INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2002; **40**:333–344 (DOI: 10.1002/fld.311)

Block preconditioners for the discrete incompressible Navier–Stokes equations

Howard C. Elman[1,*,†], David J. Silvester[2,‡] and Andrew J. Wathen[3,§]

[1]*Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, U.S.A.*
[2]*Department of Mathematics, University of Manchester Institute of Science and Technology, Manchester M601QD, U.K.*
[3]*Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford OX13QD, U.K.*

SUMMARY

We examine the convergence characteristics of iterative methods based on a new preconditioning operator for solving the linear systems arising from discretization and linearization of the steady-state Navier–Stokes equations. For steady-state problems, we show that the preconditioned problem has an eigenvalue distribution consisting of a tightly clustered set together with a small number of outliers. These characteristics are directly correlated with the convergence properties of iterative solvers, with convergence rates independent of mesh size and only mildly dependent on viscosity. For evolutionary problems, we show that implicit treatment of the time derivatives leads to systems for which convergence is essentially independent of viscosity. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: Navier–Stokes equations; preconditioning; iterative algorithms

1. INTRODUCTION

We study the performance of a preconditioning methodology designed for use with Krylov subspace iteration to compute the numerical solution of the incompressible Navier–Stokes equations

$$\begin{aligned} \alpha \mathbf{u}_t - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \operatorname{grad})\mathbf{u} + \operatorname{grad} p &= \mathbf{f}, \\ -\operatorname{div} \mathbf{u} &= 0, \end{aligned} \quad \text{in } \Omega \tag{1}$$

*Correspondence to: H. C. Elman, Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, U.S.A.
†E-mail: elman@cs.umd.edu
‡E-mail: djs@fire.ma.umist.ac.uk
§E-mail: andy.wathen@comlab.ox.ac.uk

Contract/grant sponsor: U.S. National Science Foundation; contract/grant number: DMS9972490
Contract/grant sponsor: Engineering and Physical Sciences Research Council visiting fellowship; contract/grant number: GR/N25565
Contract/grant sponsor: Engineering and Physical Sciences Research Council; contract/grant number: GR/M59044

Received May 2001
Revised October 2001

subject to suitable boundary conditions on $\partial\Omega$, where Ω is an open bounded domain in $\mathbb{R}^2$ or $\mathbb{R}^3$, $\mathbf{u}$ is the velocity and p is the pressure. The cases $\alpha = 0$ and 1 correspond to steady-state and evolutionary problems, respectively. In both cases we will consider a non-linear iteration in which the convection coefficient is lagged but all other terms are treated implicitly. This is equivalent to the linearization of (1)

$$\begin{aligned} \alpha \mathbf{u}_t - \nu \Delta \mathbf{u} + (\mathbf{w} \cdot \operatorname{grad}) \mathbf{u} + \operatorname{grad} p &= \mathbf{f} \\ -\operatorname{div} \mathbf{u} &= 0 \end{aligned} \tag{2}$$

where $\mathbf{w}$ is such that $\operatorname{div} \mathbf{w} = 0$. For steady-state problems, this system arises from a Picard iteration for (1), for transient problems, we will treat the time derivative using a backward Euler discretization.

Discretization in space using a div-stable strategy [1] leads to a linear system

$$\begin{pmatrix} F & B^{\mathrm{T}} \\ B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix} \tag{3}$$

F is a discrete convection–diffusion operator, i.e. it has the form $F = \alpha(1/\Delta t)M + \nu A + N$ where A is a discrete diffusion operator, N is a discrete convection operator, M is mass matrix defined on the velocity space, and B and B^{T} are discrete divergence and gradient operators, respectively.

Let $\mathscr{A}$ denote the coefficient matrix of (3). For a preconditioner, we consider an operator of the form

$$\mathscr{P}_{\mathscr{A}} = \begin{pmatrix} P_F & B^{\mathrm{T}} \\ 0 & -P_S \end{pmatrix} \tag{4}$$

designed to be applied with right orientation. P_F^{-1} approximates the action of the inverse of the discrete convection–diffusion operator F, and P_S^{-1} approximates the action of the inverse of the Schur complement operator $S = BF^{-1}B^{\mathrm{T}}$. In the special case $P_F = F$ and $P_S = S$, the preconditioned matrix is

$$\mathscr{A}\mathscr{P}_{\mathscr{A}}^{-1} = \begin{pmatrix} I & 0 \\ BF^{-1} & I \end{pmatrix}$$

use of this preconditioner with GMRES iteration [2] would require precisely two steps to compute the exact solution [3]. The advantage of this general approach is that, it reduces the problem of finding preconditioners for (3) to that of finding good techniques defining P_F^{-1} and P_S^{-1}.

