Two Algorithms for Mixed Problems

R. Verfürth

Abstract: We describe two algorithms for the numerical solution of symmetric indefinite problems which arise e.g. from mixed finite element approximations of the Stokes equation. For the first algorithm we transform the original problem into an equation $Lp = g$ for the 'pressure' involving a symmetric, positive definite, continuous linear operator L. We apply a conjugate gradient algorithm to this equation. Each evaluation of L requires the solution of two discrete Poisson equations. This is done approximately using a multigrid algorithm. In the second algorithm the multigrid idea is directly applied to the indefinite problem. The main difficulty besides the indefiniteness is the lack of regularity of the solution of the corresponding continuous problem. This is overcome by introducing a mesh dependent norm. Both algorithms have convergence rates bounded away from 1 independently of the meshsize. Numerical results are presented for the first algorithm.

Dr. Rüdiger Verfürth
Institut für Mathematik
Ruhr-Universität Bochum
Postfach 1o2148
D-463O Bochum
West Germany

§ 1 Introduction

In this note we describe two algorithms for the numerical solution of indefinite problems of the form:

$$\begin{aligned} &\text{Find } (\underline{u}_h, p_h) \in X_h \times M_h \text{ such that} \\ &a(\underline{u}_h, \underline{v}_h) + b(\underline{v}_h, p_h) = \ell(v_h) \qquad \forall\, \underline{v}_h \in X_h \\ &b(\underline{u}_h, q_h) \qquad\qquad = 0 \qquad\qquad \forall\, q_h \in M_h. \end{aligned} \tag{1.1}$$

Here, X_h, M_h are finite dimensional Hilbert spaces, $a : X_h \times X_h \longrightarrow \mathbb{R}$ and $b : X_h \times M_h \longrightarrow \mathbb{R}$ continuous bilinear functionals and $\ell : X_h \longrightarrow \mathbb{R}$ a continuous linear functional. Moreover, a has to be X_h-elliptic and b has to satisfy an inf-sup condition (cf. (2.10) below). Specifically, we are interested in problems of this type which arise from the discretization of the Stokes problem

$$\begin{aligned} &-\Delta \underline{u} + \nabla p = \underline{f} \quad \text{in } \Omega, \qquad \underline{u} = 0 \quad \text{on } \partial\Omega, \\ &\operatorname{div} \underline{u} \qquad = 0 \quad \text{in } \Omega \end{aligned} \tag{1.2}$$

in a simply connected, bounded domain $\Omega \subset \mathbb{R}^d$, $d=2,3$.

For the first algorithm we transform (1.1) into an equation $Lp_h = g_h$ for the 'pressure' which involves a symmetric, positive definite, continuous linear operator $L : M_h \longrightarrow M_h$. We apply a conjugate gradient algorithm to this problem. Each evaluation of L requires the solution of two discrete Poisson equations. This is done approximately by applying 2 - 4 iterations of a multigrid algorithm. The resulting iterative process has a convergence rate κ bounded away from 1 independently of the meshsize. Numerical experiments yield values for κ between .8 and .93.

In the second algorithm we apply the multigrid idea directly to (1.1). The main difficulty besides the indefiniteness is the different order of the differential operators underlying a and b. This results in a

lack of regularity of the pressure. To overcome this difficulty we introduce a mesh dependent norm. The convergence rate measured in this norm is bounded away from 1 independently of the meshsize. In contrast to elliptic problems the use of a Lanczos algorithm as smoothing operator yields a substantial improvement on Jacobi relaxation.

§ 2 Preliminaries

Let $H^m(\Omega)$, $m \gg 0$, $H_o^1(\Omega)$ and $L^2(\Omega) := H^o(\Omega)$ be the usual Sobolev and Lebesgue spaces equipped with the norm

$$\|u\|_m := \{ \int_\Omega \sum_{|\beta| \leq m} |D^\beta u(x)|^2 \, dx \}^{1/2} .$$

