INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2005; **47**:1277–1283
Published online 27 January 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.874

Efficient preconditioning of the discrete adjoint equations for the incompressible Navier–Stokes equations

René Schneider[*,†] and Peter K. Jimack

School of Computing, The University of Leeds, LS2 9JT Leeds, U.K.

SUMMARY

Preconditioning of the discrete adjoint equations is closely related to preconditioning the linear systems arising in the Newton linearization of the discretized flow equations. We investigate the use of an optimal preconditioner for both problems on the example of a finite element discretization of the steady state incompressible Navier–Stokes equations. It is demonstrated that complications arising from the use of a zero mean pressure condition in the problem formulation can be overcome by modifying the preconditioner suitably. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: Navier–Stokes; preconditioner; discrete adjoint; finite element method

1. INTRODUCTION

Shape optimization and parameter sensitivity analysis are areas of major interest in CFD due to their importance in the design process for a wide range of CFD applications. Efficient methods for computing the derivatives with respect to problem parameters have been developed and applied in a number of applications, including the solution of fluid flow problems. One particular example of this type is the discrete adjoint method (e.g. Reference [1]). In order to compute the sensitivity of a performance function $I(\varphi,\mathscr{F})$ to a shape parameter vector $\mathscr{F}$ a system of the form

$$J^{\mathrm{T}}\psi = \frac{\partial I}{\partial \varphi} \tag{1}$$

must be solved, where J is the Jacobian $J := \partial R/\partial\varphi$ of the residual $R = R(\varphi,\mathscr{F})$ of the discrete flow equations with respect to the flow solution φ (comprising the relevant flow variables). Both R and I are functions of φ and the shape parameter vector $\mathscr{F}$. One particular advantage

[*] Correspondence to: René Schneider, School of Computing, The University of Leeds, LS2 9JT Leeds, U.K.
[†] E-mail: rschneid@comp.leeds.ac.uk

Received 27 April 2004
Revised 26 October 2004
Accepted 28 October 2004

of the discrete adjoint method is that the Jacobian matrix J is readily available if the flow solver uses the Newton method to linearize the flow equations $R=0$.

A further advantage of the discrete adjoint method is that, since the transpose J^{T} of a matrix J has the same eigenvalues, once an efficient preconditioner for J is known, it should be possible to reuse this for the adjoint problem. In the case of the finite element discretization of the steady state incompressible Navier–Stokes equations efficient preconditioning techniques for the linear systems arising in the application of the Picard iteration and the Newton method have been developed, e.g. in References [2, 3], respectively. The preconditioner described in these papers is optimal in the sense that under certain assumptions the eigenvalues of the preconditioned system are bounded asymptotically independent of the mesh size parameter h. We briefly review some of the most important issues associated with the application of these preconditioning techniques in Section 2.

Despite the fact that the spectra of J and J^{T} are identical, it turns out that simply taking the transpose of the preconditioner used for J is not generally sufficient when solving the adjoint problem. In Section 3 we illustrate this and propose a modification for the preconditioner of the adjoint system which restores the optimal behaviour. Numerical results are presented in Section 4.

2. PRECONDITIONING THE LINEAR SYSTEMS IN A FINITE ELEMENT FLOW SOLVER

A finite element discretization of the Navier–Stokes equations can be formulated as follows (see for example Reference [4]):

Find $u\in(V_h)^d$ and $p\in Q_h^0$ which fulfil

$$\begin{aligned} a(u,v)+c(u,u,v)+b(v,p)&=0 \quad \forall v\in(V_{h0})^d\\ b(u,q)&=0 \quad \forall q\in Q_h^0 \end{aligned}$$

and the appropriate Dirichlet boundary conditions for the velocity u, where d is the spatial dimension of the domain Ω ($d=2$ throughout this paper), V_h and Q_h are the finite element function spaces of velocity components and pressure, respectively,

$$\begin{gathered} a(u,v):=\frac{1}{Re}\int_\Omega \nabla u:\nabla v\,\mathrm{d}\Omega,\quad b(u,p):=\int_\Omega p(\nabla\cdot u)\,\mathrm{d}\Omega,\quad c(u,v,w):=\int_\Omega u\cdot\nabla v\cdot w\,\mathrm{d}\Omega,\\ Q_h^0:=\left\{q\in Q_h:\int_\Omega q\,\mathrm{d}\Omega=0\right\},\quad V_{h0}:=\{v\in V_h: v=0 \text{ on } \partial\Omega\} \end{gathered} \tag{2}$$

and Re is the Reynolds number.