Since F is a discrete convection–diffusion operator, there is a natural way to define P_F^{-1}: the action of F^{-1} can be approximated using any appropriate iterative method (for example, multigrid) for solving the convection–diffusion equation. We will discuss some practical issues concerning this point in Section 5. In our experience, the choice of the operator P_S is more critical, and we simplify our study here by using $P_F = F$. For P_S, the approximation to the Schur complement, we use

$$P_S = A_p F_p^{-1} M_p \tag{5}$$

Table I. Iteration counts for convergence of the preconditioned GMRES solver.

	ν				
	1/20	1/40	1/80	1/160	1/320
MAC finite differences					
$h = 1/16$	17	20	24	31	39
1/32	17	21	26	33	43
1/64	16	21	26	34	44
1/128	16	20	26	33	45
$P_2 - P_1$ finite elements					
$h = 1/16$	18	21	26	37	62
1/32	17	21	26	33	47
1/64	17	20	25	33	43

where A_p and F_p represent discrete approximations to scaled Laplacian and convection–diffusion operators defined on the pressure space, and M_p is the pressure mass matrix. This idea was first proposed by Kay and Loghin [4] and considered further in Reference [5]. We refer to the operator of (5) as the *F_p-preconditioner*. This report is a summary and extension of results described in detail in Reference [6].

2. THE LINEARIZED STEADY-STATE PROBLEM

We begin with some experimental results for solving a linearized version of the driven cavity problem, in which (2) is posed on $\Omega = (0,1) \times (0,1)$, with $\mathbf{f} = 0$, boundary conditions

$$u_1 = u_2 = 0, \quad \text{for } x = 0,\ x = 1 \text{ or } y = 0$$
$$u_1 = 1, \quad u_2 = 0, \quad \text{for } y = 1$$

and convection coefficients (or 'wind')

$$\mathbf{w} = \begin{pmatrix} 2(2y-1)(1-(2x-1)^2) \\ -2(2x-1)(1-(2y-1)^2) \end{pmatrix}$$

This choice of the wind, which contains a single recirculation in Ω, is used to simulate what happens in the Picard iteration for the driven cavity problem, where the solution contains one primary recirculation. We examine two div-stable discretizations, marker-and-cell (MAC) finite differences [7, 8], and $P_2 - P_1$ finite elements [1] which uses triangular elements with piecewise quadratic bases for the velocity components and a piecewise linear basis for the pressure. We discretize (2) in Ω using a uniform mesh of width h. A_p and F_p are defined on the discrete pressure spaces in the natural way via finite differences for MAC and linear elements for $P_2 - P_1$. The mass matrices for the MAC discretization have the form $h^2 I$.

Table I shows the number of iterations required by the preconditioned GMRES solver, for both discretizations and a variety of values of the mesh size h and viscosity parameter ν. (All

