PRECONDITIONERS FOR SADDLE POINT PROBLEMS ARISING IN COMPUTATIONAL FLUID DYNAMICS

HOWARD C. ELMAN*

Abstract. Discretization and linearization of the incompressible Navier-Stokes equations leads to linear algebraic systems in which the coefficient matrix has the form of a saddle point problem

$$\begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{0.1}$$

In this paper, we describe the development of efficient and general iterative solution algorithms for this class of problems. We review the case where (0.1) arises from the steady-state Stokes equations and show that solution methods such as the Uzawa algorithm lead naturally to a focus on the Schur complement operator $BF^{-1}B^T$ together with efficient strategies of applying the action of F^{-1} to a vector. We then discuss the advantages of explicitly working with the coupled form of the block system (0.1). Using this point of view, we describe some new algorithms derived by developing efficient methods for the Schur complement systems arising from the Navier-Stokes equations, and we demonstrate their effectiveness for solving both steady-state and evolutionary problems.

*Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA, e-mail: elman@cs.umd.edu. This work was supported by the U. S. National Science Foundation under grant DMS9972490.

1. Introduction. The aim of this paper is to give an overview of the development and utility of some new algorithms for computing the numerical solution of the Navier-Stokes equations

$$\alpha \mathbf{u}_t - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \mathrm{grad})\, \mathbf{u} + \mathrm{grad}\, p = \mathbf{f} \tag{1.1}$$

subject to incompressibility constraints

$$-\mathrm{div}\, \mathbf{u} = 0. \tag{1.2}$$

The problem is posed on an open bounded domain Ω in $\mathbb{R}^2$ or $\mathbb{R}^3$, with suitable boundary conditions specified on $\partial\Omega$. The parameter α has the value 0 for steady-state versions of (1.1) and 1 for evolutionary problems.

The methods under consideration are applicable to a broad collection of discrete versions of (1.1)–(1.2), all of which treat the primitive variable formulation directly and require the solution of a series of linear systems of equations in which the coefficient matrix is of the form arising in saddle point problems,

$$\begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix}.$$

The philosophical points of view behind the algorithm development are:

1. The methods are derived using tools for simpler problems as building blocks. In particular, they depend on having efficient numerical algorithms for solving two subsidiary problems, the scalar Poisson equation and the scalar convection-diffusion equation. Moreover, it is possible to make use of approximate solutions of these subsidiary problems obtained using iterative methods.
2. The solution strategies adapt in a straightforward manner to variants of the problem. In principle, the same code could be used to handle evolutionary problems (with suitable time discretizations), steady-state problems, or Stokes systems.

An overview of the paper is as follows. We will assume throughout that the spatial discretization is div-stable; we have in mind low order finite element discretizations that satisfy an inf-sup condition, or the MAC finite difference discretization [15, 17, 21]. The solution methods considered are generalizations of techniques developed originally for the steady-state Stokes equations, and in Section 2 we outline the derivation of our point of view as it evolved for this problem class. In Section 3, we introduce the solution methodology for the steady-state Navier-Stokes equations and discuss its properties. In Section 4, we discuss some strategies for time discretization to which the methodology applies directly, and we demonstrate its performance in this setting. A brief summary of the results of these sections is that for the Stokes equations, it is possible to develop optimal algorithms whose convergence rates are independent of the discretization mesh size used, provided methods with this property (such as multigrid) are available for certain subproblems entailing the Poisson equation. A similar statement applies to the steady-state Navier-Stokes equations, although performance does depend to some extent on Reynolds numbers. For transient problems, this dependence becomes negligible especially for small time steps. In Section 5, we give a brief discussion of the relative merits of the ideas considered here and multigrid strategies for (1.1)–(1.2), and in Section 6 we make some concluding remarks.

2. Background: The steady-state Stokes equations. We begin by reviewing some results for the Stokes equations

$$\begin{aligned} -\Delta \mathbf{u} + \operatorname{grad} p &= \mathbf{f} \\ -\operatorname{div} \mathbf{u} &= 0. \end{aligned}$$

Discretization leads to a linear system

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix} \tag{2.1}$$

where, for problems in d dimensions, A is a block diagonal matrix consisting of a set of d independent discrete Laplace operators. The classical Uzawa algorithm [1] starts with an arbitrary initial value p_0 and performs the iteration

$$\begin{array}{l} \texttt{for } k = 0 \texttt{ until} \text{ convergence } \texttt{do} \\ \quad u_{k+1} = A^{-1}(f - B^T p_k) \\ \quad p_{k+1} = p_k + \beta\, B u_{k+1} \\ \texttt{enddo} \end{array} \tag{2.2}$$

Here, β is a scalar parameter that must be determined prior to the iteration.

Substitution of u_{k+1} from the first step of (2.2) into the second step shows that this computation is equivalent to an iteration for the pressures,

$$p_{k+1} = p_k + \beta\, BA^{-1}(f - B^T p_k).$$

This is a Richardson iteration [36] for the Schur complement system

$$BA^{-1}B^T p = BA^{-1} f. \tag{2.3}$$

The errors satisfy

$$p - p_k = (I - \beta(BA^{-1}B^T))^k (p - p_0),$$

and in the Euclidean norm,

$$\|p - p_k\|_2 \leq \left[\rho\left(I - \beta(BA^{-1}B^T)\right)\right]^k \|p - p_0\|_2 .$$

The optimal convergence rate is achieved with the choice of β for which the algebraically smallest and largest eigenvalues of $I - \beta(BA^{-1}B^T)$ are equal in absolute value, i.e.

$$\beta = 2/(\lambda_{min} + \lambda_{max}), \tag{2.4}$$

where λ_{min} and λ_{max} are the extreme eigenvalues of $BA^{-1}B^T$. For this choice, the *convergence factor* is

$$\rho(I - \beta(BA^{-1}B^T)) = (\kappa - 1)/(\kappa + 1),$$

where $\kappa = \lambda_{max}/\lambda_{min}$ is the condition number of $BA^{-1}B^T$.