We use the same notation for the corresponding product norm on $H^m(\Omega)^d$. Denote by $(f,g)_o$ the scalar product of $L^2(\Omega)^n$, $n=1,d,d^2$. For ease of notation put

$$X := H_o^1(\Omega)^d \quad , \quad M := \{p \in L^2(\Omega) : (p,1)_o = 0\} \tag{2.1}$$

and introduce the bilinear forms

$$a(\underline{u},\underline{v}) := (\nabla \underline{u}, \nabla \underline{v})_o \quad , \quad b(\underline{u},p) := (\operatorname{div} \underline{u}, p)_o \tag{2.2}$$

on XxX and XxM, resp.. Note, that

$$|\underline{u}|_1 := a(\underline{u},\underline{u})^{1/2} \tag{2.3}$$

is a norm on X equivalent to $\|.\|_1$ and that

$$|b(\underline{u},p)| \leq \sqrt{d} \, |\underline{u}|_1 \, \|p\|_o . \qquad \forall \, \underline{u} \in X \, , \, p \in M \tag{2.4}$$

Moreover, the inf-sup condition

$$\beta := \inf_{p \in M \setminus \{0\}} \sup_{\underline{u} \in X \setminus \{0\}} \frac{b(u,p)}{|\underline{u}|_1 \, \|p\|_o} > 0 \tag{2.5}$$

holds (cf. Theorem 3.7 in [9]).

Given $\underline{f} \in L^2(\Omega)^d$ the weak form of (1.2) is to find $(\underline{u},p) \in$ XxM such that

$$\begin{aligned} a(\underline{u},\underline{v}) + b(\underline{v},p) &= (\underline{f},\underline{v})_o && \forall \, \underline{v} \in X \\ b(\underline{u},q) &= 0 && \forall \, q \in M . \end{aligned} \tag{2.6}$$

It is well known [9] that (2.6) has a unique solution and that

$$\|\underline{u}\|_2 + |p|_1 \leq c_o \, \|\underline{f}\|_o \tag{2.7}$$

provided $\partial\Omega$ is sufficiently smooth or $\Omega \subset \mathbb{R}^2$ is a convex polygon [5,12].

Let $X_h \subset X$ and $M_h \subset M$, $h > 0$, be two families of finite dimensional spaces with $X_{2h} \subset X_h$ and $M_{2h} \subset M_h$, $h > 0$, which satisfy the usual approximation assumptions and inverse estimate:

$$\inf_{\underline{v}_h \in X_h} \|\underline{v} - \underline{v}_h\|_\alpha \leq c_1 \, h^{\beta-\alpha} \|\underline{v}\|_\beta \qquad \forall \, \underline{v} \in H^\beta(\Omega)^d, \; 0 \leq \alpha \leq 1 \leq \beta \leq 2, \tag{2.8a}$$

$$\inf_{p_h \in M_h} \|p - p_h\|_\alpha \leq c_2 \, h^{\beta-\alpha} \|p\|_\beta \qquad \forall \, p \in H^\beta(\Omega) \;, \; 0 \leq \alpha \leq \beta \leq 1, \tag{2.8b}$$

$$|\underline{v}_h|_1 \leq c_3 \, h^{-1} \, \|\underline{v}_h\|_0 \qquad \forall \, \underline{v}_h \in X_h \;. \tag{2.9}$$

$c_1, c_2, \ldots$ are generic constants which do not depend on h. The space X_h and M_h have to fit together such that

$$\inf_{p_h \in M_h \setminus \{0\}} \sup_{\underline{u}_h \in X_h \setminus \{0\}} \frac{b(\underline{u}_h, p_h)}{|\underline{u}_h|_1 \, \|p_h\|_0} \geq \gamma > 0 \tag{2.10}$$

holds with a constant γ independent of h. We equip $X_h \times M_h$ with the mesh dependent norm

$$|(\underline{u}_h, p_h)|_h := \{\|\underline{u}_h\|_0^2 + h^2 \, \|p_h\|_0^2\}^{1/2} \;. \tag{2.11}$$

Examples of spaces X_h, M_h satisfying the above assumptions are given in [15] for polygonal domains $\Omega \subset \mathbb{R}^2$. M_h is the space of piecewise linear, continuous functions on a regular triangulation $\mathcal{T}_h$ of Ω. X_h is either the space of piecewise linear, continuous functions on $\mathcal{T}_{h/2}$ or the space of piecewise quadratic, continuous functions on $\mathcal{T}_h$. Combining the ideas of [4] and [15] these results can be extended to other finite element spaces and regions $\Omega \subset \mathbb{R}^3$.

The approximation of X and M by X_h and M_h then leads to Problem (1.1). Because of (2.3), (2.4), (2.10) this problem always has a unique solution [9].