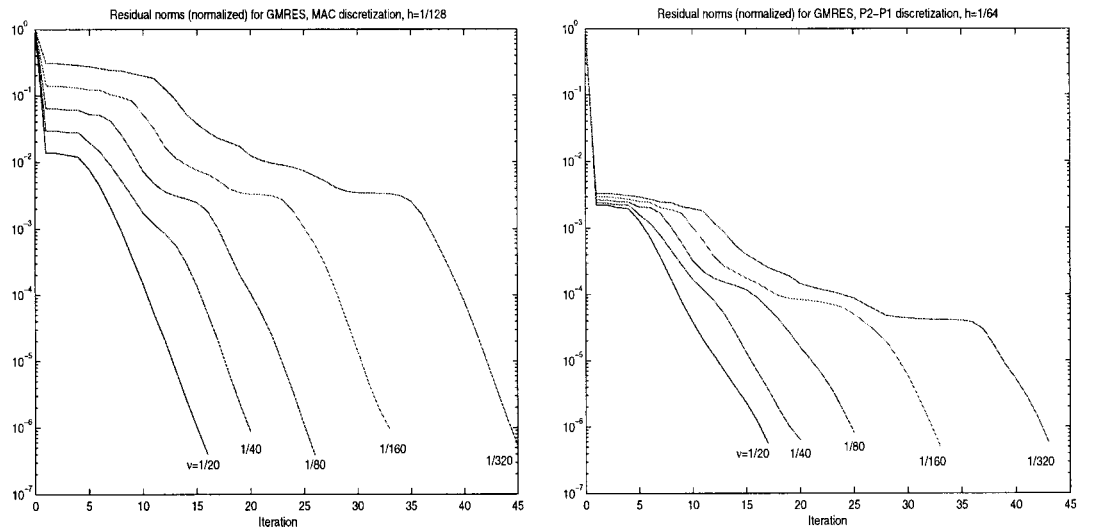


Figure 1. Complete record of $\|r_k\|_2/\|r_0\|_2$ generated by GMRES, for finite differences with $h = 1/128$ (left) and $P_2 - P_1$ finite elements with $h = 1/64$ (right).

computations were performed using Matlab on a Sun Sparc Ultra 1 computer.) The initial guess was identically zero, and the stopping criterion was $(\|r_k\|_2/\|r_0\|_2) < 10^{-6}$, where r_k is the residual vector. These experiments clearly show that convergence is essentially independent of the discretization parameter but that there is modest dependence on the viscosity, with reductions in ν (i.e. increases in the Reynolds number) leading to increases in iteration counts. Figure 1 shows the details of convergence histories for some of these entries, corresponding to fixed h and varying ν. The figures suggest that in fact the asymptotic convergence behaviour of the GMRES iteration is also independent of ν, but that there is a period of slow convergence in the early stages of the iteration, and this latency period is longer for smaller values of ν.

Remark

The behaviour exhibited here is similar to that observed by Kay and Loghin [4] in a wide variety of examples, including pipe flow, flow past a cylinder, and flow over a backward facing step. Moreover, although we only consider div-stable discretizations here, the methodology also applies to stabilized schemes, as long as suitable operators A_p and F_p are available for (5).

We next show that this convergence performance is correlated with the eigenvalues of the coefficient matrix. Let the preconditioned system be denoted

$$\mathcal{A}\boldsymbol{x} = \boldsymbol{f}$$

(so that $\mathcal{A} = \mathcal{A}\mathcal{P}_{\mathcal{A}}^{-1}$), and let the residual associated with an iterate $\boldsymbol{x}_m$ be given by

$$\boldsymbol{r}_m = \boldsymbol{f} - \mathcal{A}\boldsymbol{x}_m$$

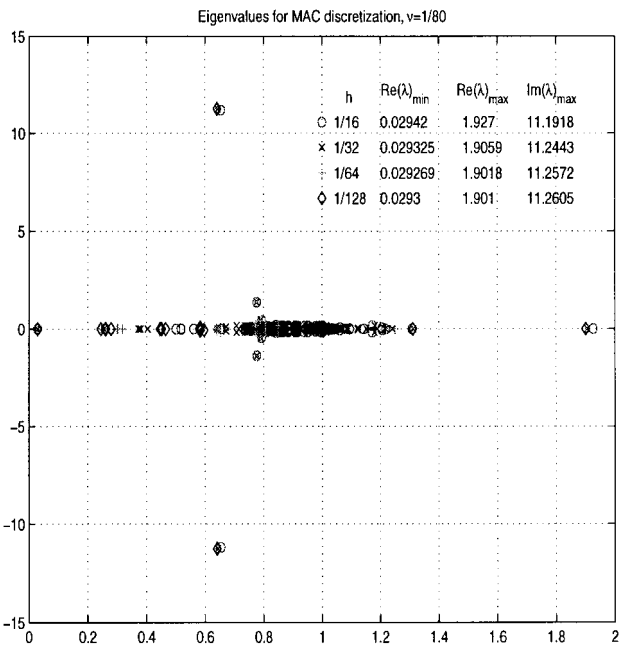


Figure 2. Dependence of eigenvalues on mesh size for the MAC discretization, $v = 1/80$.