That this iteration is rapidly convergent for the Stokes equations is a consequence of the properties of the Schur complement $BA^{-1}B^T$. Let M_p denote the mass matrix associated with the pressure discretization. It is well known that the Schur complement satisfies

$$\gamma^2 \leq \frac{(p, BA^{-1}B^T p)}{(p, M_p v)} \leq \Gamma^2,$$

where $\gamma \geq 0$ is the inf-sup constant for the discretization and $\Gamma \leq \sqrt{d}$ [37]. We are assuming that the spatial discretization is div-stable, so that γ does not depend on the discretization mesh size h. Consequently, the Schur complement operator is spectrally equivalent to the pressure mass matrix. This assertion also applies to the marker-and-cell (MAC) finite difference scheme [21], for which the mass matrix on a uniform grid is $h^d I$. Moreover, the pressure mass matrix is itself spectrally equivalent to its diagonal, which also essentially has the form $h^d I$ [40]. It follows that $\kappa(BA^{-1}B^T)$ is independent of h. This discussion also suggests that preconditioning by the mass matrix or some spectrally equivalent approximation Q_{M_p} may be beneficial, which is indeed the case [11]. The preconditioned Uzawa algorithm updates the pressures as

$$p_{k+1} = p_k + \beta\, Q_{M_p}^{-1} B u_{k+1}.$$

If β is defined as in (2.4) where λ_{min} and λ_{max} now represent extrema of the Rayleigh quotient $\frac{(q, BA^{-1}B^T q)}{(q, Q_{M_p} q)}$, then we have the convergence bound

$$\|p - p_k\|_{Q_{M_p}} \leq \rho^k \|p - p_0\|_{Q_{M_p}}$$

where $\rho = (\kappa - 1)/(\kappa + 1)$ with $\kappa = \lambda_{max}/\lambda_{min}$.

Thus, the convergence rate of the Uzawa algorithm for solving the Stokes equations is independent of discretization mesh size. An implementation requires the application of the action of the inverse of A to a vector. If fast methods such as multigrid are available for this computation, then the Uzawa method is also an optimal strategy with respect to operation counts. Nevertheless, there are some potential drawbacks:

1. *It requires the parameter* β*.* Bounds on the inf-sup parameter can often be used to estimate this quantity, but the need to specify it adds a difficulty to implementation.
2. *In some sense it is slow.* In the context of iterative methods for symmetric positive-definite systems of equations, it is known that Richardson iteration is slower to converge than the conjugate gradient method (CG). The convergence factor for CG is bounded by $\rho = (\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$ [26].
3. *The action of the inverse of* A *is potentially costly.* This is clearly the dominant cost of the algorithm. Although fast solvers for the Poisson equation are available, it would be desirable to avoid *accurate* solution of this problem at every step.

Application of CG to (2.3) resolves the first and second of these difficulties. However, a matrix-vector product by $BA^{-1}B^T$ is required at each iteration, and we know of no way of avoiding an accurate computation of this operation. This entails accurate solution of the Poisson equation. By way of contrast, because the Uzawa algorithm derives from the coupling between the pressure and velocity, its first step can be replaced by an approximate computation of the action of A^{-1} to produce an algorithm with similar convergence characteristics [4, 11, 41]. It is not straightforward to automate this process, however, since the inner iteration for the Poisson equation requires a stopping criterion.

Thus, neither of these strategies completely resolves the drawbacks listed above, and we have come to prefer an alternative approach, developed independently by Rusten and Winther [25] and Silvester and Wathen [29, 39], which treats the saddle point problem (2.1) directly. The system (2.1) is symmetric indefinite, so that the MINRES Krylov subspace method [22] is applicable. When this method is applied to a system $\mathcal{A}x = b$, the residual $r_k = b - \mathcal{A}x_k$ of the kth iterate satisfies

$$\|r_k\| \le \min_{p_k(0)\in\Pi_k} \max_{\lambda\in\sigma(\mathcal{A})} |p_k(\lambda)| \, \|r_0\|, \tag{2.5}$$

where Π_k denotes the set of all real polynomials p_k of degree at most k for which $p_k(0) = 1$, and $\sigma(\mathcal{A})$ is the set of eigenvalues of $\mathcal{A}$. (We will be more precise about the norm below.) If $\sigma(\mathcal{A})$ is contained in two intervals

$$[-a, -b] \cup [c, d], \qquad a,\ b,\ c,\ d > 0, \tag{2.6}$$

where $a-b=d-c$, then the convergence factor is bounded by

$$2\left(\frac{1-\sqrt{(bc)/(ad)}}{1+\sqrt{(bc)/(ad)}}\right)^{1/2}.$$

MINRES can be combined with a symmetric positive-definite preconditioner with the aim of reducing the size of the intervals containing the eigenvalues. The considerations above point to the choice

$$\mathcal{Q}=\begin{pmatrix} Q_A & 0 \\ 0 & Q_{M_p} \end{pmatrix}. \tag{2.7}$$

Here, as above, Q_{M_p} represents an approximation to the mass matrix, which is therefore a good approximation to the Schur complement, and Q_A represents an approximation to A. In particular, if Q_A is spectrally equivalent to A, i.e.,

$$\theta_1 \leq \frac{(v,Av)}{(v,Q_A v)} \leq \theta_2$$

for θ_1, θ_2 independent of the mesh size, then so are the intervals of (2.6), as well as the convergence factor for MINRES.

Detailed discussions of the effectiveness of this approach are given in [10, 25, 29]. The key point is that it achieves optimal convergence rates automatically, without a need for exact computation of the action of A^{-1}, or estimates of any parameters, or stopping criteria associated with an inner iteration. Q_A can be defined using any operation available, such as multigrid or domain decomposition. In practice, one step of V-cycle multigrid is an effective choice.