Two commonly used iterative solution techniques (linearizations) for this nonlinear problem are described by

$$\begin{aligned} a(u^{k+1},v)+\sigma c(u^{k+1},u^k,v)+c(u^k,u^{k+1},v)+b(v,p^{k+1})&=\sigma c(u^k,u^k,v) \quad \forall v\in(V_{h0})^d\\ b(u^{k+1},q)&=0 \quad \forall q\in Q_h^0 \end{aligned}$$

where u^k, p^k denote the kth iterate and $\sigma \in \{0, 1\}$ defines the linearization method: $\sigma = 1$ for the Newton method, and $\sigma = 0$ for Picard iteration (referred to as *simple iteration* in Reference [4]). Whilst the Newton method uses an exact linearization of the discrete equations, the Picard iteration can be regarded as using an inexact linearization. The linear systems of the Picard method are easier to solve, but this comes at the cost of slower convergence of the nonlinear system. However, since Picard iteration is globally convergent it can be used to provide a good initial iterate for the Newton method, which is not globally convergent.

If we set aside consideration of boundary conditions and the zero mean pressure condition (ZMPC), the system arising from the linearized discrete flow equations can be written as

$$\tilde{J}\varphi = \begin{bmatrix} F & B^{\mathrm{T}} \\ B & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

where the F block arises from the $a(\cdot,\cdot) + c(\cdot,\cdot,\cdot)$ terms, with B and B^{T} from the $b(\cdot,\cdot)$ terms. In Reference [2], a preconditioner is proposed and analysed for this type of problem, based upon the systems that arise from Picard iteration. The equivalent left preconditioner is, in the factorized form

$$C_{\mathrm{L}}^{-1} = \begin{bmatrix} I & 0 \\ 0 & M_{\mathrm{p}}^{-1} F_{\mathrm{p}} A_{\mathrm{p}}^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -B^{\mathrm{T}} & I \end{bmatrix} \begin{bmatrix} F^{-1} & 0 \\ 0 & I \end{bmatrix} \tag{3}$$

which utilizes the discretization of a pressure space advection–diffusion operator F_{p} (the pressure space analog to F), the inverse of the pressure space mass matrix M_{p}, and the inverse of a pressure space Laplacian A_{p}, along with the B block and the inverse of the F block of $\tilde{J}$. In the implementation of the preconditioner the actions of the different inverse matrices are replaced by (inexact) solves which can be performed very efficiently using defect correction Multigrid as preconditioner for a Krylov solver (for F and A_{p}, e.g. Reference [5]) and conjugate gradients (CG) for M_{p}, for example.

In Reference [3], this preconditioner and its analysis were extended to the systems arising from the Newton linearization, where the F_{p} part becomes an advection–diffusion–reaction operator, in analogy to the change of the F block of the system. The resulting preconditioner performs similarly well, but only up to a certain problem dependent Reynolds number. For larger Reynolds numbers the performance degrades quickly, as our numerical results confirm (see Section 4). Since application of the discrete adjoint method for sensitivity analysis requires the (exact) Jacobian, we restrict our considerations to the Newton linearization for the remainder of this paper. Further we restrict to the regime in which this preconditioner performs well.

One way to handle the boundary conditions and the ZMPC is to modify the assembled equation system to incorporate them implicitly, reducing the dimension of the system. In the case of the ZMPC however this would destroy the sparsity of the B and B^{T} blocks of the system, which is not desirable. Therefore, in our implementation we adopt an idea from Reference [6]. The core of this idea is that if we wish to solve a given system $\tilde{J}\varphi = b$ which is singular on its own, but becomes non-singular by restricting the solution to a given linear subspace, then the modified system can be expressed as $P^{\mathrm{T}}\tilde{J}P\varphi = P^{\mathrm{T}}b$, where P is a projector onto the subspace. In Reference [6] a modification to the preconditioned CG method is proposed, which allows solution of the system $P^{\mathrm{T}}\tilde{J}P\varphi = P^{\mathrm{T}}b$ operating in the unrestricted space but applying the projector at appropriate points to enforce that the solution iterates are in the desired subspace in which the problem is well defined. These modifications are analysed and it is concluded that if the left preconditioner C_{L}^{-1} is replaced by $PC_{\mathrm{L}}^{-1}P^{\mathrm{T}}$ exactly the desired effect is achieved.