Assume $\mathscr{A} = V\Lambda V^{-1}$ is diagonalizable, and let $\sigma(\mathscr{A})$ denote the set of eigenvalues of $\mathscr{A}$. Recall the standard bound on convergence of GMRES [2]

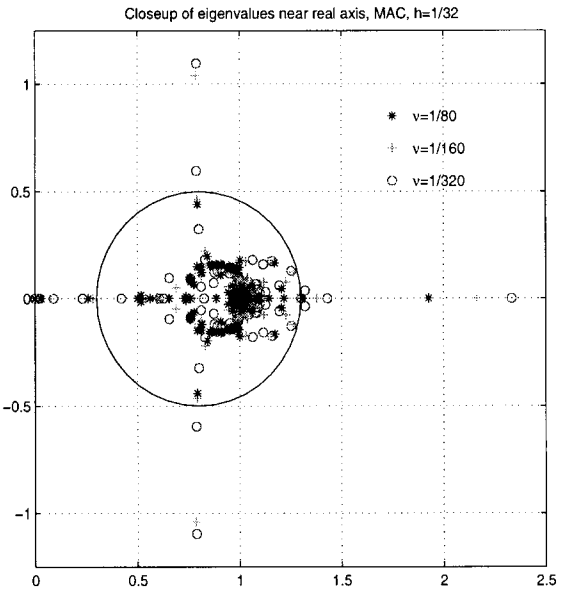
$$\|\boldsymbol{r}_m\|_2 \leqslant \kappa(V) \min_{p_m(0)=1} \max_{\lambda \in \sigma(\mathscr{A})} |p_m(\lambda)| \, \|\boldsymbol{r}_0\|_2 \tag{6}$$

where the minimum is over all polynomials of degree m taking on the value 1 at the origin, and $\kappa(V) = \|V\|_2 \, \|V^{-1}\|_2$ is the condition number of the matrix of eigenvectors.

To use this result to explain the behaviour of the GMRES iteration, we examine the eigenvalues of the F_p-preconditioned matrices SP_S^{-1}. For brevity, we only present the results for the MAC discretization, the trends for $P_2 - P_1$ finite elements are similar and can be found in Reference [6]. Figure 2 plots eigenvalues for $v = 1/80$ and four successively refined meshes. For $h = 1/16$ and $1/32$, the figure shows all eigenvalues; for $h \leqslant 1/64$ and $1/128$, only a subset consisting of extremal eigenvalues are shown. These data show that the extremal eigenvalues display no significant dependence on h, indeed, the tabulated results included in the figure show that the maximum real parts, maximum imaginary parts, and minimum real parts of all eigenvalues are virtually identical for all four mesh parameters. In addition, for the examples in which all eigenvalues have been computed ($h \geqslant 1/32$), most of them with the exception of a few outliers lie in a tightly clustered set close to 1. Table II explores these trends further by identifying the dependence of the outliers on the viscosity parameter v. It shows the extreme real and imaginary parts, as well as the three next smallest real parts, for $h = 1/128$. We see that the largest imaginary part is increasing in proportion to v^{-1}, and for small enough v, the smallest real part is decreasing in proportion to v^2. A similar statement applies for the other small real parts, although the asymptotic behaviour is evident only for somewhat smaller

Table II. Dependence of eigenvalues on ν for the MAC discretization and $h = 1/128$.

ν	Extremal real and imaginary parts			Other small real parts		
	$\Re(\lambda)_{\min}$	$\Re(\lambda)_{\max}$	$\Im(\lambda)_{\max}$	λ_2	λ_3	λ_4
1/20	0.2402	1.2017	2.8015	0.2410	0.2411	0.3323
1/40	0.1073	1.5389	5.6224	0.2486	0.2495	0.2553
1/80	0.0293	1.9010	11.2605	0.2444	0.2597	0.2608
1/160	0.0075	2.1265	22.5330	0.0786	0.2635	0.2747
1/320	0.0019	2.3008	45.0749	0.0209	0.0860	0.2158
1/640	0.0005	2.4403	90.1561	0.0053	0.0230	0.1388



Circle, center .8
radius .5

Figure 3. Containment of eigenvalues in a clustered set, MAC discretization, $h = 1/32$.

values of ν. That is, there are some eigenvalues that are decreasing in proportion to ν^2, but the set (of indices of eigenvalues) for which this pattern holds appears to be small.