We conclude this review with two additional observations. First, when preconditioning is used, the norm appearing in the expression (2.5) depends on the choice of preconditioner,

$$\|r_k\|_{\mathcal{Q}^{-1}}=\left[\left(Ae_k-B^Td_k, Q_A^{-1}(Ae_k-B^Td_k)\right)+(Be_k, Q_{M_p}^{-1}Be_k)\right]^{1/2}, \tag{2.8}$$

where $e_k=u-u_k$ and $d_k=p-p_k$. Although this might appear to be a problem, it is common for the Stokes equations to seek convergence in the energy norm, given in discrete form by

$$\left[(e_k,Ae_k)+(d_k,M_pd_k)\right]^{1/2}.$$

But if Q_A and Q_{M_p} are spectrally equivalent to the discrete Laplacian and mass matrix, respectively, then the residual norm of (2.8) is spectrally equivalent to the energy norm [30]. Consequently, the quantity minimized by preconditioned MINRES is a natural choice which is quasi-optimal with respect to the energy norm. Second, although we have considered only steady problems here, the same point of view can be adapted to the evolutionary Stokes equations [3, 6]. In this case, the matrix A consists of a linear combination of a velocity mass matrix and a discrete Laplace operator. Good preconditioners for the Schur complement operator require an approximate Poisson solve on the pressure space. The same considerations hold for this problem as in the discussion above, and approximations based on fast iterative solvers for the pressure Poisson equation can be used in a similar way.

3. The steady-state Navier-Stokes equations. We next consider the steady version of the Navier-Stokes equations, i.e., $\alpha = 0$ in (1.1). For solving the nonlinear system, we will restrict our attention to Picard iteration

$$\begin{aligned} -\nu\Delta\mathbf{u}^{(m+1)} + (\mathbf{u}^{(m)}\cdot\operatorname{grad})\,\mathbf{u}^{(m+1)} + \operatorname{grad}p^{(m+1)} &= \mathbf{f} \\ -\operatorname{div}\mathbf{u}^{(m+1)} &= 0, \end{aligned} \tag{3.1}$$

where the convection coefficient is lagged. For each m, this system has the form of the Oseen equations. Discretization leads to a linear system of the form

$$\begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix}\begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix} \tag{3.2}$$

to be solved at each step. Our solution strategy for this is to use Krylov subspace methods such as the GMRES [27], QMR [14] or BiCGSTAB(L) [32] algorithms, in combination with preconditioning. The latter is the critical component needed for rapid convergence.

The discussion of the previous section leads to the idea that a preconditioner should be derived using approximations to (the actions of the inverses of) F and the Schur complement $S = BF^{-1}B^T$. Before considering this in more detail, we first observe that for the symmetric problem discussed above, there is good reason to restrict attention to preconditioners with block diagonal form (2.7) in order to retain symmetry and take advantage of the short-term recurrences and optimality achieved by MINRES. However, no Krylov subspace method has both these capabilities for nonsymmetric systems, and for (3.2) we prefer a block-triangular preconditioner

$$\mathcal{Q} = \begin{pmatrix} Q_F & B^T \\ 0 & -Q_S \end{pmatrix}. \tag{3.3}$$

Iteration with this choice requires approximately half the steps needed with a block-diagonal version [8]. Applying the preconditioner, i.e., computing

$$\begin{pmatrix} w \\ s \end{pmatrix} = \begin{pmatrix} Q_F & B^T \\ 0 & -Q_S \end{pmatrix}^{-1}\begin{pmatrix} v \\ q \end{pmatrix}$$

for given v, q entails solving the systems

$$Q_S s = -q, \qquad Q_F w = v - B^T s. \tag{3.4}$$

The only cost not incurred by the block diagonal preconditioner is that of a (sparse) matrix-vector product $B^T s$, which is negligible.

A more significant difference from the Stokes equations concerns the construction of good approximations $Q_F \approx F$ and $Q_S \approx S$. The role of F in (3.2) is largely analogous to that of A above: F is a block diagonal matrix consisting of a set of d independent discrete convection-diffusion operators. Although the convection-diffusion equation is a more difficult problem than the Poisson equation (in particular, the analysis of solution algorithms is far less well developed), there are effective solvers available for it, see for example [2, 9, 24, 42]. The Schur complement system is less straightforward. An operator Q_S that is spectrally equivalent to the pressure mass matrix, as discussed above, is easy to implement and has also been shown to lead to (essentially) mesh independent rates of convergence for (3.1)–(3.2) [8, 19]. However, performance deteriorates if the Reynolds number becomes large, i.e., if the viscosity ν is small. To remedy this difficulty, we consider an alternative approach for constructing a preconditioner, which leads to a methodology that adapts in a natural way to both steady-state and evolutionary problems. This idea was originally developed by Kay and Loghin [18] using the structure of the Green's function for the operator of (3.1). The approach presented here follows Silvester et. al. [28].

We start with (3.1) and for fixed m let $\mathbf{w} = \mathbf{u}^{(m-1)}$ denote the lagged convection coefficient and $-\nu\Delta + \mathbf{w}\cdot\nabla$ the resulting convection-diffusion operator. Let us suppose that there is an analogous operator $(-\nu\Delta + \mathbf{w}\cdot\nabla)_p$ defined on the pressure space, and furthermore, that the commutator of the convection-diffusion operators with the gradient operator,

$$(-\nu\Delta + \mathbf{w}\cdot\nabla)\nabla - \nabla(-\nu\Delta + \mathbf{w}\cdot\nabla)_p\,, \tag{3.5}$$

is small in some sense. A discrete version of this relation is that

$$(M_u^{-1}F)\,(M_u^{-1}B^T) - (M_u^{-1}B^T)\,(M_p^{-1}F_p)$$

is also small, where M_u is the mass matrix associated with the velocity discretization and F_p is a discrete approximation to the convection-diffusion operator. A straightforward manipulation then yields the relation

$$BF^{-1}B^T \approx A_p F_p^{-1} M_p\,, \tag{3.6}$$

where $A_p = BM_u^{-1}B^T$ is a discrete Laplacian. That is, the matrix on the right hand side of (3.6) can be viewed as an approximation to the Schur complement operator, and this defines a preconditioning operator Q_S.