We have adopted the use of this idea in the preconditioned GMRES solver, using a projector P which assures that the ZMPC and the Dirichlet boundary conditions are fulfilled in each step. In practise it means that we start GMRES with an iterate which fulfils the boundary condition and the ZMPC, and all search directions in GMRES are projected to zero on the boundary and the mean pressure is subtracted (application of P). The application of P^{T} can be interpreted as making sure that the residual vector to which the preconditioner C_{L}^{-1} is applied conforms with the restrictions.

3. PRECONDITIONING THE DISCRETE ADJOINT

As the discrete adjoint Equation (1) utilizes the transpose J^{T} of the Jacobian, each unknown ψ_i of the adjoint system corresponds to one of the discrete flow equations. This implies that the adjoint variables corresponding to the ZMPC equation and the boundary condition equations are unknowns as well, therefore these equations have to be formulated explicitly and application of the projection method used in the previous section is not immediately possible. Suppose the discrete flow variables are ordered as

$$\varphi = [u_{1,1},\ldots,u_{N_{\mathrm{int}},1},\ldots,u_{N_{\mathrm{int}}+N_{\mathrm{bc}},1},u_{1,2},\ldots,u_{N_{\mathrm{int}},2},\ldots,u_{N_{\mathrm{int}}+N_{\mathrm{bc}},2},p_1,\ldots,p_{N_{\mathrm{p}}}]^{\mathrm{T}}$$

where $u_{i,j}$ denotes the jth velocity component at the ith node of the mesh, p_i the pressure at the ith pressure node, N_{int} the number of internal nodes, N_{bc} the number of boundary nodes and N_{p} the number of pressure nodes. If we explicitly incorporate the velocity boundary conditions and the ZMPC into the flow equations then the Jacobian has the block structure

$$J = \begin{bmatrix} F_{1,1\mathrm{int}} & F_{1,1\mathrm{bc}} & F_{1,2\mathrm{int}} & F_{1,2\mathrm{bc}} & B_{1\mathrm{int}}^{\mathrm{T}} \\ 0 & I_{1\mathrm{bc}} & 0 & 0 & 0 \\ F_{2,1\mathrm{int}} & F_{2,1\mathrm{bc}} & F_{2,2\mathrm{int}} & F_{2,2\mathrm{bc}} & B_{2\mathrm{int}}^{\mathrm{T}} \\ 0 & 0 & 0 & I_{2\mathrm{bc}} & 0 \\ \tilde{B}_{1\mathrm{int}} & \tilde{B}_{1\mathrm{bc}} & \tilde{B}_{2\mathrm{int}} & \tilde{B}_{2\mathrm{bc}} & 0 \\ 0 & 0 & 0 & 0 & w^{\mathrm{T}} \end{bmatrix} \tag{4}$$

where index 1int (resp. 2int) denotes the interior nodes with the first (resp. second) velocity component and 1bc (resp. 2bc) denotes the boundary nodes with the first (resp. second) velocity component. Again, the F blocks ($F_{1,*}$ and $F_{2,*}$) arise from linearizing the x and y components of the $a(u,v)+c(u,u,v)$ terms in the momentum equations, the B^{T} blocks from the $b(v,p)$ term, the B blocks from the incompressibility condition $b(u,q)=0$ and the $\tilde{B}_*$ denote B_* with the last row dropped (to be replaced by the ZMPC). The identity matrices $I_{1\mathrm{bc}}$ and $I_{2\mathrm{bc}}$ enforce the velocity boundary conditions and w is a column vector such that $w^{\mathrm{T}}p=0$ gives the ZMPC. It can be proved that this system is equivalent to a system where the ZMPC is enforced using a Lagrange multiplier approach.

