 10^{0}$	35	24	18
∞	1	1/32	4225	$1.205 \cdot 10^{-1}$	$9.974 \cdot 10^{-1}$	$8.278 \cdot 10^{0}$	34	23	18
∞	2	1/8	561	$9.060 \cdot 10^{-2}$	$9.896 \cdot 10^{-1}$	$1.092 \cdot 10^{1}$	36	27	–
∞	2	1/16	2145	$9.097 \cdot 10^{-2}$	$9.966 \cdot 10^{-1}$	$1.096 \cdot 10^{1}$	36	26	18
∞	2	1/32	8385	$9.106 \cdot 10^{-2}$	$9.978 \cdot 10^{-1}$	$1.096 \cdot 10^{1}$	36	25	19
∞	4	1/8	1105	$2.483 \cdot 10^{-2}$	$9.902 \cdot 10^{-1}$	$3.988 \cdot 10^{1}$	40	30	–
∞	4	1/16	4257	$2.490 \cdot 10^{-2}$	$9.968 \cdot 10^{-1}$	$4.003 \cdot 10^{1}$	39	29	19
∞	4	1/32	16705	$2.491 \cdot 10^{-2}$	$9.981 \cdot 10^{-1}$	$4.006 \cdot 10^{1}$	39	27	18
∞	8	1/8	2193	$6.355 \cdot 10^{-3}$	$9.904 \cdot 10^{-1}$	$1.559 \cdot 10^{2}$	45	35	–
∞	8	1/16	8481	$6.371 \cdot 10^{-3}$	$9.970 \cdot 10^{-1}$	$1.565 \cdot 10^{2}$	44	34	20
∞	8	1/32	33345	$6.375 \cdot 10^{-3}$	$9.984 \cdot 10^{-1}$	$1.566 \cdot 10^{2}$	44	32	20
∞	64	1/8	17425	$1.001 \cdot 10^{-4}$	$9.912 \cdot 10^{-1}$	$9.906 \cdot 10^{3}$	116	778	–
∞	64	1/16	67617	$1.003 \cdot 10^{-4}$	$9.977 \cdot 10^{-1}$	$9.947 \cdot 10^{3}$	116	723	21
∞	64	1/32	266305	$1.004 \cdot 10^{-4}$	$9.993 \cdot 10^{-1}$	$9.957 \cdot 10^{3}$	116	723	21

Table 2. Poiseuille flow 2D: timestep $\tau{=}\infty$. Preconditioning of S by M. Dependence of the condition number $\mathrm{cond}(S,M)$ on the mesh-size h and the geometry elongation $\mathcal{L}$. Convergence of mass-preconditioned CG method for the Schur complement system (CG), and of block LU (LU) and algebraic multigrid (MG) preconditioners for monolithic GMRES.

							iterations		
τ	$\mathcal{L}$	h	nodes	$\lambda_{\min}(S,M)$	$\lambda_{\max}(S,M)$	$\mathrm{cond}(S,M)$	(CG)	(LU)	(MG)
10^{0}	1	1/8	289	$1.233 \cdot 10^{-1}$	$9.607 \cdot 10^{-1}$	$7.791 \cdot 10^{0}$	34	25	–
10^{0}	1	1/16	1089	$1.209 \cdot 10^{-1}$	$9.774 \cdot 10^{-1}$	$8.087 \cdot 10^{0}$	35	24	14
10^{0}	1	1/32	4225	$1.204 \cdot 10^{-1}$	$9.845 \cdot 10^{-1}$	$8.176 \cdot 10^{0}$	35	22	14
10^{0}	2	1/8	561	$5.335 \cdot 10^{-2}$	$9.563 \cdot 10^{-1}$	$1.792 \cdot 10^{1}$	37	28	–
10^{0}	2	1/16	2145	$5.349 \cdot 10^{-2}$	$9.691 \cdot 10^{-1}$	$1.812 \cdot 10^{1}$	37	27	14
10^{0}	2	1/32	8385	$5.352 \cdot 10^{-2}$	$9.845 \cdot 10^{-1}$	$1.839 \cdot 10^{1}$	37	25	14
10^{0}	4	1/8	1105	$1.405 \cdot 10^{-2}$	$9.384 \cdot 10^{-1}$	$6.677 \cdot 10^{1}$	40	31	–
10^{0}	4	1/16	4257	$1.408 \cdot 10^{-2}$	$9.691 \cdot 10^{-1}$	$6.882 \cdot 10^{1}$	40	30	14
10^{0}	4	1/32	16705	$1.409 \cdot 10^{-2}$	$9.846 \cdot 10^{-1}$	$6.989 \cdot 10^{1}$	41	28	14
10^{0}	8	1/8	2193	$3.560 \cdot 10^{-3}$	$9.393 \cdot 10^{-1}$	$2.639 \cdot 10^{2}$	47	38	–
10^{0}	8	1/16	8481	$3.566 \cdot 10^{-3}$	$9.693 \cdot 10^{-1}$	$2.718 \cdot 10^{2}$	47	37	15
10^{0}	8	1/32	33345	$3.568 \cdot 10^{-3}$	$9.847 \cdot 10^{-1}$	$2.760 \cdot 10^{2}$	48	35	14
10^{0}	64	1/8	17425	$5.587 \cdot 10^{-5}$	$9.409 \cdot 10^{-1}$	$1.684 \cdot 10^{4}$	140	$> 10^{3}$	–
10^{0}	64	1/16	67617	$5.596 \cdot 10^{-5}$	$9.697 \cdot 10^{-1}$	$1.733 \cdot 10^{4}$	142	$> 10^{3}$	15
10^{0}	64	1/32	266305	$5.598 \cdot 10^{-5}$	$9.851 \cdot 10^{-1}$	$1.760 \cdot 10^{4}$	143	$> 10^{3}$	15