A precise definition of Q_S also requires that boundary conditions be specified for A_p and F_p. For an enclosed flow with Dirichlet boundary conditions for the velocities, the discrete Schur complement operator $BF^{-1}B^T$ is conventionally associated with a Neumann operator for the pressure field, see [28]. Therefore, A_p and F_p should correspond to discrete elliptic problems with Neumann boundary conditions. For a boundary segment with outflow boundary conditions, the Schur complement S (and its preconditioner Q_S) must be defined with Dirichlet data on that segment in order to ensure that the preconditioning operator is elliptic on the pressure space. See [34, pp. 50-51], [7, pp. 36-43] for further discussion of these points.

Note that we are not attributing any physical meaning to the convection-diffusion operator on the pressure space, and in addition, the notion of approximate commutativity is used only as motivation. The formal commutator (3.5) is zero for constant $\mathbf{w}$, but otherwise it may not be small. In addition, irrespective of what happens in the continuous case, there are examples where the discrete commutator is large, such as for div-stable finite element discretizations defined on different grids (for example, $Q_1(h)-Q_1(2h)$ consisting of bilinear velocities and bilinear pressures on macroelements). In practice, however, the preconditioning methodology is still valid, as long as the discrete operator F_p can be defined.

As shown in (3.4), use of this preconditioner requires application of the action of Q_S^{-1} to a vector. This entails a Poisson solve (to apply the action of A_p^{-1}), followed by a matrix-vector product by F_p, followed then by an application of the inverse of the mass matrix. The first and last of these steps can be replaced by inexpensive approximations as described in Section 2. That is, the action of A_p^{-1} can be replaced by an iteration such as multigrid (or one step of this process), and M_p can be replaced by its diagonal.

Our understanding of the convergence characteristics of solvers that use this preconditioner is largely based on empirical evidence. We show examples of performance for solving the lid driven cavity problem, in which the steady version of (1.1) is posed on $\Omega = (0,1)\times(0,1)$, and Dirichlet boundary conditions for the velocity $\mathbf{u} = (u_1, u_2)$ are given by

$$\begin{cases} u_1 = u_2 = 0 & \text{for } x = 0,\ x = 1 \text{ or } y = 0, \\ u_1 = 1,\ u_2 = 0 & \text{for } y = 1. \end{cases}$$

We consider two discretizations, the MAC finite difference method, and the P_2-P_1 finite element method, which uses triangular elements with piecewise quadratic bases for the velocity components and a piecewise linear basis for the pressure. In all cases, the discretization is on a uniform mesh of width h.

TABLE 3.1
Average inner iteration counts for Picard iteration, with outer iterations in parentheses.

MAC FINITE DIFFERENCES

	ν			
	1/40	1/80	1/160	1/320
$h = 1/16$	8.3 (6)	10.5 (8)	13.3 (11)	17.9 (13)
1/32	8.5 (6)	10.4 (8)	14.3 (10)	19.3 (10)
1/64	8.6 (6)	11.0 (7)	14.3 (9)	20.9 (11)
1/128	8.6 (5)	10.5 (6)	14.5 (8)	20.3 (9)

P_2-P_1 FINITE ELEMENTS

	ν			
	1/40	1/80	1/160	1/320
$h = 1/16$	8.8 (5)	11.2 (6)	14.0 (6)	23.8 (9)
1/32	8.5 (5)	10.7 (6)	13.7 (7)	20.4 (9)
1/64	8.3 (5)	10.4 (5)	13.4 (6)	18.1 (7)

We summarize the performance of preconditioned GMRES as follows:
1. *Convergence rates are independent of discretization mesh size.* Table 3.1 shows the average number of iterations needed to solve the linear systems arising during the course of a Picard iteration (3.1) for the driven cavity problem, for both discretizations. Numbers in parentheses are the number of Picard steps needed to satisfy the stopping criterion $\|F(x^{(m)})\|_2 \leq 10^{-5}\|f\|_2$, where $\mathbf{x}^{(m)} = \begin{pmatrix} u^{(m)} \\ p^{(m)} \end{pmatrix}$ and $F(\mathbf{x}^{(m)})$ is the nonlinear residual. Starting iterates were $x^{(0)} \equiv 0$ for the nonlinear system and the most recent nonlinear iterate for the linear system. The stopping criterion for the linear iteration was

$$\|r_k\|_2 \leq 10^{-2}\|F(\mathbf{x}^{(m-1)})\|_2 \,.$$

It is clear that for any fixed value of ν, the linear iteration counts are independent of h.
2. *Convergence depends mildly on the viscosity parameter ν.* Further consideration of the data of Table 3.1 shows that there is a mild increase in iteration counts as ν is decreased (i.e., as the Reynolds number is increased). The rate of increase is clearly less than linear in ν^{-1}. This issue is explored further in Figure 3.1, which plots the convergence history of the linear solvers applied to the system (3.2) that arises at the last step of the Picard iteration, for the P_2-P_1 discretization and $h = 1/64$. These results show more clearly the dependence on ν. For each ν, convergence is slow during the early stages of the iteration, and the number of steps in which this poor performance is exhibited becomes larger as ν is reduced. After these initial periods of latency, convergence becomes more rapid, and the asymptotic convergence rates of the GMRES iteration