Additional insight is obtained from Figure 3, which shows the clustered eigenvalues in the case $h = 1/32$ for various values of ν. In contrast to Figure 2, the axes here are in proportion and some of the eigenvalues with large imaginary part are not shown. The circle centred at $(0.8, 0)$ with radius $1/2$ represents a domain, determined by inspection, containing the majority of eigenvalues. Table III continues in this direction by showing how many eigenvalues lie outside this circle for different values of ν, for the two mesh parameters for which all eigenvalues have been computed. These data indicate that in fact the number of outliers is small but growing slightly with ν^{-1}. The number of eigenvalues with small real parts (i.e. to the left of the circle) is also increasing as ν gets smaller, but quite a bit more slowly than the total number of outliers.

Table III. Number of eigenvalues outside circle, for MAC discretization and $h = 1/16, 1/32$.

ν	No. eigenvalues outside circle		No. real eigens left of circle	
	1/16	1/32	1/16	1/32
1/40	7	7	1	1
1/80	8	8	2	2
1/160	11	12	3	3
1/320	19	21	4	4
1/640	25	29	6	6

These observations can be combined with (6) to explain the convergence behaviour of the GMRES iteration, the approach is derived from Reference [9]. Let $\sigma(\mathscr{A}) = \sigma_c(\mathscr{A}) \cup \sigma_0(\mathscr{A})$ where $\sigma_c(\mathscr{A})$ denotes the clustered set of eigenvalues of $\mathscr{A}$ and $\sigma_0(\mathscr{A}) = \{\lambda_1, \lambda_2, \ldots, \lambda_d\}$ denotes the set of d outliers. Taking $m = d + k$ in (6), we have

$$\min_{p_{k+d}(0)=1} \max_{\lambda \in \sigma(\mathscr{A})} |p_{k+d}(\lambda)| \leqslant \max_{\lambda \in \sigma_c(\mathscr{A})} |\phi_d(\lambda)|\, |C_k(\lambda)| \tag{7}$$

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 outlying eigenvalues contained in $\sigma_0(\mathscr{A})$, and C_k is any polynomial of degree k satisfying $C_k(0) = 1$. C_k can be chosen to be small on a set $\mathscr{E}$ containing $\sigma_c(\mathscr{A})$ such that

$$\max_{\lambda \in \sigma_c(\mathscr{A})} |C_k(\lambda)| \approx \rho^k$$

If the enclosing set $\mathscr{E}$ is bounded by a circle centred at c with radius r as above, then the choice $C_k(\lambda) = ((c - \lambda)/c)^k$ is optimal with respect to L_∞ on $\mathscr{E}$ [10, p. 90]. For this choice, $\rho = r/c$, and it follows from (6) and (7) that

$$\|\boldsymbol{r}_m\|_2 \leqslant \kappa(V) \max_{\lambda \in \sigma_c(\mathscr{A})} |\phi_d(\lambda)|\, \rho^k \|\boldsymbol{r}_0\|_2$$

This bound suggests that there will be a latency of d steps before the asymptotic convergence behaviour is observed. It is also demonstrated in Reference [6] that this is an accurate depiction of what the GMRES iteration does: the initial stages of the iteration construct the polynomial ϕ_d and after this, a faster asymptotic convergence rate is seen.

Note that any effects of $\kappa(V)$ are not included in this discussion. We have calculated this quantity for various choices of h and ν and found it to have values on the order of 10^2–10^3, but it exhibits no discernible pattern. We also computed eigenvalues of perturbed versions of $\mathscr{A}$, which give an indication of the ε-*pseudospectrum* of $\mathscr{A}$. In these computations, we found that only certain eigenvalues lying inside the clustered region are sensitive to perturbation, and the perturbations remain inside the clustered region. This helps explain why the analysis above characterizes convergence behaviour [8].