§ 3 A positive definite problem for the pressure

To simplify the notation we define the operators

$$B : X_h \longrightarrow M_h \quad , \quad B^* : M_h \longrightarrow X_h \quad ,$$
$$A^{-1} : X_h \longrightarrow X_h \quad , \quad J : L^2(\Omega)^d \longrightarrow X_h$$

as follows

$$(B\underline{u},p)_o = (\underline{u},B^*p)_o = b(\underline{u},p) \qquad \forall\, \underline{u}\in X_h \,,\ p\in M_h, \qquad (3.1)$$
$$a(A^{-1}\underline{u},\underline{v}) = (\underline{u},\underline{v})_o \qquad \forall\, \underline{u},\underline{v}\in X_h \,, \qquad (3.2)$$
$$(J\underline{f},\underline{v})_o = (\underline{f},\underline{v})_o \qquad \forall\, \underline{v}\in X_h \,,\ \underline{f}\in L^2(\Omega)^d. \qquad (3.3)$$

Note, that they never need to be computed explicitly. An easy calculation yields

$$\|B\underline{u}\|_o \leq \sqrt{d}\ |\underline{u}|_1 \qquad \forall\, \underline{u}\in X_h \,, \qquad (3.4)$$
$$\gamma\ \|p\|_o \leq \|B^*p\|_{-1} \leq \sqrt{d}\ \|p\|_o \qquad \forall\, p\in M_h \,, \qquad (3,5)$$
$$|A^{-1}\underline{u}|_1 = \|\underline{u}\|_{-1} \qquad \forall\, \underline{u}\in X_h \qquad (3.6)$$

where

$$\|\underline{u}\|_{-1} := \sup_{\underline{v}\in X_h\setminus\{0\}} \frac{(\underline{u},\underline{v})_o}{|\underline{v}|_1} \,.$$

Put

$$L := B\,A^{-1}\,B^* \quad , \quad g := B\,A^{-1}\,J\,\underline{f} \,. \qquad (3.7)$$

Lemma 3.1 : $L : M_h \longrightarrow M_h$ is a symmetric linear operator satisfying

$$(L\phi,\phi)_o \geq \gamma^2\ \|\phi\|_o^2 \qquad \forall\, \phi\in M_h \,, \qquad (3.8)$$
$$\|L\phi\|_o \leq d\ \|\phi\|_o \qquad \forall\, \phi\in M_h \,. \qquad (3.9)$$

The pair $(\underline{u}_h,p_h)\in X_h \times M_h$ is a solution of (1.1) if and only if $Lp_h = g$ and $\underline{u}_h = A^{-1}\,(J\underline{f} - B^*p_h)$.

Proof : The linearity of L is obvious. Equs. (3.4) - (3.6) together with

$$(L\phi,\psi)_o = a(A^{-1}B^*\phi, A^{-1}B^*\psi) \qquad \forall\, \phi,\psi\in M_h$$

imply the symmetry and (3.8),(3.9). The second part of the Lemma follows

from the definition of the operators and the unique solvability of (1.1). □

The operator L is not known explicitly. Each evaluation of Lp requires the computation of $A^{-1}\underline{w}$ for a suitable $\underline{w} \in X_h$ which is equivalent to the solution of two seperate discrete Poisson equations. This is done approximately by applying n steps of a multigrid (MG-) algorithm with zero as starting value to the problem

$$a(\underline{u},\underline{v}) = (\underline{w},\underline{v})_o \qquad \forall\, \underline{v} \in X_h \ .$$

Denote the resulting approximation to $A^{-1}\underline{w}$ by $K_n\underline{w}$. This defines a linear operator $K_n : X_h \longrightarrow X_h$ with

$$| A^{-1}\underline{w} - K_n\underline{w} |_1 \leq \kappa^n \, | A^{-1}\underline{w} |_1 \qquad \forall\, \underline{w} \in X_h \ , \ n \geq 1 \ , \qquad (3.10)$$

where κ is the convergence rate of the MG-algorithm.

κ is bounded away from 1 independently of h (cf. e.g. [7,11]). Numerical experiments often yield values $\kappa \leq .1$ (cf.[3,8,10]). The theoretical bound $\kappa \leq .205$ is derived in [6] for a MG-algorithm with checkered point Gauß-Seidel relaxation as smoothing operator on a convex polygon.