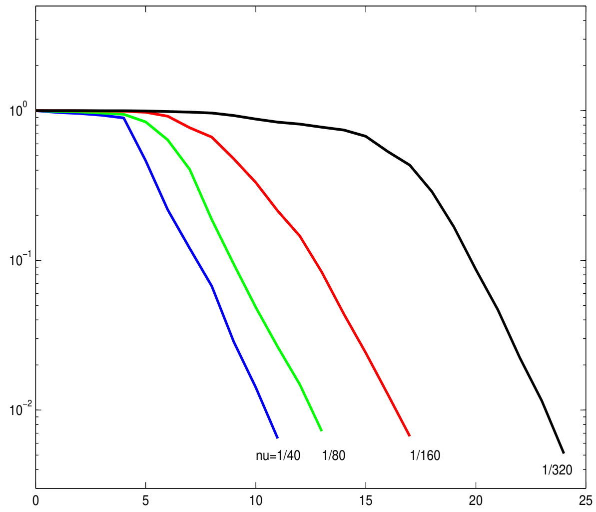

FIG. 3.1. *Convergence histories of preconditioned GMRES inner iteration for the last step of Picard iteration for several values of ν, P_2-P_1 discretization, $h = 1/64$.*

then appear to be independent of the value of the viscosity parameter.

These results are consistent with those of extensive experiments described in [13, 18, 28]. It is also shown empirically in [13] that the eigenvalues of the preconditioned matrix $\mathcal{A}\mathcal{Q}^{-1}$ are clustered in a region that does not depend on ν, except for a small number of outliers. The number of these outliers is on the order of ten, although it increases slightly as ν is decreased, and this is the cause of the latencies seen in Figure 3.1. To see this, recall that the optimality of the GMRES iteration leads to the bound on the residual norm [27]

$$\|r_k\|_2 \leq \min_{p_k(0)\in\Pi_k} \max_{\lambda\in\sigma(\mathcal{A}\mathcal{Q}^{-\infty})} |p_k(\lambda)| \, \|r_0\|_2, \tag{3.7}$$

where we have assumed that $\mathcal{A}\mathcal{Q}^{-1} = V\Lambda V^{-1}$ is diagonalizable. Let $\{\lambda_1, \ldots, \lambda_d\}$ denote the set of d outlying eigenvalues, and let $k = d + l$. Then

$$\min_{p_k(0)\in\Pi_k} \max_{\lambda\in\sigma(\mathcal{A}\mathcal{Q}^{-1})} |p_k(\lambda)| \leq \max_{\lambda\in\sigma(\mathcal{A}\mathcal{Q}^{-1})} |\phi_d(\lambda)| \, |C_l(\lambda)| \, ,$$

where

$$\phi_d(\lambda) = \left(1 - \frac{1}{\lambda_1}\lambda\right)\left(1 - \frac{1}{\lambda_2}\lambda\right)\cdots\left(1 - \frac{1}{\lambda_d}\lambda\right)$$

is the polynomial of degree d whose roots are the outliers. and C_l is any polynomial of degree l satisfying $C_l(0) = 1$. That is, convergence tends to be slow until the roots of the residual polynomial include the outliers (and the corresponding eigenvectors are deflated from the error), after which convergence will be dictated by the distribution of the remaining clustered set of eigenvalues. This distribution is independent of ν. However, since the number of outliers increases with ν^{-1}, so does the length of the latency period. Although this analysis only establishes a bound on convergence, results given in [13] demonstrate that performance is consistent with these observations.

The experiments described here are for "exact" versions of the preconditioners, i.e., no iterations or approximations have been used in place of the action of F^{-1}, A_p^{-1} or M_p^{-1}. Results in [18] indicate that there is only a small increase in the numbers of iterations when such approximations are made. In addition, preliminary experiments performed for three-dimensional examples display the same trends for both "exact" and approximate versions of the preconditioner [12].

There has been a limited amount of analysis of this preconditioning strategy that provides insight into convergence. Loghin [20] has shown that the eigenvalues of the preconditioned linear systems are contained in a set that is independent of the mesh size. This does not firmly establish that convergence rates are also independent of h, since the bound (3.7) on convergence of GMRES also depends on the condition number of the matrix of eigenvectors, but it clearly agrees with performance. Loghin has also established bounds on the relation of the eigenvalues to ν: the largest eigenvalues are bounded by ν^{-1} and the smallest ones are bounded below by ν^2. These are consistent with other analysis and experiments described in [13], where it is shown that large eigenvalues have imaginary parts that grow like ν^{-1}. However, there is no analysis establishing that the majority of eigenvalues are clustered near 1 independent of both h and ν.

Finally, observe that in the special case of the Stokes equations, where $F = A$, the preconditioner for the Schur complement as defined in (3.6) reverts to $Q_S = M_p$, i.e., it is the same choice as that discussed in Section 2. It would be straightforward to design a code that automatically handles Stokes flow by switching to a MINRES strategy in this case.

4. The evolutionary Navier-Stokes equations. We now consider how the preconditioning methodology described in Section 3 can be adapted to handle evolutionary problems. Assume that the implicit time discretization strategies entails the solution of systems of the form (3.2) at each time step. Two such approaches for discretizing in time are the backward Euler (BE) method

$$\begin{aligned} \frac{\mathbf{u}^{(n+1)} - \mathbf{u}^{(n)}}{\Delta t} - \nu\Delta\mathbf{u}^{(n+1)} + (\mathbf{u}^{(n+1)} \cdot \operatorname{grad})\, \mathbf{u}^{(n+1)} + \operatorname{grad} p^{(n+1)} &= \mathbf{f} \\ -\operatorname{div} \mathbf{u}^{(n+1)} &= 0, \end{aligned}$$

and the Crank-Nicolson (CN) method

$$\frac{\mathbf{u}^{(n+1)} - \mathbf{u}^{(n)}}{\Delta t} + \tfrac{1}{2}\left(-\nu\Delta\mathbf{u}^{(n+1)} + (\mathbf{u}^{(n+1)}\cdot \operatorname{grad})\,\mathbf{u}^{(n+1)}\right) + \operatorname{grad} p^{(n+1)}$$
$$= \mathbf{f} - \tfrac{1}{2}\left(-\nu\Delta\mathbf{u}^{(n)} + (\mathbf{u}^{(n)}\cdot \operatorname{grad})\,\mathbf{u}^{(n)}\right)$$
$$-\operatorname{div}\mathbf{u}^{(n+1)} = 0.$$

Our emphasis is not on the relative merits of these alternatives, but we note the well-known facts that BE is first order accurate in time and CN is second order accurate [16, §3.16.1]. Lack of A-stability may inhibit the utility of CN if large time steps are used [33], but it is effective for computing time-accurate solutions. From the point of view of the algorithms considered in this study, the two methods entail the same type of computations at each time step. Other time-stepping strategies that are amenable to our algorithmic approach are discussed in [34].