Table IV. Average inner iteration counts for Picard iteration, with outer iterations in parentheses, for $h = 1/64$.

	v				
	1/20	1/40	1/80	1/160	1/320
MAC	6.5 (4)	9.0 (5)	11.6 (7)	15.6 (9)	22.5 (11)
$P_2 - P_1$	6.8 (4)	8.3 (4)	9.6 (5)	13.2 (6)	18.1 (7)

3. PICARD ITERATION FOR THE NON-LINEAR STEADY-STATE PROBLEM

The results above are for a fixed velocity field $\mathbf{w}$ in the steady-state Oseen equations (2) where, in the experiments, $\mathbf{w}$ was chosen to resemble the driven cavity flow field. The fixed wind was used for convenience but it is somewhat artificial, since the structure of the convection coefficient $\mathbf{w}$ in a non-linear iteration will depend on the viscosity v and the stage of the iteration in which the linearized system (2) appears. We now show the results of some experiments with the full non-linear Picard iteration

$$\begin{aligned} -v\Delta\mathbf{u}^{(m)} + (\mathbf{u}^{(m-1)} \cdot \operatorname{grad})\,\mathbf{u}^{(m)} + \operatorname{grad} p^{(m)} &= \mathbf{f} \\ -\operatorname{div}\mathbf{u}^{(m)} &= 0 \end{aligned} \tag{8}$$

for both discretizations and $h = 1/64$.

Table IV shows the average iteration counts required when preconditioned GMRES is used to solve each linear system arising during the non-linear iteration. For completeness, the number of Picard iterations is also shown, in parentheses. These tests were run using an 'inexact' non-linear iteration (8), with the stopping criterion for the linear solver tied to the residual of the non-linear system. That is, the linear iteration was stopped at step k when the linear residual vector r_k satisfied

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

where

$$\mathbf{x}^{(m-1)} = \begin{pmatrix} \mathbf{u}^{(m-1)} \\ p^{(m-1)} \end{pmatrix}$$

and

$$F(\mathbf{x}^{(m-1)})$$

is the non-linear residual. The starting iterates were $\mathbf{x}^{(0)} \equiv 0$ for the non-linear system and the most recent non-linear iterate for the linear system.

These results are consistent with what we observed for a fixed velocity field. In particular, the dependence on v shown in Table IV is essentially the same as that depicted in Table I. Similar results are also given in Reference [4]. The lower iteration counts are due to the less stringent stopping criterion. Further comparison is provided by Figure 4, which shows the complete convergence histories of the linear solves for one problem, with $v = 1/160$ and

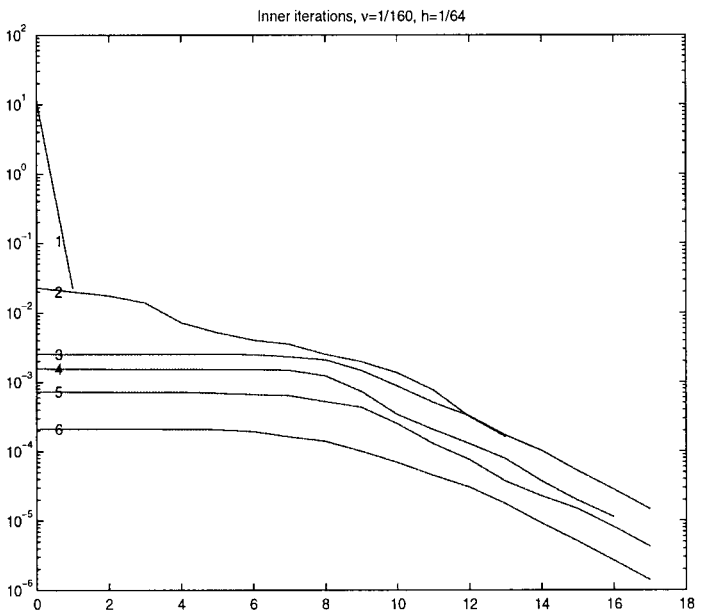


Figure 4. Convergence histories of preconditioned GMRES inner iteration for the full Picard iteration with $v = 1/160$, $P_2 - P_1$ discretization, $h = 1/64$. The curve labelled 'i' shows the residual norms for the inner iteration at the ith Picard step.

the $P_2 - P_1$ discretization. This data is typical of the results for all the problems. Except in the first non-linear step, which requires a Stokes solve, there is a latency exactly like that observed above.

