Mathematical Modelling and Analysis
Volume 6 Number 1, 2001, pages 7–27

SOLVING THE STOKES PROBLEM ON A MASSIVELY PARALLEL COMPUTER

O. AXELSSON[1], V. A. BARKER[2], M. NEYTCHEVA[3] and B. POLMAN[4]

[1,3,4] *Department of Mathematics, University of Nijmegen*
Toernooiveld 1, 6525 ED Nijmegen, The Netherlands
E-mail: axelsson@sci.kun.nl, neytchev@sci.kun.nl
E-mail: polman@sci.kun.nl

[2] *Institute for Numerical Analysis, The Technical University of Denmark*
Building 305, DK-2600 Lyngby, Denmark
E-mail: vab@imm.dtu.dk

Received June 7, 2000

ABSTRACT

We describe a numerical procedure for solving the stationary two-dimensional Stokes problem based on piecewise linear finite element approximations for both velocity and pressure, a regularization technique for stability, and a defect-correction technique for improving accuracy. Eliminating the velocity unknowns from the algebraic system yields a symmetric positive semidefinite system for pressure which is solved by an inner-outer iteration. The outer iterations consist of the unpreconditioned conjugate gradient method. The inner iterations, each of which corresponds to solving an elliptic boundary value problem for each velocity component, are solved by the conjugate gradient method with a preconditioning based on the algebraic multi-level iteration (AMLI) technique. The velocity is found from the computed pressure. The method is optimal in the sense that the computational work is proportional to the number of unknowns. Further, it is designed to exploit a massively parallel computer with distributed memory architecture. Numerical experiments on a Cray T3E computer illustrate the parallel performance of the method.

1. INTRODUCTION

The Stokes equations model the flow of a slow viscous incompressible fluid as well as that of an isotropic incompressible elastic material. They are the linear part of the Navier-Stokes equations, and since many of the problems of solving the full Navier-Stokes equations are present when solving the Stokes equations, the latter have been an intensive topic of research. For problems in fluid

dynamics it is not at all exceptional to solve two or three-dimensional problems with perhaps millions of unknowns. Thus, large scale scientific computations are involved, and it is natural to try to exploit massively parallel computers.

This study concerns a parallelizable method for solving the stationary case of the two-dimensional Stokes equations. This is the problem of finding a vector function $\boldsymbol{u}(x,y) = [u_1(x,y), u_2(x,y)]$ and a scalar function $p(x,y)$ that satisfy the system

$$\begin{aligned} -\Delta \boldsymbol{u} + \nabla p &= \boldsymbol{f} \quad \text{in } \Omega, \\ \operatorname{div} \boldsymbol{u} &= 0 \quad \text{in } \Omega, \\ \boldsymbol{u} &= \boldsymbol{g} \quad \text{on } \partial\Omega, \end{aligned} \tag{1.1}$$

where $\Omega \subset \mathbb{R}^2$ is a bounded domain with boundary $\partial\Omega$. The function $\boldsymbol{u}$ is the velocity of the fluid and p is the kinematic pressure.

The finite element treatment of the Stokes problem (1.1) leads to a linear algebraic system which can be ordered so as to assume the form

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{U}_h \\ \mathbf{p}_h \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \mathbf{G} \end{bmatrix}. \tag{1.2}$$

The vectors $\mathbf{U}_h$ and $\mathbf{p}_h$ are the discrete velocity and pressure, respectively. The coefficient matrix is symmetric and indefinite. Further, because of the Dirichlet boundary conditions for $\boldsymbol{u}$, the matrix block $\mathbf{A}$ is positive definite.

A standard approach to discretize the Stokes problem is to apply mixed finite element methods. It is well known, however, that the spaces used for $\boldsymbol{u}$ and p cannot be chosen independently. More precisely, they must satisfy the *inf-sup*, or Ladyženskaya-Babuška-Brezzi (LBB), condition (Section 2). Typically, stable element pairs consist of piecewise polynomials of different degrees for velocity and pressure. Pairs of finite elements that satisfy the LBB condition are referred to as *LBB–stable* or *compatible*. There are a number of mixed elements that are known to satisfy the LBB condition (cf. [1], [9], [15], [21], for example). One of the simplest is the so-called Mini-element (P1-bubble/P1), where the velocity is approximated by the standard piecewise linear basis functions, enriched by a 'bubble' function on each triangle, and the pressure is approximated by piecewise linear basis functions with no enrichment. Generally it is the pressure variable that is most sensitive to instability. In particular, unstable element pairs can give rise to unwanted 'spurious' pressure modes, illustrated for example in [1], [15], [13].

The restraint imposed by the LBB condition is unfortunate in so far as there are significant practical benefits in choosing the same finite element space for both velocity and pressure, an approach hereafter referred to as 'equal–order approximation'. The advantage of equal-order approximation is especially strong in the context of parallel computation. Accumulated experience with massively parallel machines indicates that efficient solution methods on typical parallel architectures favor simplicity and homogeneous treatment

of all components involved in the solution process because this unifies the data structures and access to them. Fortunately, it was discovered that the LBB condition can be 'evaded' by regularization techniques. These allow equal–order approximation at the expense of introducing a mesh–dependent perturbation in (1.2). More specifically, (1.2) is replaced by a system of the form

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & -\sigma\mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{U}_h \\ \mathbf{p}_h \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \mathbf{G} + \sigma\widetilde{\mathbf{G}} \end{bmatrix}, \tag{1.3}$$

where σ is a regularization parameter and $\mathbf{C}$ and $\widetilde{\mathbf{G}}$ are a properly chosen matrix and vector, respectively. Actually, systems of this type arise from certain mixed element methods via the process of static condensation, an example being the Mini-element. However, the generality of (1.3) opens the door to new discretization methods. For a discussion of the regularization approach see, for example, [2], [6], [8], [10], [14], [17], [18].

This paper describes a solution procedure for the Stokes problem based on standard piecewise linear approximation for both variables combined with a consistent regularization technique for stability and a defect-correction process for improving the accuracy of the solution (especially the pressure). The velocity unknowns are eliminated from (1.3) to obtain a system for pressure alone. This system is then solved by inner-outer iterations, each based on the conjugate gradient method. At the heart of this process is an inner-iteration preconditioner consisting of an algebraic multilevel iteration (AMLI) solver which can be very effectively implemented on parallel architectures. The remainder of the paper is organized as follows: Section 2 reviews some of the basic theory of the Stokes problem and its classical finite element treatment. Section 3 describes our finite element method, gives an analysis of the discretization errors associated with it, and presents a defect-correction technique for improving accuracy. Section 4 discusses the algebraic aspects of the problem, including the inner-outer iterations technique referred to above. Section 5 presents the results of some numerical experiments on a Cray T3E MPP computer, and Section 6 states our conclusions.

2. BACKGROUND

In this section we review some of the basic theory of the Stokes problem and its solution by mixed finite element methods. Standard notations are used throughout the paper, for details see [3].

Let Ω be a bounded and connected domain of $\mathbb{R}^2$ with a Lipschitz continuous boundary $\partial\Omega$. Let the Stokes data $\boldsymbol{f}$ and $\boldsymbol{g}$ in (1.1) be functions in $\left(L^2(\Omega)\right)^2$ and $\left(H^{1/2}(\partial\Omega)\right)^2$, respectively, where $\boldsymbol{g}$ satisfies the compatibility condition

$$\int_{\partial\Omega} \boldsymbol{g} \cdot \boldsymbol{n}\, dS = 0. \tag{2.1}$$

The following weak formulation of the Stokes problem (1.1) can be derived:

Find $\boldsymbol{u} \in \left(H^1(\Omega)\right)^2$ and $p \in L^2(\Omega)$ such that

$$\begin{aligned} (\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) - (p, \operatorname{div} \boldsymbol{v}) &= (\boldsymbol{f}, \boldsymbol{v}) \qquad \forall \boldsymbol{v} \in \overset{o}{\mathcal{V}}(\Omega), \qquad (2.2) \\ \boldsymbol{u} &= \boldsymbol{g} \qquad\qquad\qquad \text{on } \partial\Omega, \end{aligned}$$

where $\overset{o}{\mathcal{V}}(\Omega) = \left\{ \boldsymbol{v} \in \left(H_0^1(\Omega)\right)^2 : \operatorname{div} \boldsymbol{v} = \boldsymbol{0} \right\}$ and where the inhomogeneous boundary condition is to be understood in the sense of traces. Problem (2.2) has a unique solution. (See [9], for example).