We assume that K_n is symmetric with respect to $(.,.)_o$. This holds for most MG-algorithm used in practice. Put

$$L_n := B\, K_n\, B^* \ . \qquad (3.11)$$

Lemma 3.2 : Assume that $\kappa^n < \gamma^2 d^{-1}$. Then L_n is a symmetric linear operator satisfying for each $\phi \in M_h$:

$$\| L_n\phi - L\phi \|_o \leq \kappa^n d \, \| \phi \|_o \ , \qquad (3.12)$$

$$(L_n\phi, \phi)_o \geq (\gamma^2 - d\kappa^n) \, \| \phi \|_o^2 \ , \qquad (3.13)$$

$$\| L_n\phi \|_o \leq d(1 + \kappa^n) \, \| \phi \|_o \ . \qquad (3.14)$$

Proof : The linearity and symmetry of L_n are obvious. Equs. (3.4) - (3.6) and (3.10) imply (3.12). Equs. (3.13), (3.14) immediately follow from (3.12) by using the triangle inequality. □

In the sequel we assume that $\kappa^n < \gamma^2 d^{-1}$. Since, in principle, γ can be computed explicitly, we could use the above cited bounds for κ to determine the required number n of MG-steps. However, these estimates are far too pessimistic. Our numerical results show that in general 2 - 4 MG-iterations are sufficient.

§ 4 A combined conjugate gradient - multigrid algorithm

The results of §3 show a possible way to solve Problem (1.1) approximately: We compute an approximation g^* to $g := B A^{-1} J \underline{f}$ using the operator K_n and apply an iteration process to the equation $L_n p = g^*$ which only requires the evaluation of L_n. Since L_n is a symmetric, positive definite operator and approximates an operator with the same properties, an appropriate iteration process is the conjugate gradient algorithm.

Algorithm 4.1 :

0. Preprocessing phase : Compute $g^* := B K_n J \underline{f}$.

1. Start : Given an initial guess $p^0 \in M_h$ for the pressure p_h solving (1.1). Compute $q^0 := L_n p^0$ and put $r^0 := q^0 - g^*$, $d^0 := -r^0$. Set $i := 0$.

2. Iteration Step : If $\|r^i\|_0 \leq \varepsilon$ go to Step 3. Otherwise compute $q^{i+1} := L_n d^i$ and put

$$\alpha^{i+1} := - \frac{(r^i, d^i)_0}{(d^i, q^{i+1})_0} ,$$

$$p^{i+1} := p^i + \alpha^{i+1} d^i , \quad r^{i+1} := r^i + \alpha^{i+1} q^{i+1} ,$$

$$\beta^{i+1} := \frac{(r^{i+1}, r^{i+1})_0}{(r^i, r^i)_0} ,$$

$$d^{i+1} := -r^{i+1} + \beta^{i+1} d^{i+1} .$$

Replace i by i+1 and return to the beginning of Step 2.

3. Pastprocessing phase : Compute $\underline{u}^i := K_n (J \underline{f} - B^* p^i)$ and use $(\underline{u}^i, p^i) \in X_h \times M_h$ as final approximation to the solution of (1.1). □

From Lemma 3.2 and Exercise 10 of §8.8 in [13] we obtain

$$\|r^i\|_0 \leq \frac{2}{\delta} \left(\frac{1-\sqrt{\delta}}{1+\sqrt{\delta}} \right)^i \|r^0\|_0 \qquad (4.1)$$

where

$$\delta = \frac{\gamma^2 - d\kappa^n}{d(1+\kappa^n)} \leq \frac{\gamma^2}{d} \quad . \qquad (4.2)$$

Note that δ is independent of h. Equs. (4.1), (4.2) imply that the mean convergence rate of Algorithm 4.1 is of the form $1 - O(\gamma)$.

The following error estimate is proved in [16].