Both approaches as defined require the solution of a nonlinear equation at each time step, which, after spatial discretization, yields a nonlinear algebraic system. This difficulty can be avoided by suitable treatment of the coefficient $\mathbf{u}^{(n+1)}$ of the convection term. In this study, we replace this by the lagged value $\mathbf{u}^{(n)}$ for the backward Euler method, and by

$$\left(\left(\tfrac{3}{2}\mathbf{u}^{(n)} - \tfrac{1}{2}\mathbf{u}^{(n-1)}\right)\cdot \operatorname{grad}\right)\, \mathbf{u}^{(n+1)}$$

for the Crank-Nicolson method. The latter approach retains second order accuracy [31]. We will also refer to these linearized methods as BE and CN, respectively, and the experiments described below are for these strategies.

After spatial discretization, both time-stepping strategies lead to systems (3.2). F now has the form

$$F = \delta M_u + A + N, \tag{4.1}$$

where A and N are discrete diffusion and convection operators, respectively, and $\delta = \frac{1}{\Delta t}$ for BE and $\frac{2}{\Delta t}$ for CN. The vector f in the right hand side has boundary conditions and (where applicable) the explicit component of CN incorporated into it. The preconditioning operator is defined as in (3.3) and (3.6), where, by analogy with (4.1),

$$F_p = \delta M_p + A_p + N_p\,.$$

We present here two sets of experimental results, both using MAC finite differences for the spatial discretization. In Table 4.1, we show results of integrating the driven cavity problem from $t = 0$ to $t = 1$ using

TABLE 4.1
Average number of GMRES iterations per linear solve, for integration of the driven cavity problem from $t = 0$ to $t = 1$, with MAC spatial discretization and $h = 1/64$ and $1/128$.

		ν			
		1/40	1/80	1/160	1/320
h=1/64	$\Delta t = 1/8$	8.3	9.3	9.9	10.1
	1/16	6.4	7.4	7.9	8.2
	1/32	4.7	5.5	6.1	6.3
	1/64	3.4	3.8	4.2	4.5
		1/40	1/80	1/160	1/320
h=1/128	$\Delta t = 1/8$	7.9	9.4	10.3	10.9
	1/16	6.1	7.4	8.6	9.1
	1/32	4.3	5.4	6.4	7.2
	1/64	3.0	3.8	4.4	5.1

BE, for several choices of time steps, ν, and h. The table presents the average number of preconditioned GMRES iterations required during the course of the integration. The stopping criterion for the linear solves was

$$\|r_k\|_2 \leq 10^{-6}\|f\|_2$$

where the initial guess at each time step was the solution at the previous step, and zero at the first step. Comparison with Table 3.1 shows a dramatic reduction in dependence on the viscosity, especially as the time step is reduced. There is also virtually no dependence on the discretization mesh size. These results are consistent with Loghin's analysis [20].

As in Section 4, we can explore these trends further by examining the performance of the individual linear solves more closely. Figure 4.1 shows the detailed convergence behavior for examples of the inner preconditioned iterations, at time $t = 1/4$, $1/2$ and $3/4$, where $\Delta t = 1/64$ and $h = 1/64$. Here, we consider both the BE and CN time discretizations.[1] These results demonstrate that, in contrast to the steady-state case, there is now no latency associated with the inner solves, and convergence rates are insensitive to the value of the viscosity as well as the point in time at which the systems arise. The number of iterations required for CN is slightly smaller than for BE, because the larger value of δ in (4.1) can be viewed as having the effect of using a smaller time step.

Note that for evolutionary problems, the subsidiary convection-diffusion equations that must be solved at each step are time-dependent ones, which are easier to handle than in the steady-state case. The decreased

[1]The linearized CN method is not self-starting, and in these experiments the first three time steps were performed using BE.