We will now derive an alternative variational formulation of the Stokes problem which includes pressure. To do this we need the following theorem:

Theorem 2.1. *Let L be a continuous linear functional on $\boldsymbol{V}(\Omega)$. Then $L(\boldsymbol{v}) = 0$ for all $\boldsymbol{v} \in \overset{o}{\mathcal{V}}(\Omega)$ if and only if there exists $p \in L^2(\Omega)$ such that $L(\boldsymbol{v}) = (p, \operatorname{div} \boldsymbol{v})$ for all $\boldsymbol{v} \in \boldsymbol{V}(\Omega)$. Further, if p exists then it is unique within an additive constant.*

(See [9]). Hereafter, p will denote the unique pressure function in $L_0^2(\Omega)$.

Taking as L the linear functional

$$L(\boldsymbol{v}) = (\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) - (\boldsymbol{f}, \boldsymbol{v}), \quad \boldsymbol{v} \in \boldsymbol{V}(\Omega)$$

where $\boldsymbol{u}$ is the solution of problem (2.2) we can deduce from the preceding theorem that there exists a function $p \in L_0^2(\Omega)$, such that

$$(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) - (p, \operatorname{div} \boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v}) \quad \forall \boldsymbol{v} \in \boldsymbol{V}(\Omega).$$

This establishes that $\boldsymbol{u}$ and p are a solution of the following variational problem:

Find $\boldsymbol{u} \in \left(H^1(\Omega)\right)^2$ and $p \in L_0^2(\Omega)$ such that

$$\begin{aligned} a(\boldsymbol{u}, \boldsymbol{v}) - b(p, \boldsymbol{v}) &= (\boldsymbol{f}, \boldsymbol{v}) \qquad\qquad \forall \boldsymbol{v} \in \boldsymbol{V}(\Omega), \qquad (2.3) \\ b(q, \boldsymbol{u}) &= 0 \qquad\quad \forall q \in H^1(\Omega) \cap L_0^2(\Omega), \qquad (2.4) \\ \boldsymbol{u} &= \boldsymbol{g} \qquad\qquad\qquad \text{on } \partial\Omega, \qquad (2.5) \end{aligned}$$

where $a(\boldsymbol{u}, \boldsymbol{v}) = (\nabla \boldsymbol{u}, \nabla \boldsymbol{v})$, $\quad b(q, \boldsymbol{v}) = (q, \operatorname{div} \boldsymbol{v})$.

Theorem 2.2. *The solution of problem* (2.3)-(2.5) *is unique.*

(See [9], for example).

When $\boldsymbol{u}$ and p are sufficiently smooth then they are the solution of the original formulation of the Stokes problem (1.1). It will be noted that problem (2.3)-(2.5), in contrast to problem (2.2), is not expressed in terms of the divergence-free space $\overset{o}{\mathcal{V}}(\Omega)$. This is a great advantage for discretizations based on the mixed finite element method, because the construction of finite element subspaces of $\overset{o}{\mathcal{V}}(\Omega)$ is far from straightforward.

The existence and uniqueness of the solution of problem (2.3)-(2.5) are intimately connected with two basic properties of the Stokes problem, namely:

(i) The bilinear form $a(\cdot,\cdot)$ is $\boldsymbol{V}$-elliptic on $\boldsymbol{V}(\Omega)$; i.e., there exists a positive constant α such that

$$a(\boldsymbol{v},\boldsymbol{v}) \geq \alpha \|\boldsymbol{v}\|^2_{\boldsymbol{V}} \quad \forall \boldsymbol{v} \in \boldsymbol{V}(\Omega). \tag{2.6}$$

(ii) The bilinear form $b(\cdot,\cdot)$ satisfies the *inf-sup* condition; i.e., there exists a positive constant β such that

$$\sup_{\boldsymbol{v}\in\boldsymbol{V}(\Omega)} \frac{(q,\operatorname{div}\boldsymbol{v})}{\|\boldsymbol{v}\|_{\boldsymbol{V}}} \geq \beta \|q\|_0 \quad \forall q \in L^2_0(\Omega). \tag{2.7}$$

We consider now the discretization of the Stokes problem by the mixed finite element method, assuming for simplicity that Ω is a polygon and $\boldsymbol{g} = \boldsymbol{0}$ on $\partial\Omega$. Let $\mathcal{T}_h = \{T_k\}_{k=1}^{N_e}$ be a triangulation of Ω, where h is the maximum element edge length. Let $\boldsymbol{V}^h$ and P^h be finite element subspaces of $\boldsymbol{V}(\Omega)$ and $L^2_0(\Omega)$ based on $\mathcal{T}_h$ and containing the complete polynomials of degree k and l, respectively. Then the corresponding discrete analog of problem (2.3)-(2.5) is the following:

$$\begin{aligned} &\text{Find } \boldsymbol{u}_h \in \boldsymbol{V}^h \text{ and } p \in P^h \text{ such that} \\ &a(\boldsymbol{u}_h,\boldsymbol{v}_h) - b(p_h,\boldsymbol{v}_h) = (\boldsymbol{f},\boldsymbol{v}_h) \qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}^h, \\ &b(q_h,\boldsymbol{u}_h) = 0 \qquad \forall q_h \in P^h. \end{aligned} \tag{2.8}$$

This leads to an algebraic system having the form of (1.2).

We would like conditions (i) and (ii) for the well-posedness of problem (2.3)-(2.5) to be satisfied also in the discrete case. As regards $\boldsymbol{V}$-ellipticity, the situation is simple and satisfactory. Since $\boldsymbol{V}^h \subset \boldsymbol{V}(\Omega)$, (2.6) implies

$$a(\boldsymbol{v}_h,\boldsymbol{v}_h) \geq \alpha \|\boldsymbol{v}_h\|^2_{\boldsymbol{V}} \quad \forall \boldsymbol{v}_h \in \boldsymbol{V}^h,$$

where α is independent of h. In contrast, the situation regarding the discrete LBB condition is more complicated. For any fixed mesh let

$$\beta(h) = \inf_{q_h \in P^h} \sup_{\boldsymbol{v}_h \in \boldsymbol{V}^h} \frac{(q_h, \operatorname{div} \boldsymbol{v}_h)}{\|q_h\|_0 \, \|\boldsymbol{v}_h\|_{\boldsymbol{V}}} \tag{2.9}$$

and consider a regular sequence of mesh refinements with $h \to 0$. The following cases can arise:

1. $\beta(h) \geq \beta_0 > 0 \quad \forall h \quad$ (for some β_0).
2. $\beta(h) > 0 \ \ \forall h$ and $\beta(h) \to 0, \quad h \to 0$.
3. $\beta(h) = 0 \ \ \forall h$.

Mixed elements satisfying Case 1 are said to be LBB-stable, or compatible.

We give the following discretization error estimates to illustrate the accuracy of LBB-stable finite elements (see [9]). They are derived under the assumption that $\mathcal{T}_h$ is regular. The second estimate requires, in addition, that Ω is convex.

(A) *First-order approximations on triangular elements.* The velocity is approximated by special quadratic polynomials and the pressure by piecewise constants.

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_1 + \|p - p_h\|_0 \leq C_1 \, h(\|\boldsymbol{u}\|_2 + \|p\|_1).$$

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_0 \leq C_2 \, h^2(\|\boldsymbol{u}\|_2 + \|p\|_1).$$

The same estimate holds for the Mini-element, see [1].

(B) *Higher-order approximations.* The velocity components are approximated by polynomials of degree $k+1, k \geq 2$. Polynomials of degree $k-1$ are used for the pressure.