Proposition 4.2 : Let $(\underline{u}^i, p^i)$ be the last iterate of Algorithm 4.1 and $(\underline{u}_h, p_h)$ be the solution of (1.1). Then we have

$$\|\underline{u}_h - \underline{u}^i\|_1 + \|p_h - p^i\|_0$$

$$\leq \frac{1}{\gamma^2 - d\kappa^n} \{ 5\varepsilon + 8\kappa^n \|\underline{f}\|_{-1} + 15\kappa^n \|p_h\|_0 \} \quad . \qquad (4.3) \quad \square$$

The proof of Proposition 4.2 only exploits Lemma 3.1, 3.2 and the stopping criterion $\|r^i\|_0 \leq \varepsilon$. Hence it also holds for other iteration processes. The number κ^n in (4.3) is the relative accuracy with which g^*, the last residue r^i and the final approximation $\underline{u}^i$ for the velocity are computed. Hence, Steps 1 and 2 of Algorithm 4.1 need only be performed with a moderate accuracy. Once the residue is smaller than ε_1 we may switch to a higher accuracy in the solution of Poisson's equation. This improves the efficiency of Algorithm 4.1 substantially. Finally, we note that we can use any Poisson solver satisfying (3.10).

§ 5 A multilevel algorithm

In this § we assume that the regularity assumption (2.7) holds. As usual for multilevel algorithms we have a sequence of meshsizes $h_0 > h_1 > \ldots > h_R$ with $h_{k-1} = 2\,h_k$, $1 \leqslant k \leqslant R$. Actually we want to solve Problem (1.1) on level R. If no ambiguity can arise, we replace subscripts h_k by k. Instead of (1.1) we have to consider the slightly more general problem

$$a(\underline{u}_k,\underline{v}) + b(\underline{v},p_k) + b(\underline{u}_k,q) = G_k(\underline{v},q) \qquad \forall\ (\underline{v},q) \in X_k \times M_k \ . \quad (5.1)$$

G_k is a linear functional on $X_k \times M_k$ and on the finest grid:

$$G_R(\underline{v},q) = (\underline{f},\underline{v})_0 \qquad \forall\ (\underline{v},q) \in X_k \times M_k \ . \quad (5.2)$$

Algorithm 5.1 : *(One iteration step at level k , $1 \leqslant k \leqslant R$, with m smoothing steps)*

1. Smoothing : Let $(\underline{u}_k^0, p_k^0) \in X_k \times M_k$ be a given approximation to the solution of Problem (5.1). For $\ell = 1,2,\ldots,m$ compute the solutions of

$$(\underline{w}_k^\ell,\underline{v})_0 + h_k^2\,(r_k^\ell,q)_0 = \omega_k^{-2}\,\{\, G_k(\underline{v},q) - a(\underline{u}_k^{\ell-1},\underline{v}) - b(\underline{v},p_k^{\ell-1}) - b(\underline{u}_k^{\ell-1},q)\,\} \qquad \forall\ (\underline{v},q) \in X_k \times M_k$$

and

$$(\underline{u}_k^\ell - \underline{u}_k^{\ell-1},\underline{v})_0 + h_k^2\,(p_k^\ell - p_k^{\ell-1},q)_0 = a(\underline{w}_k^\ell,\underline{v}) + b(\underline{v},r_k^\ell) + b(\underline{w}_k^\ell,q) \qquad \forall\ (\underline{v},q) \in X_k \times M_k \ .$$

2. Correction : Let $(\underline{u}_{k-1}^*, p_{k-1}^*) \in X_{k-1} \times M_{k-1}$ be the solution of Problem (5.1) with

$$G_{k-1}(\underline{v},q) := G_k(\underline{v},q) - a(\underline{u}_k^m,\underline{v}) - b(\underline{v},p_k^m) - b(\underline{u}_k^m,\underline{v}) \qquad \forall\ (\underline{v},q) \in X_{k-1} \times M_{k-1} \ .$$

If $k = 1$, put $(\tilde{\underline{u}}_{k-1}, \tilde{p}_{k-1}) = (\overset{*}{\underline{u}}_{k-1}, \overset{*}{p}_{k-1})$.

If $k > 1$, compute an approximation $(\tilde{\underline{u}}_{k-1}, \tilde{p}_{k-1})$ to $(\overset{*}{\underline{u}}_{k-1}, \overset{*}{p}_{k-1})$ by applying μ, $\mu \geq 2$, iterations of the $(k-1)$-level scheme to (5.1) with starting value zero.