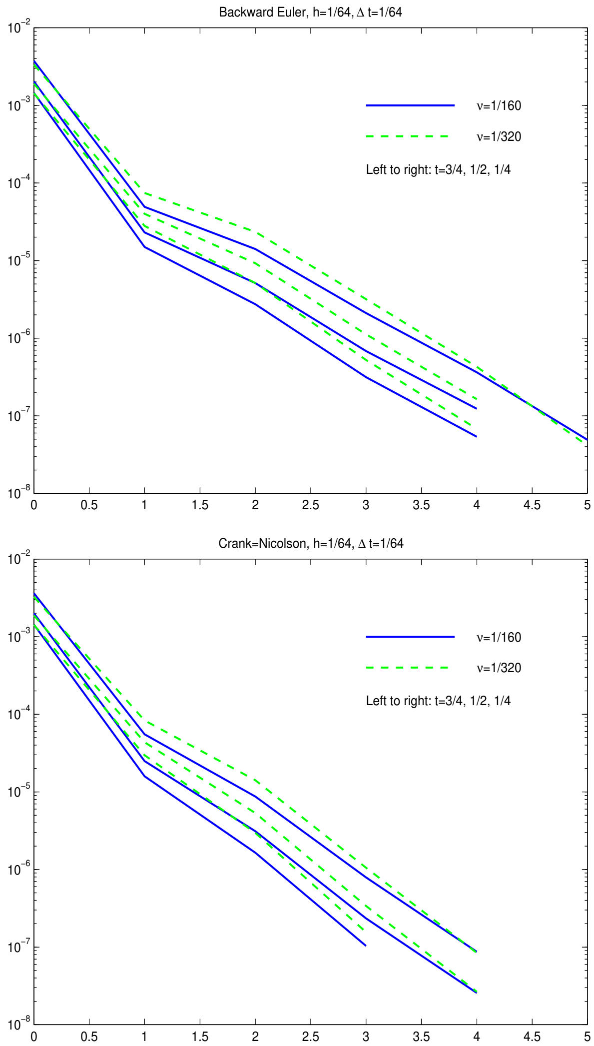

FIG. 4.1. *Convergence histories of preconditioned GMRES inner iteration at various time steps, for the MAC discretization with* $h = 1/64$ *and* $\nu = 1/160$ *and* $1/320$*.*

dependence on ν is largely due to the fact that M_u and M_p occupy increasingly dominant places in F and F_p, respectively, as the time step is decreased.

5. Comparative remarks on multigrid methods. As we have noted, an assumption underlying the utility of the preconditioning methods considered here is the potential for implementing them using building blocks for subsidiary computations. One candidate for handling the subsidiary jobs, although by no means the only one, is multigrid (MG). There are also various examples of MG methods that can be applied directly to saddle point problems, and it is natural to ask how direct application of

MG compares with the ideas discussed above.

The most popular example of multigrid methods for saddle point problems takes the form of so-called "distributive iterations," in which the MG smoothing iteration is applied to a system obtained from a change of variables [5, 35, 43, 42]. This approach shares with (3.6) the use of a discrete convection-diffusion operator F_p. Consider the transformation

$$\begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} I & B^T \\ 0 & -F_p \end{pmatrix} \begin{pmatrix} \hat{u} \\ \hat{p} \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}, \quad \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} I & B^T \\ 0 & -F_p \end{pmatrix} \begin{pmatrix} \hat{u} \\ \hat{p} \end{pmatrix}. \tag{5.1}$$

The coefficient matrix of the transformed system is

$$\tilde{\mathcal{A}} = \begin{pmatrix} F & C \\ B & BB^T \end{pmatrix}$$

where $C = FB^T - B^T F_p$ is a commutator. If C is small, then $\tilde{\mathcal{A}}$ is nearly of block triangular form, where the diagonal blocks consist of a convection-diffusion operator and a scaled discrete Laplacian. Smoothers for (5.1) are derived from smoothers for these individual blocks: see the references above for details. See also [23, 38] for other multigrid methods derived from the squared system associated with (3.2).

Thus, we see that multigrid methods share many characteristics of the preconditioning approach considered here. We have performed a comparison of these two classes of methods for solving steady-state Stokes systems, using the preconditioned MINRES method described in Section 2 [10]. This study showed that the fastest variant of MG is somewhat more efficient than preconditioned MINRES, but that there is no difference in the asymptotic behavior with respect to mesh size of the two methodoligies. We know of no direct comparison for more general Navier-Stokes systems, although we would not expect significant differences. However, in order to be useful, MG methods require the commutator to be small; they are ineffective otherwise. The approach considered here has the advantage of being more generally applicable, and it is also not explicitly dependent on use of multigrid in cases where a hierarchy of grids is unavailable.

6. Concluding remarks. We conclude by reiterating the general philosophy behind the development of the algorithms considered in this study, and we then mention some open issues. The approach described here is to derive solution algorithms for the Navier-Stokes equations by taking advantage of the saddle point structure of the linear systems that arise from standard discretizations, and to make use of algorithms for subsidiary problems such as the convection-diffusion and Poisson equations. The resulting methods have been shown to be effective, and they automatically adapt to a variety of scenarios including both steady and transient flows.

Issues that have not been fully explored include the effects of boundary conditions and of pressure discretization. In particular, even when commutativity of discrete operators "nearly" holds in the interior of Ω, it fails to hold near the boundaries, and this may have an effect on performance. More generally, the effect of the choice of boundary conditions for F_p and A_p is not understood. In addition, the preconditioning methodology depends explicitly on having the discrete operators F_p and A_p on the pressure space. Our experience with finite element discretizations has been limited to continuous pressures, where it is natural to define such operators. The question of how to handle discontinuous pressures is open.

REFERENCES

[1] K. Arrow, L. Hurwicz, and H. Uzawa. *Studies in Nonlinear Programming.* Stanford University Press, Stanford, CA, 1958.

[2] J. Bey and G. Wittum. Downwind numbering: robust multigrid for convection-diffusion problems. *Applied Numerical Mathematics*, 23:177–192, 1997.

[3] J. H. Bramble and J. E. Pasciak. Iterative techniques for time dependent Stokes problems. In W. Habashi, editor, *Solution Techniques for Large-Scale CFD Problems*, pages 201–216. John Wiley, New York, 1995.

[4] J. H. Bramble, J. E. Pasciak, and A. T. Vassilev. Analysis of the inexact Uzawa algorithm for saddle point problems. *SIAM J. Numer. Ansl*, 34:1072–1092, 1997.

[5] A. Brandt and N. Dinar. Multigrid solutions to elliptic flow problems. In S. V. Parter, editor, *Numerical Methods for Partial Differential Equations*, pages 53–147. Academic Press, New York, 1979.

[6] J. Cahouet and J.-P. Chabard. Some fast 3D finite element solvers for the generalized Stokes problem. *Int. J. Numer. Meth. Fluids*, 8:869–895, 1988.

[7] E. Dean and R. Glowinski. On some finite element methods for the numerical simulation of incompressible viscous flow. In M. D. Gunzburger and R. Y. Nicolaides, editors, *Incompressible Computational Fluid Dynamics*, pages 17–65. Cambridge University Press, New York, 1993.