If $\boldsymbol{u} \in \left(H^{k+1}(\Omega) \cap H_0^1(\Omega)\right)^2$ and $p \in H^k(\Omega) \cap L_0^2(\Omega)$ then

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_1 + \|p - p_h\|_0 \leq C_1 \, h^k(\|\boldsymbol{u}\|_{k+1} + \|p\|_k),$$

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_0 \leq C_2 \, h^{k+1}(\|\boldsymbol{u}\|_{k+1} + \|p\|_k).$$

3. AN EQUAL-ORDER APPROXIMATION METHOD

As stated earlier, our aim is the efficient iterative solution of the discrete Stokes equations on massively parallel (distributed memory) computers and a way to reach this goal is to use equal-order approximation with suitable regularization. Here we will focus on piecewise linear polynomials for the

approximation of both velocity and pressure. Here and in the remainder of the paper we assume that $P^h \subset H^1(\Omega) \cap L_0^2(\Omega)$.

Our regularization technique is based on a perturbed form of the weak formulation of the Stokes problem given by (2.3)-(2.5). To derive it, we note that the first equation in (1.1) implies the identity

$$-\Delta \boldsymbol{u} \cdot \nabla q + \nabla p \cdot \nabla q = \boldsymbol{f} \cdot \nabla q$$

and integrating over Ω we obtain, for any $q \in H^1(\Omega)$,

$$c(p,q) = (\boldsymbol{f}, \nabla q) + (\Delta \boldsymbol{u}, \nabla q)$$

where we have defined $c(p,q) = (\nabla p \cdot \nabla q)$. Thus the problem of finding $\boldsymbol{u} \in \left(H^2(\Omega)\right)^2$ and $p \in H^1(\Omega) \cap L_0^2(\Omega)$ such that

$$a(\boldsymbol{u}, \boldsymbol{v}) - b(p, \boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v}) \qquad \forall \boldsymbol{v} \in \boldsymbol{V}(\Omega), \tag{3.1}$$

$$b(q, \boldsymbol{u}) + \sigma\, c(p,q) = \sigma\, (\boldsymbol{f}, \nabla q) + \sigma\, d(q, \boldsymbol{u}) \qquad \forall q \in H^1(\Omega) \cap L_0^2(\Omega), \tag{3.2}$$

$$\boldsymbol{u} = \boldsymbol{g} \qquad \text{on } \partial\Omega, \tag{3.3}$$

where

$$d(q, \boldsymbol{u}) = \sum_{T_k} \int_{T_k} (\Delta \boldsymbol{u} \cdot \nabla q)\, d\Omega \tag{3.4}$$

is consistent with (2.3)-(2.5) for any σ. For the purpose of stabilization we take σ positive.

Suppose now for simplicity that $\boldsymbol{g} = \boldsymbol{0}$ and consider the following discrete analog of (3.1)-(3.3):

Find $\boldsymbol{u}_h \in \boldsymbol{V}^h$ and $p_h \in P^h$ such that

$$a(\boldsymbol{u}_h, \boldsymbol{v}_h) - b(p_h, \boldsymbol{v}_h) = (\boldsymbol{f}, \boldsymbol{v}_h) \qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}^h, \tag{3.5}$$

$$b(q_h, \boldsymbol{u}_h) + \sigma\, c(p_h, q_h) = \sigma\, (\boldsymbol{f}, \nabla q_h) + \sigma\, d(q_h, \boldsymbol{u}_h^*) \qquad \forall q_h \in P^h, \tag{3.6}$$

where $\boldsymbol{u}_h^*$ is an approximation of $\boldsymbol{u}$. Two special cases are

$$\boldsymbol{u}_h^* = \boldsymbol{u}, \tag{3.7}$$

which is of theoretical interest only, and

$$\boldsymbol{u}_h^* = \boldsymbol{u}_h, \tag{3.8}$$

which makes $d(q_h, \boldsymbol{u}_h^*) = 0$ for all q_h (because $\boldsymbol{u}_h$ is piecewise linear). The case of (3.8) is well known. (See, for example, [14] and [10]). It is shown in [14] that the following error estimates hold for this case when $\sigma = O(h^2)$:

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_1 + h\,\|p - p_h\|_1 \leq C\left(h\,\|\boldsymbol{u}\|_2 + h^2\|p\|_2\right). \tag{3.9}$$

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_0 + h\,\|p - p_h\|_0 \leq C\left(h^2\,\|\boldsymbol{u}\|_2 + h^3\|p\|_2\right). \tag{3.10}$$

Our goal now is to investigate the errors that arise from (3.5)-(3.6) in the case of a more general function $\boldsymbol{u}_h^*$. We begin by putting $\boldsymbol{v} = \boldsymbol{v}_h$ and $q = q_h$ in (3.1)-(3.6), obtaining

$$\begin{array}{rcll} a(\boldsymbol{u}, \boldsymbol{v}_h) - b(p, \boldsymbol{v}_h) & = & (\boldsymbol{f}, \boldsymbol{v}_h) & \forall \boldsymbol{v}_h \in \boldsymbol{V}^h, \\ b(q_h, \boldsymbol{u}) + \sigma\, c(p, q_h) & = & \sigma\,(\boldsymbol{f}, \nabla q_h) + \sigma\, d(q_h, \boldsymbol{u}) & \forall q_h \in P^h. \end{array} \tag{3.11}$$

Subtracting (3.5)-(3.6) from (3.11) and using the bilinear property of the various terms, we find that

$$\begin{array}{rcll} a(\boldsymbol{u} - \boldsymbol{u}_h, \boldsymbol{v}_h) - b(p - p_h, \boldsymbol{v}_h) & = & 0 & \forall \boldsymbol{v}_h \in \boldsymbol{V}^h, \\ b(q_h, \boldsymbol{u} - \boldsymbol{u}_h) + \sigma\, c(p - p^h, q_h) & = & \sigma\, d(q_h, \boldsymbol{u} - \boldsymbol{u}_h^*) & \forall q_h \in P^h. \end{array}$$

Let $\boldsymbol{u}_I$ denote the interpolant of $\boldsymbol{u}$ in $\boldsymbol{V}^h$, and let p_I denote the interpolant of p in P^h. We add $a(\boldsymbol{u}_I, \boldsymbol{v}_h)$ and $b(p_I, \boldsymbol{v}_h)$ to both sides of the first equation above and $b(q_h, \boldsymbol{u}_I)$ and $c(p_I, q_h)$ to both sides of the second and obtain, after suitable rearrangement,

$$\begin{aligned} a(\boldsymbol{u}_I - \boldsymbol{u}_h, \boldsymbol{v}_h) - b(p_I - p_h, \boldsymbol{v}_h) &= a(\boldsymbol{u}_I - \boldsymbol{u}, \boldsymbol{v}_h) - b(p_I - p, \boldsymbol{v}_h) \\ &\qquad\qquad \forall \boldsymbol{v}_h \in \boldsymbol{V}^h, \\ b(q_h, \boldsymbol{u}_I - \boldsymbol{u}_h) + \sigma\, c(p_I - p^h, q_h) &= b(q_h, \boldsymbol{u}_I - \boldsymbol{u}) + \sigma\, c(p_I - p, q_h) \\ &\quad + \sigma\, d(q_h, \boldsymbol{u} - \boldsymbol{u}_h^*) \qquad \forall q_h \in P^h. \end{aligned}$$

Putting $\boldsymbol{v}_h = \boldsymbol{u}_I - \boldsymbol{u}_h$ and $q_h = p_I - p_h$ and adding the two equations we have then

$$\begin{aligned} & a(\boldsymbol{u}_I - \boldsymbol{u}_h, \boldsymbol{u}_I - \boldsymbol{u}_h) + \sigma c(p_I - p_h, p_I - p_h) \\ = \;& a(\boldsymbol{u}_I - \boldsymbol{u}, \boldsymbol{u}_I - \boldsymbol{u}_h) - b(p_I - p, \boldsymbol{u}_I - \boldsymbol{u}_h) + b(p_I - p_h, \boldsymbol{u}_I - \boldsymbol{u}) \\ & + \sigma\, c(p_I - p, p_I - p_h) + \sigma d(p_I - p_h, \boldsymbol{u} - \boldsymbol{u}_h^*). \end{aligned} \tag{3.12}$$