Put

$$(\underline{u}_k^{m+1}, p_k^{m+1}) := (\underline{u}_k^m + \tilde{\underline{u}}_{k-1}, p_k^m + \tilde{p}_{k-1}) \quad . \qquad \square$$

If we introduce a basis for $X_k \times M_k$, Problem (5.1) can be written in matrix-vector notation as $A_k x_k = d_k$ with a symmetric, indefinite matrix A_k. In the smoothing part of Algorithm 5.1 m Jacobi relaxation steps are applied to the squared system $A_k^2 x_k = A_k d_k$. The relaxation parameter ω_k has to be greater than the spectral radius of A_k.

Let $\delta_{k,m}$ be the convergence rate of one iteration of Algorithm 5.1 at level k with m smoothing steps measured in the $|.|_{h_k}$ - norm. Denote by $(\overset{*}{\underline{u}}_k, \overset{*}{p}_k) \in X_k \times M_k$ the solution of Problem (5.1) and by $(\underline{e}_k^\ell, \varepsilon_k^\ell) := (\overset{*}{\underline{u}}_k - \underline{u}_k^\ell, \overset{*}{p}_k - p_k^\ell)$ the error of the ℓ-th iterate. Put $N_k := \dim X_k \times M_k$. There is a complete set of eigenfunctions $(\underline{\phi}_k^j, \psi_k^j)$, $1 \leq j \leq N_k$, defined by

$$a(\underline{\phi}_k^j, \underline{v}) + b(\underline{v}, \psi_k^j) + b(\underline{\phi}_k^j, q)$$

$$= \lambda_j \{ (\underline{\phi}_k^j, \underline{v})_0 + h_k^2 \, (\psi_k^j, q)_0 \} \qquad \forall \ (\underline{v}, q) \in X_k \times M_k \ , \qquad (5.4a)$$

$$(\underline{\phi}_k^i, \underline{\phi}_k^j)_0 + h_k^2 \, (\psi_k^i, \psi_k^j)_0 = \delta_{ij} \qquad 1 \leq i, j \leq N_k \ . \qquad (5.4b)$$

Because of (2.10) the eigenvalues can be arranged such that

$$0 < |\lambda_1| \leq \ldots \leq |\lambda_{N_k}| =: \Lambda_k \ . \qquad (5.5)$$

An easy calculation [2,17] yields

$$\Lambda_k \leq c_4 \, h_k^{-2} \quad . \qquad (5.6)$$

The following analysis holds, if we have $\Lambda_k \leq \omega_k$ and $\omega_k = O(h_k^{-2})$. To simplify the notation, we assume that $\omega_k = \Lambda_k$.

We define a scale of norms $\|\!|\!| . |\!|\!|_s$, $s \in \mathbb{R}$, on $X_k \times M_k$ as follows. Let c_j , $1 \leq j \leq N_k$, be the coefficients of $(\underline{u}_k, p_k) \in X_k \times M_k$ with respect to the basis $(\underline{\phi}_k^j, \psi_k^j)$, $1 \leq j \leq N_k$, then

$$\|\!|\!| (\underline{u}_k, p_k) |\!|\!|_s := \{ \sum_{j=1}^{N_k} |\lambda_j|^s c_j^2 \}^{1/2} . \tag{5.7}$$

Because of (5.4) we have

$$\|\!|\!| (\underline{u}_k, p_k) |\!|\!|_o = | (\underline{u}_k, p_k) |_{h_k} . \tag{5.8}$$

It is easy to prove (cf. [2,17]) the *smoothing property*

$$\begin{aligned} & \|\!|\!| (\underline{e}_k^m, \varepsilon_k^m) |\!|\!|_2 \\ & \leq \Lambda_k \max_{-1 \leq x \leq 1} |x(1-x^2)^m| \, \|\!|\!| (\underline{e}_k^o, \varepsilon_k^o) |\!|\!|_o \\ & \leq c_5 \, h_k^{-2} \frac{1}{\sqrt{2m+1}} | (\underline{e}_k^o, \varepsilon_k^o) |_k . \end{aligned} \tag{5.9}$$

The crucial point is the *approximation property*

$$| (\underline{e}_k^m - \underline{u}_{k-1}^*, \varepsilon_k^m - p_{k-1}^*) |_k \leq c_6 \, h_k^2 \, \|\!|\!| (\underline{e}_k^m, \varepsilon_k^m) |\!|\!|_2 . \tag{5.10}$$

For the proof of (5.10) we refer to [17]. It's general structure is similar to that of Bank's convergence analysis [1]. However, his regularity assumptions are not met by the Stokes problem. Our choice of the norm $|.|_{h_k}$ reflects this loss of regularity. Moreover, we have to estimate the velocity and pressure components seperately.and to use additional duality arguments.