[8] H. Elman and D. Silvester. Fast nonsymmetric iterations and preconditioning for Navier-Stokes equations. *SIAM J. Sci. Comput.*, 17:33–46, 1996.

[9] H. C. Elman. Relaxed and stabilized incomplete factorizations for non-self-adjoint linear systems. *BIT*, 29:890–915, 1989.

[10] H. C. Elman. Multigrid and Krylov subspace methods for the discrete Stokes equations. *Int. J. Numer. Meth. Fluids*, 227:755–770, 1996.

[11] H. C. Elman and G. H. Golub. Inexact and preconditioned Uzawa algorithms for saddle point problems. *SIAM J. Numer. Anal.*, 30:1645–1661, 1994.

[12] H. C. Elman, V. E. Howles, J. Shadid, and R. Tuminaro. In preparation, 2001.

[13] H. C. Elman, D. J. Silvester, and A. J. Wathen. Performance and Analysis of Saddle Point Preconditioners for the Discrete Steady-State Navier-Stokes Equations. Technical Report UMIACS-TR-2000-54, Institute for Advanced Computer Studies, University of Maryland, 2001. To appear in *Numer. Math.*

[14] R. Freund and N. M. Nachtigal. QMR: a quasi-minimal residual method for non-Hermitian linear systems. *Numer. Math.*, 60:315–339, 1991.

[15] V. Girault and P. A. Raviart. *Finite Element Approximation of the Navier-Stokes Equations.* Springer-Verlag, New York, 1986.

[16] P. M. Gresho and R. L. Sani. *Incompressible Flow and the Finite Element Method.* John Wiley and Sons, New York, 1998.

[17] F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids*, 8:2182–2189, 1965.

[18] D. Kay and D. Loghin. A Green's function preconditioner for the steady-state Navier-Stokes equations. Technical Report 99/06, Oxford University Computing Laboratory, 1999.

[19] A. Klawonn and G. Starke. Block triangular preconditioners for nonsymmetric saddle point problems: field-of-values analysis. *Numer. Math.*, 81:577–594, 1999.

[20] D. Loghin. Analysis of Preconditioned Picard Iterations for the Navier-Stokes Equations. Technical Report 01/10, Oxford University Computing Laboratory, 2001.

[21] R. A. Nicolaides. Analysis and convergence of the MAC scheme I. *SIAM J. Numer. Anal.*, 29:1579–1591, 1992.

[22] C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM. J. Numer. Anal.*, 12:617–629, 1975.

[23] J. Pitkäranta and T. Saarinen. A multigrid version of a simple finite element method for the Stokes problem. *Math. Comp.*, 45:1–14, 1985.

[24] A. Reusken. Convergence Analysis of a Multigrid Method for Convection-Diffusion Equations. Technical Report IGPM Report 190, RWTH Aachen, 2000.

[25] T. Rusten and R. Winther. A preconditioned iterative method for saddle point problems. *SIAM J. Matr. Anal. Appl.*, 13:887–904, 1992.

[26] Y. Saad. *Iterative Methods for Sparse Linear Systems.* PWS Publishing, Boston, 1996.

[27] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.

[28] D. Silvester, H. Elman, D. Kay, and A. Wathen. Efficient preconditioning of the linearized Navier-Stokes equations for incompressible flow. *J. Comp. Appl. Math.*, 128:261–279, 2001.

[29] D. Silvester and A. Wathen. Fast iterative solution of stabilized Stokes systems. Part II: Using block preconditioners. *SIAM J. Numer. Anal.*, 31:1352–1367, 1994.

[30] D. J. Silvester and A. J. Wathen. Fast and robust solvers for time-discretised incompressible Navier-Stokes equations. In D. F. Griffiths and G. A. Watson, editors, *Numerical Analysis: Proceedings of the 1995 Dundee Biennial Conference.* Longman, 1996. Pitman Research Notes in Mathematics Series 344.

[31] J. C. Simo and F. Armero. Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes equations. *Comp. Meths. Appl. Mech. Engrg.*, 111:111–154, 1994.

[32] G. L. G. Sleijpen and D. R. Fokkema. BICGSTAB(L) for linear equations involving unsymmetric matrices with complex spectrum. *Electr. Trans. Num. Anal.*, 6:162–181, 1997.

[33] A. Smith and D. Silvester. Implicit algorithms and their linearisation for the transient incompressible Navier-Stokes equations. *IMA J. Numer. Anal.*, 17:527–545, 1997.

[34] S. Turek. *Efficient Solvers for Incompressible Flow Problems.* Springer-Verlag, Berlin, 1999.

[35] S. P. Vanka. Block-implicit multigrid solution of Navier-Stokes in primitive variables. *J. Comput. Phys.*, 65:138–158, 1986.

[36] R. S. Varga. *Matrix Iterative Analysis.* Prentice-Hall, Englewood Cliffs, New Jersey, 1962.

[37] R. Verfürth. A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem. *IMA J. Numer. Anal.*, 4:441–455, 1984.

[38] R. Verfürth. A multilevel algorithm for mixed problems. *SIAM J. Numer. Anal.*, 21:264–271, 1984.

[39] A. Wathen and D. Silvester. Fast iterative solution of stabilized Stokes systems. Part I: Using simple diagonal preconditioners. *SIAM J. Numer. Anal.*, 30:630–649, 1993.

[40] A. J. Wathen. Realistic eigenvalue bounds for the Galerkin mass matrix. *IMA J. Numer. Anal.*, 7:449–457, 1987.

[41] B. D. Welfert. Convergence of Inexact Uzawa Algorithms for Saddle Point Problems. Technical report, Mathematics Department, University of Arizona, 1993.

[42] P. Wesseling. *An Introduction to Multigrid Methods.* John Wiley & Sons, New York, 1992.

[43] G. Wittum. Multi-grid methods for the Stokes and Navier-Stokes equations. *Numer. Math.*, 54:543–564, 1989.