Now $a(\boldsymbol{u}_I - \boldsymbol{u}_h, \boldsymbol{u}_I - \boldsymbol{u}_h) = |\boldsymbol{u}_I - \boldsymbol{u}_h|_1^2$ and $c(p_I - p_h, p_I - p_h) = |p_I - p_h|_1^2$. Further, applying Cauchy-Schwarz and Young inequality:

$$ab \leq \alpha\, a^2 + \frac{1}{4\alpha}\, b^2, \quad a, b, \alpha \in R, \;\; \alpha > 0$$

to each of the terms on the right-hand side of (3.12) we obtain

$$\begin{aligned}
a(\boldsymbol{u}_I-\boldsymbol{u},\boldsymbol{u}_I-\boldsymbol{u}_h) &\leq \alpha_1|\boldsymbol{u}_I-\boldsymbol{u}|_1^2+\tfrac{1}{4\alpha_1}|\boldsymbol{u}_I-\boldsymbol{u}_h|_1^2,\\
-b(p_I-p,\boldsymbol{u}_I-\boldsymbol{u}_h) &\leq \alpha_2|p_I-p|_0^2+\tfrac{1}{4\alpha_2}|\boldsymbol{u}_I-\boldsymbol{u}_h|_1^2,\\
b(p_I-p_h,\boldsymbol{u}_I-\boldsymbol{u}) &\leq \alpha_3|p_I-p_h|_1^2+\tfrac{1}{4\alpha_3}|\boldsymbol{u}_I-\boldsymbol{u}|_0^2,\\
c(p_I-p,p_I-p_h) &\leq \alpha_4|p_I-p|_1^2+\tfrac{1}{4\alpha_4}|p_I-p_h|_1^2,\\
d(p_I-p_h,\boldsymbol{u}-\boldsymbol{u}_h^*) &\leq \alpha_5|p_I-p_h|_1^2+\tfrac{1}{4\alpha_5}\textstyle\sum_{T_k}|\Delta\boldsymbol{u}-\Delta\boldsymbol{u}_h^*|_{0,T_k}^2.
\end{aligned}$$

In deriving the third inequality we have used the identity

$$\int_\Omega (p_I-p_h)\,\mathrm{div}(\boldsymbol{u}_I-\boldsymbol{u})\,d\Omega=-\int_\Omega \nabla(p_I-p_h)\cdot(\boldsymbol{u}_I-\boldsymbol{u})\,d\Omega.$$

Incorporating these results in (3.12) yields

$$\begin{aligned}
|\boldsymbol{u}_I-\boldsymbol{u}_h|_1^2 \;+\; \sigma|p_I-p_h|_1^2 &\leq \alpha_1\,|\boldsymbol{u}_I-\boldsymbol{u}|_1^2+\sigma\alpha_4\,|p_I-p|_1^2+\frac{1}{4\alpha_3}\,|\boldsymbol{u}_I-\boldsymbol{u}|_0^2\\
&+\left(\frac{1}{4\alpha_1}+\frac{1}{4\alpha_2}\right)|\boldsymbol{u}_I-\boldsymbol{u}_h|_1^2+\left(\alpha_3+\frac{\sigma}{4\alpha_4}+\sigma\alpha_5\right)|p_I-p_h|_1^2\\
&+\alpha_2\,|p_I-p|_0^2+\frac{\sigma}{4\alpha_5}\sum_{T_k}|\Delta\boldsymbol{u}-\Delta\boldsymbol{u}_h^*|_{0,T_k}^2. \qquad (3.13)
\end{aligned}$$

The α parameters are arbitrary positive numbers. With

$$\alpha_1=\alpha_2=1\,,\quad \alpha_3=\sigma/6\,,\quad \alpha_4=3/2\,,\quad \alpha_5=1/6$$

we can rewrite (3.13) as

$$\begin{aligned}
\frac{1}{2}\,|\boldsymbol{u}_I-\boldsymbol{u}_h|_1^2+\frac{\sigma}{2}\,|p_I-p_h|_1^2 &\leq |\boldsymbol{u}_I-\boldsymbol{u}|_1^2+\frac{3\sigma}{2}\,|p_I-p|_1^2 \qquad (3.14)\\
+\frac{3}{2\sigma}\,|\boldsymbol{u}_I-\boldsymbol{u}|_0^2 \;+\; |p_I-p|_0^2&+\frac{3\sigma}{2}\sum_{T_k}|\Delta\boldsymbol{u}-\Delta\boldsymbol{u}_h^*|_{0,T_k}^2.
\end{aligned}$$

Now applying standard interpolation theory we see that

$$\begin{aligned}
\frac{1}{2}\,|\boldsymbol{u}_I-\boldsymbol{u}_h|_1^2+\frac{\sigma}{2}\,|p_I-p_h|_1^2 &\leq O(h^2)+\sigma O(h^2)+\sigma^{-1}O(h^4)+O(h^4)\\
&+\frac{3\sigma}{2}\sum_{T_k}|\Delta\boldsymbol{u}-\Delta\boldsymbol{u}_h^*|_{0,T_k}^2. \qquad (3.15)
\end{aligned}$$

Thus if we put $\sigma=O(h^2)$ and assume only that

$$\sum_{T_k}|\Delta\boldsymbol{u}-\Delta\boldsymbol{u}_h^*|_{0,T_k}^2=O(h^0)=O(1) \qquad (3.16)$$

then we obtain the estimates

$$|\boldsymbol{u}_I - \boldsymbol{u}_h|_1 = O(h) \quad \text{and} \quad |p_I - p_h|_1 = O(1).$$

Finally, we can invoke triangle inequalities to bound the discretization errors as follows:

$$|\boldsymbol{u} - \boldsymbol{u}_h|_1 \leq |\boldsymbol{u} - \boldsymbol{u}_I|_1 + |\boldsymbol{u}_I - \boldsymbol{u}_h|_1 = O(h) + O(h) = O(h),$$

$$|p - p_h|_1 \leq |p - p_I|_1 + |p_I - p_h|_1 = O(h) + O(1) = O(1).$$

Estimates in the L^2-norm

Following a similar derivation in [14] and applying the classical Aubin-Nitsche trick, we obtain the following estimates

$$\|\boldsymbol{u} - \boldsymbol{u}_h\|_0 + h\|p - p_h\|_0 \leq h\|\boldsymbol{u} - \boldsymbol{u}_h\|_1 + h^2\|p - p_h\|_1 + h^2\|\boldsymbol{f}\|_0. \qquad (3.17)$$

A detailed derivation can be found in [3].

Our real interest, however, is not the orders-of-magnitude of these errors, which are those expected, but the role of $\Delta\boldsymbol{u}_h^*$ as revealed by the bound in (3.15). The bound decreases as $\Delta\boldsymbol{u}_h^* \to \Delta\boldsymbol{u}$, and this is the motivation for the following defect-correction scheme: Initially we put $\boldsymbol{u}_h^* = \boldsymbol{u}_h$, making $d(q_h, \boldsymbol{u}_h^*) = 0$ in (3.5)-(3.6) and

$$\frac{3\sigma}{2} \sum_{T_k} |\Delta\boldsymbol{u} - \Delta\boldsymbol{u}_h^*|_{0,T_k}^2 = \frac{3\sigma}{2} \sum_{T_k} |\Delta\boldsymbol{u}|_{0,T_k}^2$$

in (3.15). We solve (3.5)-(3.6) for $\boldsymbol{u}_h$ and p_h and then use $\boldsymbol{u}_h$ to compute a piecewise linear function $\boldsymbol{w}_h$ that approximates $\Delta\boldsymbol{u}$. (We do *not* compute a function $\boldsymbol{u}_h^*$ with the property $\Delta\boldsymbol{u}_h^* = \boldsymbol{w}_h$ since all that is needed is an approximation to $\Delta\boldsymbol{u}$). Then we solve (3.5)-(3.6) again, this time with

$$d(q_h, \boldsymbol{u}_h^*) \equiv \sum_{T_k} \int_{T_k} (\boldsymbol{w}_h \cdot \nabla q_h)\, d\Omega \qquad (3.18)$$

with the expectation that

$$\frac{3\sigma}{2} \sum_{T_k} |\Delta\boldsymbol{u} - \boldsymbol{w}_h|_{0,T_k}^2 < \frac{3\sigma}{2} \sum_{T_k} |\Delta\boldsymbol{u}|_{0,T_k}^2$$

decreasing the bound in (3.15). This process can naturally be repeated.