Table 3. Poiseuille flow 2D: timestep $\tau{=}10^{0}$. Preconditioning of S by M. Dependence of the condition number $\mathrm{cond}(S,M)$ on mesh-size h and geometry elongation $\mathcal{L}$. Convergence of mass-preconditioned CG method for the Schur complement system (CG), and of block LU (LU) and algebraic multigrid (MG) preconditioners for monolithic GMRES.

References

1. Adams, M.F.: Algebraic multigrid methods for constrained linear systems with applications to contact problems in solid mechanics, Numer. Linear Algebra Appl. 11 (2004), 141–153.
2. Braess, D.; Sarazin, R.: An efficient smoother for the Stokes problem, Appl. Numer. Math. 23 (1997), 3–19.
3. Brezina, M.; Vaněk, P.: A black-box iterative solver based on a two-level Schwarz method, Computing 63 (1999), no. 3, 233–263.
4. Cui, M.R.: Analysis of iterative algorithms of Uzawa type for saddle point problems, Appl. Numer. Math. 50 (2004), 133–146.
5. Dobrowolski, M.: On the LBB constant on stretched domains, Math. Nachr. 254–255 (2003), 64–67.
6. Elman, H.C.: Preconditioning strategies for models of incompressible flow, Research Report CS-TR no.4543/UMIACS TR no.2003-111, University of Maryland, November 2003.
7. Elman, H.C.; Silvester, D.J.; Wathen, A.J.: Performance and analysis of saddle point preconditioners for the discrete steady-state Navier-Stokes equations, Numer. Math. 90 (2002), 665–688.
8. Elman, H.C.; Howle, V.E.; Shadid, J.N.; Tuminaro, R.S.: A parallel block multi-level preconditioner for the 3D incompressible Navier-Stokes equations, J. Comput. Physics 187 (2003), 504–523.
9. Franca, L. and Stenberg, R.: Error analysis of some GLS methods for elasticity equations, SIAM J. Numer. Anal. 28 (1991), 1680–1697.
10. Griebel, M.; Neunhoffer, T.; Regler, H.: Algebraic multigrid methods for the solution of the Navier-Stokes equations in complicated geometries, Int. J. Numer. Methods Fluids 26 (1998), 281–301.