In order to define a left preconditioner for J^{T} we can use the transpose of a right preconditioner C_{R}^{-1} for J since

$$(JC_{\mathrm{R}}^{-1})^{\mathrm{T}} = (C_{\mathrm{R}}^{-1})^{\mathrm{T}} J^{\mathrm{T}}$$

In Section 2 we used the projection idea to apply (3) in the presence of boundary conditions and the ZMPC. For the adjoint problem this is not immediately possible, but some equivalent action must still be taken. Looking at the last block row of the adjoint equations,

$$B_{1\mathrm{int}}\psi_{1\mathrm{int}} + B_{2\mathrm{int}}\psi_{2\mathrm{int}} + w\psi_w = r_{\mathrm{p}}$$

it is apparent that the adjoint variable ψ_w corresponding to the ZMPC affects the residual components of the whole pressure space vector. We are able to reduce this influence of the weight vector w by introducing an additional step into the preconditioner which updates ψ_w such that the square of the residual of the last block row of the adjoint equation becomes minimal, which leads to

$$\psi_w = \frac{w^{\mathrm{T}}(r_{\mathrm{p}} - B_{1\mathrm{int}}\psi_{1\mathrm{int}} - B_{2\mathrm{int}}\psi_{2\mathrm{int}})}{w^{\mathrm{T}}w}$$

Additional care should be taken to ensure that the other parts of the preconditioner do not alter ψ_w in the preceding steps.

Our numerical results for this modification of the adjoint preconditioner show the desired performance, i.e. iteration numbers independent of h, see Section 4. Hence we propose the following algorithm as a preconditioner for the discrete adjoint equations.

To apply $y = \tilde{C}^{-1} r$ to a vector $r = [r_{u_1\mathrm{int}}, r_{u_1\mathrm{bc}}, r_{u_2\mathrm{int}}, r_{u_2\mathrm{bc}}, r_{\mathrm{p}}]^{\mathrm{T}}$:

1. Solve

$$\begin{bmatrix} F_{1,1\mathrm{int}}^{\mathrm{T}} & & F_{2,1\mathrm{int}}^{\mathrm{T}} & \\ F_{1,1\mathrm{bc}}^{\mathrm{T}} & I_{1\mathrm{bc}}^{\mathrm{T}} & F_{2,1\mathrm{bc}}^{\mathrm{T}} & \\ F_{1,2\mathrm{int}}^{\mathrm{T}} & & F_{2,2\mathrm{int}}^{\mathrm{T}} & \\ F_{1,2\mathrm{bc}}^{\mathrm{T}} & & F_{2,2\mathrm{bc}}^{\mathrm{T}} & I_{2\mathrm{bc}}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} y_{u_1\mathrm{int}} \\ y_{u_1\mathrm{bc}} \\ y_{u_2\mathrm{int}} \\ y_{u_2\mathrm{bc}} \end{bmatrix} = \begin{bmatrix} r_{u_1\mathrm{int}} \\ r_{u_1\mathrm{bc}} \\ r_{u_2\mathrm{int}} \\ r_{u_2\mathrm{bc}} \end{bmatrix}$$

2. Define a temporary vector $z_1 = r_{\mathrm{p}} - B_{1\mathrm{int}} y_{u_1\mathrm{int}} - B_{2\mathrm{int}} y_{u_2\mathrm{int}}$.
3. Calculate $y_w = (w^{\mathrm{T}} z_1)/(w^{\mathrm{T}} w)$.
4. Update $z_1 := z_1 - y_w w$.
5. Solve $M_{\mathrm{p}} z_2 = z_1$.
6. Multiply $z_1 := F_{\mathrm{p}}^{\mathrm{T}} z_2$.
7. Solve $A_{\mathrm{p}} z_2 = z_1$.
8. The resulting vector is $y = [y_{u_1\mathrm{int}}, y_{u_1\mathrm{bc}}, y_{u_2\mathrm{int}}, y_{u_2\mathrm{bc}}, y_{\mathrm{p}}, y_w]^{\mathrm{T}}$, where y_{p} is z_2 with the last component discarded.