Regarding the computation of $\boldsymbol{w}^h$ in our numerical experiments (Section 5), we have worked with a uniform mesh on a rectangle. This has allowed us to define values of $\boldsymbol{w}^h(x,y)$ at the mesh nodes by applying the standard five-point difference approximation for Δ to any given approximation $\boldsymbol{u}_h$ to $\boldsymbol{u}$. Then we have extended the definition of $\boldsymbol{w}^h(x,y)$ to the entire domain by the usual piecewise linear interpolation.

One might hope that defect-correction, if allowed to continue indefinitely, would produce a sequence of solutions that converged to the solution of (3.5)-(3.6) for the case $\boldsymbol{u}^* = \boldsymbol{u}$, which would make

$$\frac{3\sigma}{2} \sum_{T_k} |\Delta \boldsymbol{u} - \Delta \boldsymbol{u}_h^*|^2_{0,T_k} = 0$$

in (3.15). This cannot happen, however, because the pointwise error in $\boldsymbol{w}^h(x,y)$ at the nodes is only $O(h^0)$. Nevertheless, in our experiments two to four iterations of defect-correction improved the solution noticeably (particularly the pressure), see Figures 1 and 2. After that the solution is more or less stable .

Finally, we mention a computationally attractive variant of the above procedure. It arises from the identity

$$\int_\Omega (\Delta \boldsymbol{v} \cdot \nabla q)\, d\Omega = \int_{\partial\Omega} q\, (\Delta \boldsymbol{v} \cdot \boldsymbol{n})\, ds$$

($\boldsymbol{n}$ is the unit outer normal on $\partial\Omega$), which holds for any divergence-free function $\boldsymbol{v}$ (i.e., $\operatorname{div} \boldsymbol{v} = 0$). Thus if we redefine the bilinear form in (3.4) as

$$d(q, \boldsymbol{u}) = \sum_{T_k} \int_{T_k \cap \partial\Omega} q\, (\Delta \boldsymbol{u} \cdot \boldsymbol{n})\, ds \tag{3.19}$$

then problem (3.1)-(3.3) is still consistent with (2.3)-(2.5).

Consider now the discrete formulation (3.5)-(3.6) with d defined by (3.19). The new solution is in general different from the old one because $\boldsymbol{u}_h^*$ may not be divergence-free. However, if $\boldsymbol{u}_h^*$ is a good approximation to $\boldsymbol{u}$ then it will be nearly divergence-free and the new solution to (3.5)-(3.6) will be close to the old one. The advantage of the new formulation is that the boundary integrals can be computed much more quickly than the corresponding interior integrals. In most of our computations we have used this faster variant, and the numerical results were in all cases indistinguishable.

4. THE ALGEBRAIC PROBLEM

The discrete variational problem (3.5)-(3.6) leads to an algebraic system in the form of (1.3). One can choose, however, between singular and nonsingular

versions. The singular version arises when the requirement

$$\int_\Omega p_h \, d\Omega = 0 \tag{4.1}$$

is dropped, the consequence being that both the function p_h and the vector $\boldsymbol{p}_h$ are unique only within an additive constant. (Our scheme has no spurious modes that would further increase the nonuniqueness). This simplifies the construction of (1.3) and is natural also from a theoretical point of view, since the restraint (4.1) is usually nonphysical. A way to impose (4.1) is to require that all basis functions in the expansion for $\boldsymbol{p}_h$ satisfy the same restraint. In practice, however, nonsingularity is often achieved simply by first constructing the singular system and then requiring that the pressure has a given value at a given node. A drawback of this procedure is that it can adversely affect the rate of convergence of iterative methods. Thus to avoid both the use of modified basis functions and a possible loss of rate of convergence, we work directly with the singular system. A consequence of this is that our computed pressure depends on the choice of the initial vector for the iterations. For the sake of uniqueness we remove the constant pressure mode at the end of the computation, thereby imposing (4.1).

Eliminating $\mathbf{U}_h$ from (1.3) leads to the Schur complement system

$$(\mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} + \sigma\mathbf{C})\boldsymbol{p}_h = \mathbf{B}^T\mathbf{A}^{-1}\mathbf{F} - \mathbf{G} - \sigma\widetilde{\mathbf{G}}. \tag{4.2}$$

After the pressure has been computed, the velocity can be recovered via the relation

$$\mathbf{U}_h = \mathbf{A}^{-1}\left(\mathbf{F} - \mathbf{B}\mathbf{p}_h\right). \tag{4.3}$$

We solve (4.2) by the conjugate gradient method. (Hereby we will not use any preconditioner. For problems with an irregular mesh it would, however, be advisable to use a diagonal scaling matrix as a preconditioner.) The rate of convergence depends on the effective condition number, $\kappa(\mathbf{S})$, of the symmetric positive semidefinite matrix

$$\mathbf{S} = \mathbf{B}^T\mathbf{A}^{-1}\mathbf{B} + \sigma\mathbf{C}. \tag{4.4}$$

The effective condition number is defined here as the ratio of the largest eigenvalue to the smallest *positive* eigenvalue. Let N denote the order of $\mathbf{S}$ (this is the number of nodes in the mesh) and let $\boldsymbol{e} = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$. Let $\mathbf{P}^h$ be the orthogonal complement of span$\{\boldsymbol{e}\}$ in $\mathbb{R}^N$, and suppose we have an equality of the form

$$0 < \gamma^2 \le \frac{\mathbf{p}^T\mathbf{S}\mathbf{p}}{\mathbf{p}^T\mathbf{M}_p\mathbf{p}} \le \Gamma^2 \quad \forall \mathbf{p} \in \mathbf{P}^h, \tag{4.5}$$

where $\mathbf{M}_p$ is the pressure mass matrix. Then it is easy to show that

$$\kappa(\mathbf{S}) \leq \frac{\Gamma^2}{\gamma^2}\,\kappa(\mathbf{M}_p).$$

Now it is well known that $\kappa(\mathbf{M}_p) = O(h^0)$, so the dependency of $\kappa(\mathbf{S})$ on h is a function of the ratio Γ^2/γ^2.

Inequalities of type (4.5) play an important role in the analysis of the LBB condition. More precisely, in the case of mixed elements without regularization ($\sigma = 0$) (4.5) is satisfied for $\gamma = \beta(h)$, where $\beta(h)$ is given by (2.9), and for a constant Γ independent of h. Thus mixed elements that are LBB–stable have the property that $\kappa(\mathbf{S}) = O(1)$. A major goal of regularization ($\sigma \neq 0$) is to make $\kappa(\mathbf{S}) = O(1)$ also for element pairs that do *not* satisfy the LBB condition. Considering now our own $\mathbf{S}$, an analysis based on [12] (Result 5.1) leads to the bound

$$\kappa(\mathbf{S}) \leq \frac{\left(2 + 36\frac{\sigma}{h^2}\right)}{\min\left(1, \dfrac{80\,\sigma}{h^2}\right)}\,\widehat{\gamma}^{-2}\kappa(\mathbf{M}_p), \tag{4.6}$$

where $\widehat{\gamma}$ is β_0 (the positive lower bound of $\beta(h)$) for the LBB-stable Mini-element.