							iterations		
τ	$\mathcal{L}$	h	nodes	$\lambda_{\min}(S,M)$	$\lambda_{\max}(S,M)$	cond(S,M)	(CG)	(LU)	(MG)
10^{-2}	1	1/8	289	$5.733 \cdot 10^{-3}$	$7.055 \cdot 10^{-1}$	$1.231 \cdot 10^{2}$	38	31	–
10^{-2}	1	1/16	1089	$5.703 \cdot 10^{-3}$	$8.491 \cdot 10^{-1}$	$1.489 \cdot 10^{2}$	42	33	11
10^{-2}	1	1/32	4225	$5.696 \cdot 10^{-3}$	$8.611 \cdot 10^{-1}$	$1.512 \cdot 10^{2}$	44	34	12
10^{-2}	2	1/8	561	$1.441 \cdot 10^{-3}$	$7.055 \cdot 10^{-1}$	$4.894 \cdot 10^{2}$	49	41	–
10^{-2}	2	1/16	2145	$1.433 \cdot 10^{-3}$	$8.077 \cdot 10^{-1}$	$5.635 \cdot 10^{2}$	52	44	11
10^{-2}	2	1/32	8385	$1.431 \cdot 10^{-3}$	$8.612 \cdot 10^{-1}$	$6.017 \cdot 10^{2}$	56	46	15
10^{-2}	4	1/8	1105	$3.609 \cdot 10^{-4}$	$6.972 \cdot 10^{-1}$	$1.932 \cdot 10^{3}$	68	92	–
10^{-2}	4	1/16	4257	$3.588 \cdot 10^{-4}$	$7.538 \cdot 10^{-1}$	$2.101 \cdot 10^{3}$	72	101	11
10^{-2}	4	1/32	16705	$3.583 \cdot 10^{-4}$	$8.613 \cdot 10^{-1}$	$2.404 \cdot 10^{3}$	77	100	13
10^{-2}	8	1/8	2193	$9.026 \cdot 10^{-5}$	$6.451 \cdot 10^{-1}$	$7.147 \cdot 10^{3}$	102	371	–
10^{-2}	8	1/16	8481	$8.973 \cdot 10^{-5}$	$7.541 \cdot 10^{-1}$	$8.405 \cdot 10^{3}$	110	379	12
10^{-2}	8	1/32	33345	$8.960 \cdot 10^{-5}$	$8.613 \cdot 10^{-1}$	$9.613 \cdot 10^{3}$	117	354	14
10^{-2}	64	1/8	17425	$1.411 \cdot 10^{-6}$	$6.452 \cdot 10^{-1}$	$4.574 \cdot 10^{5}$	577	$> 10^{3}$	–
10^{-2}	64	1/16	67617	$1.402 \cdot 10^{-6}$	$8.491 \cdot 10^{-1}$	$6.055 \cdot 10^{5}$	625	$> 10^{3}$	21
10^{-2}	64	1/32	266305	$1.400 \cdot 10^{-6}$	$8.614 \cdot 10^{-1}$	$6.152 \cdot 10^{5}$	667	$> 10^{3}$	30

Table 4. Poiseuille flow 2D: timestep τ=10^{-2}. Preconditioning of S by M. Dependence of the condition number cond(S,M) on mesh-size h and geometry elongation $\mathcal{L}$. Convergence of mass-preconditioned CG method for the Schur complement system (CG), and of block LU (LU) and algebraic multigrid (MG) preconditioners for monolithic GMRES.