Equs. (5.9), (5.10) imply

$$\delta_{1,m} \leq \frac{c_7}{\sqrt{2m+1}} . \tag{5.11}$$

Finally, we conclude

$$\begin{aligned} & | (\underline{e}_k^{m+1}, \varepsilon_k^{m+1}) |_k \\ & \leq | (\underline{e}_k^m - \underline{u}_{k-1}^*, \varepsilon_k^m - p_{k-1}^*) |_k + | (\underline{u}_{k-1}^* - \tilde{\underline{u}}_{k-1}, p_{k-1}^* - \tilde{p}_{k-1}) |_k \end{aligned}$$

$$\leq \{ \frac{c_7}{\sqrt{2m+1}} + 2\, \delta_{k-1,m}^{\mu} \,(1 + \frac{c_7}{\sqrt{2m+1}})\}\, |(\underline{e}_k^o, \varepsilon_k^o)|_k \quad .$$

Hence

$$\delta_{k,m} \leq \frac{c_7}{\sqrt{2m+1}} + 2\, \delta_{k-1,m}^{\mu} \,(1 + \frac{c_7}{\sqrt{2m+1}}) \quad . \tag{5.12}$$

By induction this proves:

Proposition 5.2 : Let $m \geq 8\, c_7^2\, 25^{1/\mu}$. Then the convergence rate of Algorithm 5.1 is bounded by

$$\delta_{k,m} \leq \frac{2c_7}{\sqrt{2m+1}} \leq \frac{1}{2} \qquad \forall\, k \in \mathbb{N} \quad . \tag{5.13}$$ □

Instead of Jacobi relaxation we could perform m steps of a conjugate residual (CR-) algorithm (cf. [14]) in the smoothing part of Algorithm 5.1. Assume that $m \geq 2$ is even. Instead of (5.9) we then have (cf. [17]):

$$\begin{aligned} &|||\, (\underline{e}_k^m, \varepsilon_k^m)\, |||_2 \\ &\leq \Lambda_k \min_{p \in \tilde{\Pi}_m} \max_{-1 \leq x \leq 1} |x\, p(x)|\; |(\underline{e}_k^o, \varepsilon_k^o)|_k \\ &\leq c_8\, \frac{1}{m+1}\, |(\underline{e}_k^o, \varepsilon_k^o)|_k \quad . \end{aligned} \tag{5.14}$$

Here, $\tilde{\Pi}_m$ is the space of polynomials of degree $\leq m$ with constant coefficient 1. Since (5.10) is independent of the smoothing procedure, we get for $m \geq 4\, c_9\, 5^{1/\mu}$ and $k \in \mathbb{N}$

$$\delta_{1,m} \leq \frac{c_9}{m+1} \quad , \quad \delta_{k,m} \leq \frac{2c_9}{m+1} \quad . \tag{5.15}$$

This is in contrast to elliptic problems where the use of conjugate gradient type smoothing procedures yields no substantial improvement on Jacobi relaxation.

§ 6 Numerical results for Algorithm 4.1

We consider three different regions $\Omega \subset \mathbb{R}^2$

(i) the unit square $\Omega_c := (0,1)x(0,1)$,

(ii) the L-shaped region $\Omega_L := \Omega_c \setminus (0.5,1)x(0.5,1)$,

(iii) the slit unit square $\Omega_S := \Omega_c \setminus (0.5,1)x\{0.5\}$

and the following right hand sides $\underline{f}$:

example 1 : $\underline{f}^{(1)}(x,y) := \underline{e} := (1,-1)^T$,

example 2 : $\underline{f}^{(2)}(x,y) := 100\, x(1-x)\, y(1-y)\, \underline{e}$,

example 3 : $\underline{f}^{(3)}(x,y) := 100 \exp(-100(x^2+y^2))\, \underline{e}$.

We use Courant's triangulation with isosceles, rectangular triangles with short sides of length h and put $X_h := S_h^2 \cap X$, $M_h := S_{2h} \cap M$. Here, S_h is the space of continuous, piecewise linear functions on the triangulation with meshsize h.