Corollary 4.1. Our regularized equal-approximation method makes $\kappa(\mathbf{S}) = O(1)$ if we put $\sigma = C\,h^2$, where C is a constant. Further, the number of conjugate gradient iterations required to solve (4.2) to given accuracy is then bounded independently of h.

Analogous results for reduced systems based on compatible pairs are presented in [5], [11], [15] and [20].

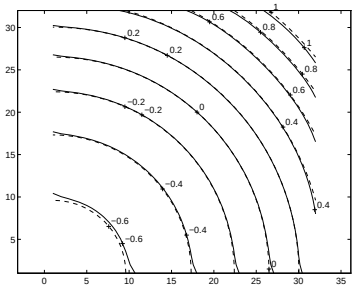
We turn our attention now to a single step of of the conjugate gradient method applied to (4.2). Almost all of the work is required for an operation of the type

$$\mathbf{q} = \mathbf{S}\mathbf{p} = \mathbf{B}^T(\mathbf{A}^{-1}(\mathbf{B}\mathbf{p})) + \sigma\mathbf{C}\mathbf{p}$$

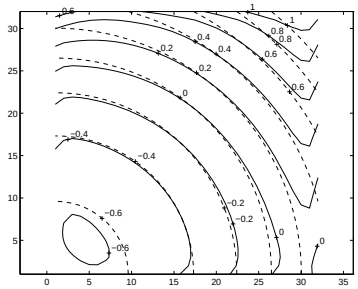
for given $\mathbf{p}$. The work here is dominated, in turn, by the need to solve a symmetric positive definite system of the type

$$\mathbf{A}\mathbf{w} = \widetilde{\mathbf{w}} \tag{4.7}$$

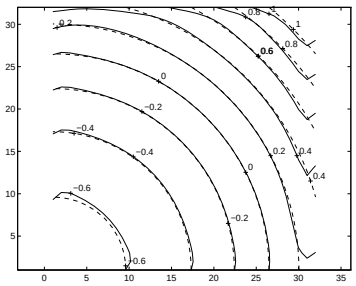
for given $\widetilde{\mathbf{w}}$. To solve (4.7) we apply the preconditioned conjugate gradient method, thus creating an inner-outer iteration type of algorithm. We have applied here a short-length version of the AMLI preconditioner described extensively in [4]. It turns out that (4.7) can be solved to given accuracy in a



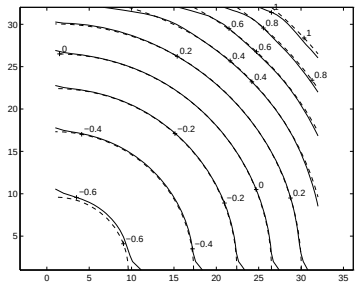
a) The rhs term $\sum_{T_k} \int_{T_k} (\Delta \boldsymbol{u}_h \cdot \nabla q) \ d\Omega$ computed from exact u



b) The rhs term $\sum_{T_k} \int_{T_k} (\Delta \boldsymbol{u}_h \cdot \nabla q) \ d\Omega$ neglected



c) After four defect-correction steps

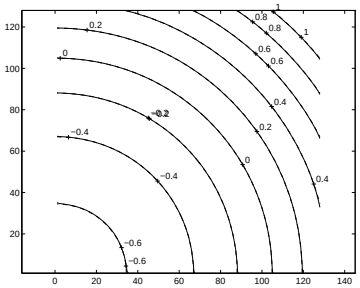


d) After four defect-correction steps

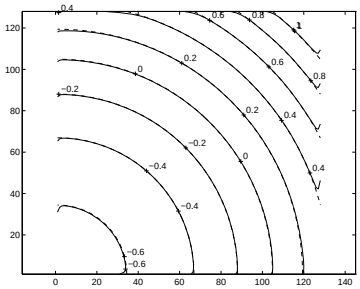
Figure 1. Contour lines for the pressure: $h = 0.03226$ (exact pressure: - - -; computed pressure: ——)

number of iterations that is bounded independently of h. Consequently, the entire algorithm for solving (4.2) is optimal in the sense that the total work is proportional to the number of unknowns.

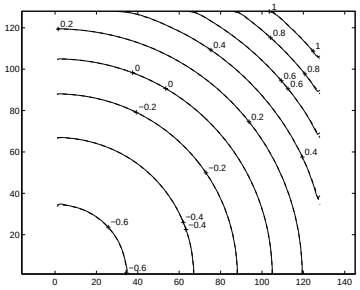
Remark 4.1. We stress that aiming at the efficient solution of the Stokes problem on massively parallel computer architectures and using an equal-order approximation method, combined with a proper regularization, go hand-in-hand. As discussed in [4], an iterative solution method can be efficient on a parallel machine if it is optimal on a serial computer. The latter condition restricts the choice of a solution method to the class of multilevel methods. However, the efficient implementation of LBB-stable finite element pairs is far from trivial, and multigrid methods, for instance, are not even always applicable for compatible pairs (see [7]). Further, although the Mini-element is compatible, it yields a pressure solution that it is often polluted by oscillations. This is demonstrated in [13], see also Figure 5.1 in [3]. The explanation of this phenomenon is that in general stabilization via bubble functions is equivalent to a regularized formulation with a particular value of the regular-



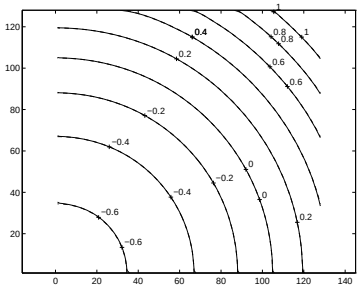
a) The rhs term $\sum_{T_k} \int_{T_k} (\Delta \boldsymbol{u}_h \cdot \nabla q) \ d\Omega$ computed from exact u



b) The rhs term $\sum_{T_k} \int_{T_k} (\Delta \boldsymbol{u}_h \cdot \nabla q) \ d\Omega$ neglected



c) After four defect-correction steps



d) After four defect-correction steps

Figure 2. Contour lines for the pressure: $h = 0.007874$ (exact pressure: - - -; computed pressure: ——)

ization parameter σ (see also [13] and [12]), and the presence of oscillations depends on the size of σ. In the case of the Mini-element, it turns out that the value of σ is too small such that significant oscillations are produced.

5. NUMERICAL RESULTS

The Stokes equations (1.1) in $\mathbb{R}^2$ read as follows:

$$\begin{aligned} -\Delta u + p_x &= f_1(x,y) \ \text{ in } \Omega, \\ -\Delta v + p_y &= f_2(x,y) \ \text{ in } \Omega, \\ u_x + v_y &= 0 \ \text{ in } \Omega, \\ u(x,y)|_{\partial\Omega} &= g_1(x,y), \\ v(x,y)|_{\partial\Omega} &= g_2(x,y). \end{aligned} \tag{5.1}$$

Let $\Omega = (0,1)^2$ be the unit square. We consider the following two test problems:

Problem 5.1. The exact solution of (5.1) is chosen to be $u(x,y) = x^3 + x^2 - 2xy + x$, $v(x,y) = -3x^2y + y^2 - 2xy - y$ and $p(x,y) = x^2 + y^2$. The functions f_1 and f_2, and the boundary conditions are computed correspondingly.

Problem 5.2. [*The driven cavity flow problem*] In this case f_1, f_2, g_1 and g_2 are zero, except $g_1(x,1) = 1$.