							iterations		
τ	$\mathcal{L}$	h	nodes	$\lambda_{\min}(S,M)$	$\lambda_{\max}(S,M)$	cond(S,M)	(CG)	(LU)	(MG)
10^{-4}	1	1/8	289	$1.494 \cdot 10^{-4}$	$3.600 \cdot 10^{-1}$	$2.409 \cdot 10^{3}$	117	189	–
10^{-4}	1	1/16	1089	$8.427 \cdot 10^{-5}$	$3.596 \cdot 10^{-1}$	$4.267 \cdot 10^{3}$	167	445	9
10^{-4}	1	1/32	4225	$6.698 \cdot 10^{-5}$	$4.192 \cdot 10^{-1}$	$6.258 \cdot 10^{3}$	195	358	15
10^{-4}	2	1/8	561	$3.815 \cdot 10^{-5}$	$3.600 \cdot 10^{-1}$	$9.438 \cdot 10^{3}$	180	448	–
10^{-4}	2	1/16	2145	$2.111 \cdot 10^{-5}$	$3.599 \cdot 10^{-1}$	$1.705 \cdot 10^{4}$	252	$> 10^{3}$	9
10^{-4}	2	1/32	8385	$1.675 \cdot 10^{-5}$	$4.192 \cdot 10^{-1}$	$2.503 \cdot 10^{4}$	291	$> 10^{3}$	17
10^{-4}	4	1/8	1105	$9.638 \cdot 10^{-6}$	$3.600 \cdot 10^{-1}$	$3.735 \cdot 10^{4}$	287	$> 10^{3}$	–
10^{-4}	4	1/16	4257	$5.283 \cdot 10^{-6}$	$3.600 \cdot 10^{-1}$	$6.814 \cdot 10^{4}$	397	$> 10^{3}$	9
10^{-4}	4	1/32	16705	$4.188 \cdot 10^{-6}$	$4.192 \cdot 10^{-1}$	$1.001 \cdot 10^{5}$	454	$> 10^{3}$	19
10^{-4}	8	1/8	2193	$2.422 \cdot 10^{-6}$	$3.600 \cdot 10^{-1}$	$1.486 \cdot 10^{5}$	447	$> 10^{3}$	–
10^{-4}	8	1/16	8481	$1.321 \cdot 10^{-6}$	$3.600 \cdot 10^{-1}$	$2.724 \cdot 10^{5}$	604	$> 10^{3}$	11
10^{-4}	8	1/32	33345	$1.047 \cdot 10^{-6}$	$4.192 \cdot 10^{-1}$	$4.004 \cdot 10^{5}$	695	$> 10^{3}$	21
10^{-4}	64	1/8	17425	$3.802 \cdot 10^{-8}$	$3.600 \cdot 10^{-1}$	$9.468 \cdot 10^{6}$	2559	$> 10^{3}$	–
10^{-4}	64	1/16	67617	$2.066 \cdot 10^{-8}$	$3.600 \cdot 10^{-1}$	$1.743 \cdot 10^{7}$	3474	$> 10^{3}$	12
10^{-4}	64	1/32	266305	$1.579 \cdot 10^{-8}$	$3.600 \cdot 10^{-1}$	$2.280 \cdot 10^{7}$	3772	$> 10^{3}$	25

Table 5. Poiseuille flow 2D: timestep τ=10^{-4}. Preconditioning of S by M. Dependence of the condition number cond(S,M) on mesh-size h and geometry elongation $\mathcal{L}$. Convergence of mass-preconditioned CG method for the Schur complement system (CG), and of block LU (LU) and algebraic multigrid (MG) preconditioners for monolithic GMRES.

11. Loghin, D. and Wathen, A.J.: Schur complement preconditioning for elliptic systems of partial differential equations, Numer. Linear. Algebra Appl. 10 (2003), 423–443.
12. Picasso, M.; Rappaz, J.: Stability of time-splitting schemes for the Stokes problem with stabilized finite elements, Numer. Methods Partial Differential Equations 17 (2001), no. 6, 632–656.
13. Powell, C. and Silvester, D.: Black-Box Preconditioning for Mixed Formulation of Self-Adjoint Elliptic PDEs, Challenges in scientific computing—CISC 2002, 268–285, Lect. Notes Comput. Sci. Eng., 35 Springer, Berlin, 2003.
14. Silvester, D.; Elman, H.; Kay, D.; Wathen, A.: Efficient preconditioning of the linearized Navier-Stokes equations for incompressible flow, J. Comput. Appl. Math. 128 (2001), 261–279.
15. Schöberl, J.; Zulehner, W.: On Schwarz-type Smoothers for Saddle Point Problems, Numer. Math. 95 (2003), 377–399.
16. Stüben, K.: A review of algebraic multigrid, Computational and Applied Mathematics 128 (2001), 281–309.
17. Vaněk, P.; Brezina, M.; Mandel, J.: Convergence of algebraic multigrid based on smoothed aggregation, Numer. Math. 88 (2001), no. 3, 559–579.
18. Verfürth, R.: Error estimates for a mixed finite element approximation of the Stokes problem, RAIRO Anal. Num. 18 (1984), 175–182.
19. Wabro, M.: Coupled algebraic multigrid methods for the Oseen problem, Comput. Vis. Sci. 7 (2004), 141–151.
20. Webster, R.: An algebraic multigrid solver for Navier-Stokes problems, Int. J. Numer. Methods Fluids 18 (1994), 761–780.
21. Wesseling, P.; Oosterlee, C.W.: Geometric multigrid with applications to computational fluid dynamics, J. Comput. Appl. Math. 128 (2001), 311-334.