4. TRANSIENT PROBLEMS

We next show that the methodology described in Section 1 is directly applicable to transient problems ($\alpha = 1$ in (1)), and the sensitivity to viscosity observed for steady-state problems is eliminated.

For time discretization, we consider the *linearized backward Euler* scheme

$$\frac{\mathbf{u}^{(m+1)} - \mathbf{u}^{(m)}}{\Delta t} - v\Delta\mathbf{u}^{(m+1)} + (\mathbf{u}^{(m)} \cdot \text{grad})\,\mathbf{u}^{(m+1)} + \text{grad}\; p^{(m+1)} = \mathbf{f}$$

$$-\text{div}\,\mathbf{u}^{(m+1)} = 0$$

This is a fully implicit discretization except the convection coefficient is evaluated at the previous time step. It is first-order accurate in time and unconditionally stable [12]. Spatial discretization then leads to a linear system of the form (3) to be solved at each time step, where $F = (1/\Delta t)M + vA + N$ corresponds to discretization of a time-dependent convection–

Table V. Average number of GMRES iterations per linear solve, for integration of the driven cavity problem from $t=0$ to 1, with MAC spatial discretization and various combinations of parameter.

		1/40	1/80	1/160	1/320
$h=1/32$	$\Delta t=1/8$	7.9	8.4	8.6	8.5
	1/16	6.4	6.7	6.8	6.6
	1/32	4.7	5.0	5.1	5.0
	1/64	3.4	3.5	3.6	3.8
$h=1/64$	$\Delta t=1/8$	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
$h=1/128$	$\Delta t=1/8$	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

diffusion operator.[‖] As a preconditioner, we use (4) with $P_F = F$ and P_S as in (5), where $F_p = (1/\Delta t)M_p + \nu A_p + N_p$ is defined on the discrete pressure space.

Experimental results for this example are shown in Table V. In all the tests, we integrated the driven cavity problem from $t=0$ to 1, starting from an initial condition $\mathbf{u}(\mathbf{x},0)=0$. We used four different time steps, for several values of ν and the discretization mesh size h. The MAC spatial discretization was used; similar results for $P_2 - P_1$ finite elements were shown in Reference [5]. The table shows the average number of iterations for these inner solves, taken over all time steps. A stringent stopping criterion ($\|r_k\|_2/\|r_0\|_2 < 10^{-6}$ was used for every linear system solve. We did not examine accuracy with respect to the spatial or time discretizations.

The main observation to be made from these data is the reduced dependence on viscosity of performance of the linear solves. One way to see this clearly is to view the data across the diagonal in the table corresponding to $h=1/128$. Here, the influence of spatial accuracy is minimal and Δt is reduced in proportion to ν. In this case, the iteration counts are then independent of ν. An analogous conclusion is reached if Δt is fixed and the horizontal rows are considered, as long as the time step is small enough. We do note that there now appears to be some new dependence on the discretization mesh size not seen previously; however, the spatial dependence is actually less pronounced as the mesh is refined and our speculation is that the asymptotic behaviour as $h \to 0$ will not depend on h.

[‖]Many other time discretizations, for example, variants of the Crank–Nicolson scheme [12], lead to such systems. Our only goal is to demonstrate the general utility of the preconditioning methodology and we will not examine any issues relating to time discretization here.

5. PRACTICAL IMPLEMENTATION

We conclude by discussing some issues that would arise in a practical implementation of these ideas. First, a precise specification of the preconditioner P_S of (5) requires that boundary conditions be defined for the discrete operators A_p and F_p. For the case of an enclosed flow with specified velocity conditions on $\partial\Omega$, the discrete Schur complement operator $BF^{-1}B^{\mathrm{T}}$ is conventionally associated with a Neumann operator for the pressure field, see Reference [5]. This means that A_p and F_p should correspond to discrete elliptic problems with a standard Neumann boundary condition. In the case of a boundary segment with standard outflow boundary conditions, the Schur complement S (and its preconditioner P_S) must be defined with Dirichlet data for the pressure on that part of the boundary in order to ensure that the preconditioning operator is elliptic over the pressure solution space. See Reference [13, pp. 50–51], [14, pp. 36–43] for further discussion of these points.