The Poisson equations are solved with the MG-routine HELMH of W. Hackbusch [10]. We choose n between 2 and 4 and $\varepsilon = 10^{-3}$. The MG iterations terminate, if the L^2 - norm of the difference of the last two iterates is smaller than 10^{-7}. According to §4 we compute g^*, the last residue r^i and the final approximation $\underline{u}^i$ for the velocity more accurately by replacing n by 4n. In contrast to the theoretical analysis we store the last iterate of the MG-algorithm and use it as starting value for the next call of the MG-routine. This reduces the total number of MG iterations.

We usually choose $p^o = 0$ as starting value for Algorithm 4.1. If we have already computed an approximation to p_{2h} on the grid Ω_{2h}, we take its linear interpolant as starting value on the grid Ω_h. We use the meshsizes $h = 2^{-k}$, $k = 2,3,4,5$. All computations were done in single

precission arithmetic on the Control Data 175 in Bochum.

Let p^i be the last iterate of Algorithm 4.1. Then

$$\kappa := \{ \|r^i\|_o / \|r^o\|_o \}^{1/i}$$

is a measure for the mean convergence rate of Algorithm 4.1. In Table 1 we have listed κ for the different examples. A hyphen indicates that $\|r^o\|_o \leq \varepsilon$. For further numerical results we refer to [16].

	example 1			example 2			example 3		
1/h	Ω_C	Ω_L	Ω_S	Ω_C	Ω_L	Ω_S	Ω_C	Ω_L	Ω_S
4	.761	.622	.642	.750	.715	.756	-	-	-
8	.507	.635	.583	.800	.801	.636	.841	.763	.815
16	-	.781	.716	.838	.827	.802	.921	.886	.892
32	.709	.816	.840	.842	.890	.800	.920	.833	.859

Table 1 : Mean convergence rate κ of Algorithm 4.1

References :

1 R.E. Bank: A comparison of two multilevel iterative methods for non-symmetric and indefinite elliptic finite element equations. SIAM J. Numer. Anal. 18, 724-743 (1981)

2 R.E. Bank, T.Dupont: An optimal order process for solving finite element equations. Math. Comp. 36, 35-51 (1981)

3 R.E. Bank, A.H. Sherman: An adaptive multi-level method for elliptic boundary value problems. Computing 26, 91-105 (1981)

4 M. Bercovier, O. Pironneau: Error estimates for finite element method solution of the Stokes problem in the primitive variables. Numer. Math. 33, 211-224 (1979)

5 C. Bernardi, G. Raugel: Méthodes d'éléments finis mixtes pour les équations de Stokes et de Navier-Stokes dans un polygone non convexe. Calcolo 18, 255-291 (1981)

6 D. Braess: The convergence rate of a multigrid method with Gauss-Seidel relaxation for the Poisson equation. In: Multigrid Methods (W. Hackbusch, U. Trottenberg; eds.), pp. 368-386, Berlin-Heidelberg-New York, Springer 1982

7 D. Braess, W. Hackbusch: A new convergence proof for the multigrid method including the V-cycle. Preprint 1982, Universität Bochum

8 A. Brandt: Multi-level adaptive solution to boundary value problems. Math. Comp. 31, 333-391 (1977)

9 V. Girault, P.A. Raviart: Finite element approximation of the Navier-Stokes equations. Berlin-Heidelberg-New York, Springer 1979

10 W. Hackbusch: Ein iteratives Verfahren zur schnellen Auflösung elliptischer Randwertprobleme. Report 76-12, Universität Köln 1976

11 W. Hackbusch: Multi-grid convergence theory. In: Multigrid Methods (W. Hackbusch, U. Trottenberg; eds.), pp. 177-219, Berlin-Heidelberg-New York, Springer 1982

12 O. Ladyzenskaya: The mathematical theory of viscous incompressible flows. London, Gordon & Breach 1963

13 D.G. Luenberger: Introduction to linear and nonlinear programming. Addison-Wesley 1973

14 J. Stoer: Solution of large linear systems of equations by conjugate gradient type methods. Preprint

15 R. Verfürth: Error estimates for a mixed finite element approximation of the Stokes equations. RAIRO (to appear)

16 R. Verfürth: A combined conjugate gradient - multigrid algorithm for the numerical solution of the Stokes problem. (submitted)

17 R. Verfürth: A multilevel algorithm for mixed problems. (submitted)