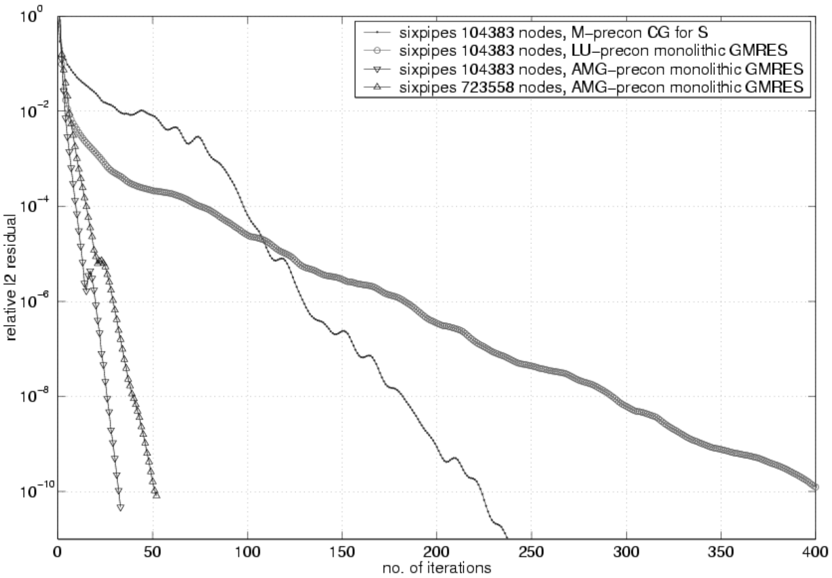

Fig. 5. `sixpipes` geometry: convergence of the mass-matrix preconditioned CG method for pressure Schur complement, block LU preconditioner for monolithic GMRES and algebraic multigrid preconditioner for monolithic GMRES for the original and the refined mesh

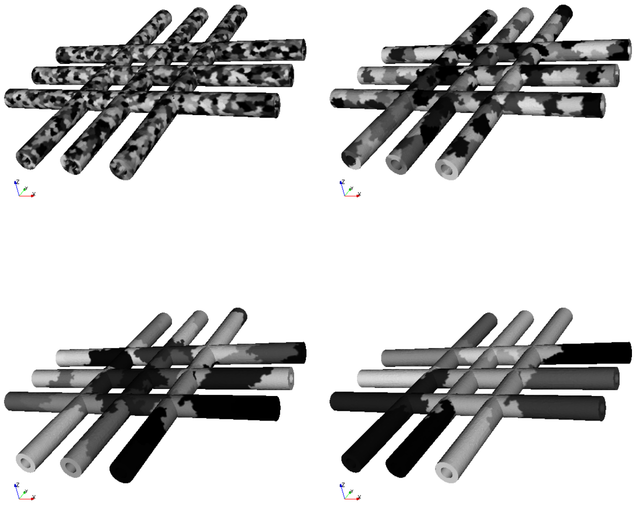

Fig. 6. `sixpipes` geometry: nodal aggregates of the first four multigrid levels

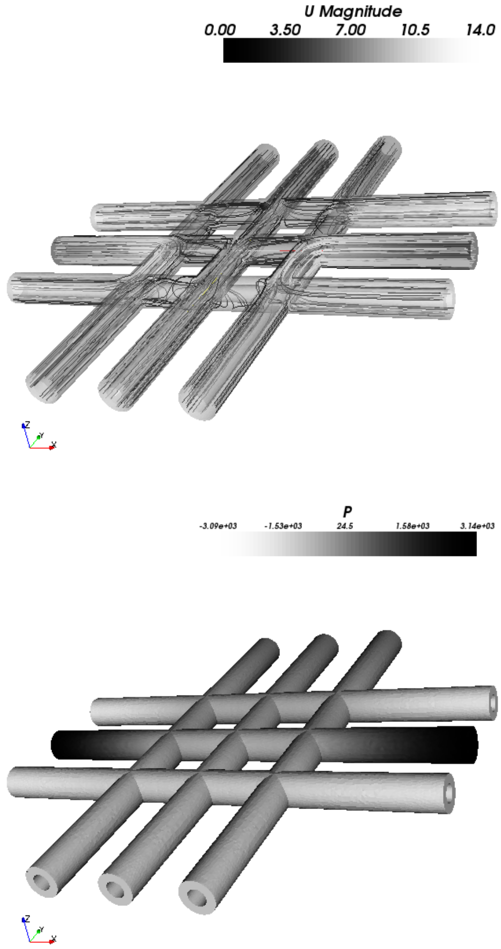

Fig. 4. `sixpipes` geometry: velocity (above) and pressure fields