Next, we discuss the algorithmic requirements of using the ideas presented here. As we observed in Section 1, one requirement is an efficient method for solving (or approximating the solution to) the convection–diffusion equation. Similarly, $P_S^{-1} = M_p^{-1} F_p A_p^{-1}$, which means that applying this operator requires a Poisson solve (for A_p^{-1}) and the solution of a system of equations with the mass matrix (for M_p^{-1}). The latter computation has negligible cost, and fast algorithms such as multigrid or domain decomposition can be used for the convection–diffusion and Poisson solves. Kay and Loghin [4] showed that use of such inner iterations for steady-state problems leads to performance consistent with that of the 'exact' preconditioner. For evolutionary problems, the presence of the mass matrix in F will make the resulting convection–diffusion subproblems easier to handle. Besides these requirements, the only others entail efficient implementation of the operations used by Krylov subspace solvers such as matrix–vector products and inner products. The key point is that a complete solver with low cost per iteration can be constructed in a modular fashion using more basic building blocks designed for the subsidiary computations. Finally, although we have restricted our attention to GMRES for the Krylov subspace method, we expect other solvers to perform well also.

ACKNOWLEDGEMENTS

H. C. Elman was supported in part by the U.S. National Science Foundation under Grant DMS9972490, and in part by the Oxford University Computing Laboratory. D. J. Silvester was supported in part by the Engineering and Physical Sciences Research Council Visiting Fellowship Grant GR/N25565. A. J. Wathen was supported in part by the Engineering and Physical Sciences Research Council under Grant GR/M59044.

REFERENCES

1. Girault V, Raviart PA. *Finite Element Approximation of the Navier–Stokes Equations*. Springer: New York, 1986.
2. Saad Y, Schultz MH. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *Society for Industrial and Applied Mathematics Journal on Scientific and Statistical Computing* 1986; **7**:856–869.
3. Murphy MF, Golub GH, Wathen AJ. A note on preconditioning for indefinite linear systems. *Society for Industrial and Applied Mathematics Journal on Scientific Computing* 2000; **21**:1969–1972.
4. Kay D, Loghin D. A Green's function preconditioner for the steady-state Navier–Stokes equations. *Technical Report* 99/06, Oxford University Computing Laboratory, 1999.

5. Silvester D, Elman H, Kay D, Wathen A. Efficient preconditioning of the linearized Navier–Stokes equations for incompressible flow. *Journal of Computational and Applied Mathematics* 2001; **128**:261–179.
6. Elman HC, Silvester DJ, Wathen AJ. Performance and analysis of saddle point preconditioners for the discrete steady-state Navier–Stokes equations. *Numerische Mathematik* 2002; **90**(4):641–664.
7. Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids* 1965; **8**:2182–2189.
8. Nicolaides RA. Analysis and convergence of the MAC scheme I. *SIAM Journal on Numerical Analysis* 1992; **29**:1579–1591.
9. Jennings A. Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method. *Journal of the Institute of Mathematics and its Applications* 1977; **20**:61–72.
10. Rivlin T. *Chebyshev Polynomials*: *From Approximation Theory to Algebra and Number Theory*. (2nd edn). Wiley: New York, 1990.
11. Nachtigal NM, Reichel L, Trefethen LN. A hybrid GMRES algorithm for nonsymmetric linear systems. *SIAM Journal on Matrix Analysis and Applications* 1992; **13**:796–825.
12. Simo JC, Armero F. Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier–Stokes equations. *Journal of Computational and Applied Mathematics* 1994; **111**: 111–154.
13. Turek S. *Efficient Solvers for Incompressible Flow Problems*. Springer: Berlin, 1999.
14. Dean E, Glowinski R. On some finite element methods for the numerical simulation of incompressible viscous flow. In *Incompressible Computational Fluid Dynamics*, Gunzburger, MD Nicolaides RY (eds). Cambridge University Press: New York, 1993; 17–65.