The stopping criteria for the outer (unpreconditioned) CG method and for the inner (AMLI preconditioned) CG method are, respectively,

$$\frac{\boldsymbol{r}_i^T \boldsymbol{r}_i}{\boldsymbol{r}_1^T \boldsymbol{r}_1} < 10^{-12}, \quad \frac{\boldsymbol{r}_i^T \mathbf{M}^{-1} \boldsymbol{r}_i}{\boldsymbol{r}_1^T \mathbf{M}^{-1} \boldsymbol{r}_1} < 10^{-12},$$

where $\boldsymbol{r}_1$ and $\boldsymbol{r}_i$ are the initial and current computed residuals, respectively. A short-length version of the AMLI preconditioner is used, and the systems on the coarsest level are solved by a diagonally preconditioned CG method with a stopping criterion

$$\frac{\boldsymbol{r}_i^T \mathbf{D}^{-1} \boldsymbol{r}_i}{\boldsymbol{r}_1^T \mathbf{D}^{-1} \boldsymbol{r}_1} < 10^{-3}.$$

Figure 1 shows the behavior of the computed pressure for $h = 0.03226$ in the following cases: (a) computing the right-hand side term (3.4) using the exact solution $\boldsymbol{u}$, (b) neglecting this term in the right-hand side, (c) after one defect-correction step using (3.18) to approximate (3.4), and (d) after four defect-correction steps. Figure 2 shows the corresponding results for $h = 0.007874$. Table 1 shows the accuracy of the computed velocity components and pressure for Problem 5.1. In Table 2, the iteration counts for different sizes of the Schur complement system $\mathbf{S}$ are shown. Table 3 shows the performance of the method on a Cray T3E-600 computer with 64 processors (DEC Alpha 21164, 300 Mhz, 3D–torus). We present iteration counts and elapsed time in seconds. There are four different times shown in Table 3:

- *total* is the total execution time which includes generation of the system matrix and the construction of the preconditioner;
- *outer* is the total solution time;
- *coars.* is the time spent to solve systems on the coarsest level during each preconditioning step;
- *comm.* is the total time spent on communications during the solution process.

The code is written in High Performance Fortran (HPF) using `BLAS` system subroutines for the vector operations and the `shmem` communication library available for the Cray T3E computer.

The major observations regarding Tables 3 are the following:

(1) The solution method shows very good scalability. The factor two in decreasing the total execution time is clearly present in all experiments.

(2) The experiments confirm the theory how to choose the coarsest level for the short-AMLI preconditioner. There is a clearly seen minimum in the total solution time for a properly chosen coarsest level in the multilevel recursion, corresponding to the formula $\lfloor \frac{2}{3}\ell \rfloor$, where ℓ is the total number of levels (for two-dimensional problems). The derivation of the formula how to determine the coarsest level number in the AMLI recursion can be found, for instance, in [4]. Earlier numerical experiments with the same method implemented on a CM-200 and CM-5 computer showed as a best choice the coarsest level determined from the formula $\lceil \frac{2}{3}\ell \rceil$. The difference reflects the fact that the communication network for the Cray T3E computer is significantly better than that of the CM-200/5 computers. On the Cray the communications are almost negligible, so although the communication differences are larger, the coarsest level has to be lower in order to balance the overall computational work.

(3) When the problem size is increased (in our case four times each time) the total computational time grows with a factor five. The factor is fairly constant for the best choice of the coarsest level in the multilevel recursive preconditioner. It is not the ideal factor four, however, due to the fact that we have a V-cycle preconditioner of nearly optimal order.

The superlinear speedup observed for some experiments presented in Table 3 is due to subproblem sizes better matching the cache size of the processors.

The numerical results show that the chosen solution method parallelizes very well on distributed memory machines and confirms that the short AMLI preconditioner can be recommended when preconditioned iterative methods are to be implemented for large parallel computations.

6. CONCLUSIONS

In this study we have solved the Stokes problem by combining an equal-order approximation for velocity and pressure (piecewise-linear approximation in both cases) with a consistent regularization procedure. The method allows the use of a bigger stabilization constant than in the mini-element. To improve the accuracy of the computed pressure and remove unwanted oscillations we have carried out a few steps of a defect-correction technique. In this way the approximation order, which is better than for the element pair piecewise linear–piecewise constants, is close to that for the mini-element but without the associated oscillations in the pressure occurring for the latter and with simpler elements. The expected improvement in the computed pressure is confirmed by numerical experiments.

The elimination of the velocity unknowns yields a symmetric positive semidefinite system for pressure which we have solved by an inner-outer iteration procedure. For the outer iterations we have used the unpreconditioned con-

The boundary term containing Δu is neglected

Grid	h	$\|\widehat{\boldsymbol{u}} - \boldsymbol{u}\|_\infty$	$\|\widehat{\boldsymbol{v}} - \boldsymbol{v}\|_\infty$	$\|\widehat{p} - p\|_\infty$	$\|\widehat{p} - p\|_{L_2}$
64^2	$1.587\,10^{-2}$	$1.911\,10^{-3}$	$1.250\,10^{-3}$	0.201	$2.246\,10^{-2}$
128^2	$7.874\,10^{-3}$	$4.709\,10^{-4}$	$3.154\,10^{-4}$	$9.970\,10^{-2}$	$7.808\,10^{-3}$
256^2	$3.922\,10^{-3}$	$1.171\,10^{-4}$	$7.940\,10^{-5}$	$5.041\,10^{-2}$	$2.572\,10^{-3}$
512^2	$1.957\,10^{-3}$	$2.924\,10^{-5}$	$1.988\,10^{-5}$	$2.587\,10^{-2}$	$8.662\,10^{-4}$
1024^2	$9.775\,10^{-4}$	$8.035\,10^{-6}$	$5.154\,10^{-6}$	$1.358\,10^{-2}$	$2.976\,10^{-4}$

After one defect-correction step:

Grid	h	$\|\widehat{\boldsymbol{u}} - \boldsymbol{u}\|_\infty$	$\|\widehat{\boldsymbol{v}} - \boldsymbol{v}\|_\infty$	$\|\widehat{p} - p\|_\infty$	$\|\widehat{p} - p\|_{L_2}$
64^2	$1.587\,10^{-2}$	$1.206\,10^{-3}$	$1.008\,10^{-3}$	0.108	$1.579\,10^{-2}$
128^2	$7.874\,10^{-3}$	$3.001\,10^{-3}$	$2.557\,10^{-4}$	$5.335\,10^{-2}$	$5.210\,10^{-3}$
256^2	$3.922\,10^{-3}$	$7.497\,10^{-5}$	$6.468\,10^{-5}$	$2.638\,10^{-2}$	$1.744\,10^{-3}$
512^2	$1.957\,10^{-3}$	$1.863\,10^{-5}$	$1.646\,10^{-5}$	$1.300\,10^{-2}$	$5.931\,10^{-4}$
1024^2	$9.775\,10^{-4}$	$5.504\,10^{-6}$	$4.560\,10^{-6}$	$8.240\,10^{-3}$	$2.054\,10^{-4}$

After two defect-correction steps:

Grid	h	$\|\widehat{\boldsymbol{u}} - \boldsymbol{u}\|_\infty$	$\|\widehat{\boldsymbol{v}} - \boldsymbol{v}\|_\infty$	$\|\widehat{p} - p\|_\infty$	$\|\widehat{p} - p\|_{L_2}$
64^2	$1.587\,10^{-2}$	$6.968\,10^{-4}$	$9.464\,10^{-4}$	$8.259\,10^{-2}$	$1.112\,10^{-2}$
128^2	$7.874\,10^{-3}$	$1.769\,10^{-4}$	$2.408\,10^{-4}$	$4.240\,10^{-2}$	$3.705\,10^{-3}$
256^2	$3.922\,10^{-3}$	$4.461\,10^{-5}$	$6.103\,10^{-5}$	$2.151\,10^{-2}$	$1.248\,10^{-3}$
512^2	$1.957\,10^{-3}$	$1.097\,10^{-5}$	$1.582\,10^{-5}$	$1.078\,10^{-2}$	$4.262\,10^{-4}$

Table 1.
Problem 5.1: Accuracy of the computed solution ($\sigma = h^2$).

Table 2.
Iteration counts ($\sigma = h^2$).

Grid	Coarsest level No. (total no. of levels)	Problem 5.1 outer CG iter.	Problem 5.1 inner PCG iter.	Problem 5.2 outer CG iter.	Problem 5.2 inner PCG iter.	Problem 5.1 outer CG iter.	Problem 5.1 inner CG iter. min/max
64^2	8(12)	28	8	29	8	28	160/169
128^2	10(14)	26	8	30	8	27	319/329
256^2	12(16)	26	8	30	8	27	625/672
512^2	12(18)	26	10	30	10	28	1233/1363
1024^2	14(18)	26	11	30	11	29	2361/2558

Grid size	Levels: coarse (fine)	Number of PEs 2	4	8	16	32	64	Time (sec)
		161.76	79.77	41.96				total
128^2	10(14)	160.76	79.09	41.66				outer
		89.18	45.61	24.00				coars.
		2.46	2.67	3.29				comm.
			406.62	190.54	94.61	49.55	28.90	total
	10(16)		403.49	189.06	93.86	49.18	28.71	outer
			159.75	80.09	41.63	21.97	13.20	coars.
			5.31	5.93	5.58	4.62	3.89	comm.
			440.90	213.34	107.87	56.79	34.21	total
256^2	12(16)		438.01	211.96	107.16	56.43	34.04	outer
			283.38	142.39	74.42	39.19	24.25	coars.
			5.75	7.32	7.41	5.90	4.49	comm.
					632.60	304.24	154.65	total
512^2	12(18)				629.44	302.71	153.81	outer
					363.38	183.18	96.15	coars.
					14.28	12.14	10.14	comm.
						1662.73	829.71	total
1024^2	12(20)					1655.73	826.22	outer
						810.11	422.25	coars.
						29.89	22.26	comm.

Table 3.
Problem 5.1: Performance results on the Cray T3E-600 computer ($\sigma = h^2$).

jugate gradient method and for the inner iterations the conjugate gradient method with a preconditioning based on the short-recursion AMLI technique. Our presentation has emphasized the parallel aspects of this approach. The numerical results obtained using the MPP Cray T3E-600 computer demonstrate that this method for the Stokes problem is (nearly) optimal, efficient and scalable.

ACKNOWLEDGEMENTS

The parallel numerical experiments on the MPP Cray T3E-600 computer were done within the NCF project MP-10 *Preconditioning methods for large scale scientific models* sponsored by the Dutch Foundation for National Computer Facilities (Stichting Nationale Computer Faciliteiten).

REFERENCES

[1] D.N. Arnold, F. Brezzi and M. Fortin. A Stable Finite Element for the Stokes Equations. *Calcolo*, **21** , 1984, 377 – 344.

[2] O. Axelsson. Preconditioning of Indefinite Problems by Regularization. *SIAM J. Numer. Anal.*, **16** , 1979, 55 – 69.

[3] O. Axelsson, V.A. Barker, M. Neytcheva, B. Polman. Solving the Stokes problem on a Massively Parallel Computer. Report 9914 (1999), Dep. of Mathematics, University of Nijmegen, The Netherlands .

[4] O.Axelsson, M.Neytcheva, Scalable Algorithms for the solution of Navier's Equations of Elasticity, *J. Comput. Appl. Math.*, 63 (1995), 149-178.

[5] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods*, Springer-Verlag, New York, 1991.

[6] F. Brezzi and J. Pitkäranta, On the Stabilization of Finite Element Approximations of the Stokes Equations. In W. Hackbusch (ed.), *Efficient Solution of Elliptic Systems, Notes on Numerical Fluid Mechanics* 10, Vieweg, Braunschweig, 1984, 11-19.

[7] H.C. Elman, Multigrid and Krylov Subspace Methods for the Discrete Stokes Equations. In *Proceedings PCG'94, Matrix Analysis and Parallel Computing*, Keio University, March 14-16, 1994, 151-164.

[8] L. Franca, T. Hughes and R. Stenberg, Stabilized Finite Element Methods. In M. Gunzburger and R. Nicolaides (eds), *Incompressible Computational Fluid Dynamics. Trends and Advances*, Cambridge University Press, 1993, 87-107.

[9] V. Girault and P.-A. Raviart, *Finite Element Approximations of the Navier-Stokes Equations*, Lecture Notes in Mathematics, 749, Springer-Verlag, 1979.

[10] T. Hughes, L. Franca and M. Balestra, A New Finite Element Formulation for Computational Fluid Dynamics: V. Circumventing the Babuška-Brezzi condition: A Stable Petrov-Galerkin Formulation of the Stokes Problem Accommodating Equal-Order Interpolations, *Comput. Meth. Appl. Mech. Eng.*, 59 (1986), 85-99.

[11] U. Langer and W. Queck, Preconditioned Uzawa-type Iteration Methods for Solving Mixed Finite Element Equations, Theory- Applications-Software. Wissenschaftliche Schriftenreihe der TU Karl-Marx-Stadt, 3, 1987.

[12] J.F. Maitre and E. Wabo, Stabilized Formulations and Mini-Element for the n-Dimensional Stokes Equations: Properties and Solution Procedures. In M. Křižek et al. (eds), *Finite Element Methods. 50 years of the Courant element.* Conference held at the University of Jyvaeskylae, Finland, 1993. *Lecture Notes Pure Appl. Math.*, 164 (1994), 285-299.

[13] R. Pierre, Simple C^0 Approximations for the Computation of Incompressible Flows, *Comp. Meth. Appl. Mech. Eng.*, 68 (1988), 205-227.

[14] R. Pierre, Regularization Procedures of Mixed Finite Element Approximations of the Stokes Problem, *Num. Meth. Partial Diff. Equations*, 5 (1989), 241-258.

[15] O. Pironneau, *Finite Element Methods for Fluids*, John Wiley & Sons and Masson, 1989.

[16] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*, Springer-Verlag, 1994.

[17] D. Silvester and A. Wathen, Fast Iterative Solution of Stabilized Stokes Systems Part I: Using Simple Diagonal Preconditioners, *SIAM J. Numer. Anal.*, 30 (1993), 630-649.

[18] A. Wathen and D. Silvester, Fast Iterative Solution of Stabilized Stokes Systems Part II: Using General Block Preconditioners, *SIAM J. Numer. Anal.*, 31 (1994), 1352-1367.

[19] R. Temam, *Navier-Stokes Equations. Theory and Numerical Analysis*, Studies in Mathematics and its Applications, North Holland, 1979.

[20] R. Verfürth, A Combined Conjugate Gradient-Multigrid Algorithm for the Numerical Solution of the Stokes Problem, *IMA J. Numer. Anal.* 4 (1984), 441-455.

[21] O. Zienkewicz, Y. Liu and G. Huang, Error Estimates and Convergence Rates for Various Incompressible Elements, *Int. J. Num. Meth. Eng.*, 28 (1989), 2191-2202.

STOKSO UŽDAVINIO SPRENDIMAS GALINGAIS LYGIAGREČIAISIAIS KOMPIUTERIAIS

O. Axelsson, V.A. Barker, M. Neytcheva, B. Polman

Aprašomas skaitinis metodas stacionariajam dvimačiam Stokso uždaviniui. Metodas pagrįstas baigtinių elementų aproksimacija greičiui ir slėgiui, stabilumo reguliarizacija ir defektų taisymo metodu, kuris pagerina tikslumą. Eliminuojant nežinomus greičius iš algebrinės lygčių sistemos slėgiui surasti gaunama simetrinė teigiamai pusapibrėžtinė sistema, kuri sprendžiama vidinėmis-išorinėmis iteracijomis. Išorinė iteracija naudoja besąlyginį jungtinių gradientų metodą. Vidinės iteracijos, kurių kiekviena atitinka kraštinio elipsinio uždavinio sprendimą kiekvienai greičio komponentei, naudoja sąlyginį jungtinių gradientų metodą. Žinant slėgį surandamas greitis. Metodas yra ekonomiškas, nes kompiuterio skaičiavimai proporcingi nežinomųjų skaičiui. Metodas pritaikytas išnaudoti galingų lygiagrečiųjų kompiuterių su paskirstyta atmintimi architektūrą. Skaitiniai eksperimentai kompiuteriu Cray T3E iliustruoja metodo išlygiagretinimą.