4. NUMERICAL RESULTS

To support our research we have written a finite element solver for the steady state incompressible Navier–Stokes equations in two dimensions. The solver works on unstructured triangular meshes, using the Taylor–Hood (P_2/P_1) element pair.

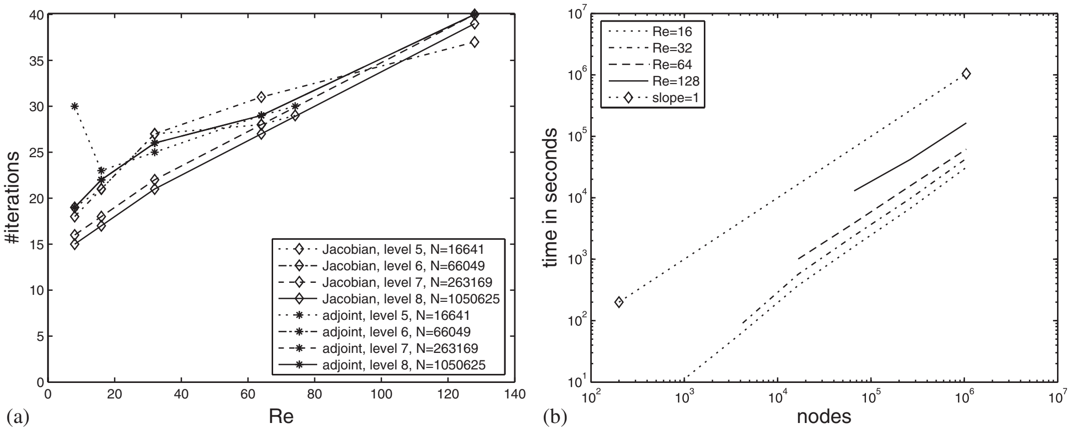

Figure 1. (a) GMRES iterations for Jacobian and adjoint system; and (b) overall time to solution.

A key element in making these preconditioning techniques competitive is an efficient solver for the F block, the advection–diffusion–reaction operator. In the case of Picard iteration this reduces to an advection–diffusion operator for which GMRES preconditioned by a defect correction multigrid scheme utilizing stabilized coarse grid discretizations works very well, giving optimal (h-independent) performance, see Reference [7]. This technique also seems to work well for the advection–diffusion–reaction operator which arises in the Newton linearization, although only up to moderate Reynolds number.

We have tested these techniques on a variety of test problems including flow in a channel with a bump, flow around an obstacle and a lid driven cavity. Here, we present some of the results for the driven cavity, as they are representative of the other problems also.

To demonstrate the behaviour of the F_{p} preconditioner for different Reynolds number regimes, the lines marked with diamonds in Figure 1(a) show the iteration numbers required for a relative residual reduction of 10^{-5} in the first Newton step on the specified mesh, plotted against the Reynolds number. The different lines represent different refinement levels, for which the number of nodes N is given in the legend. These lines are very close to each other, indicating essentially mesh independent iteration counts, whilst the iteration counts grow with Re. These results have been obtained with all inner solves set to a 10^{-6} relative improvement in residual norm. Note that there is a trade-off between the number of outer iterations and the number of iterations taken for the inner parts of the preconditioner. Whilst for low Reynolds number rather inaccurate inner solves perform well, at larger Reynolds number a higher accuracy for the inner solves is desirable in terms of the overall cost of the solution.

The lines marked with stars in Figure 1(a) show the iteration counts required for a relative residual reduction by 10^{-5} for the adjoint equation, with the modified preconditioner. Note that only one line is visible for levels 6, 7, and 8 because the iteration counts coincide exactly. We conclude that the number of iterations for the adjoint problem compares well with the number of iterations for the first Newton step.

Finally Figure 1(b) plots the overall time to solution against the number of nodes in the mesh, for four different Reynolds number regimes. The overall time to solution includes the computation of the flow solution, performance criterion (drag at the bottom of the cavity) and gradient of the performance criterion with respect to the node positions at the bottom of the domain. The results confirm that the proposed preconditioning technique can be used to implement a near optimal flow and sensitivity calculation, in the sense that, for given Reynolds number, the overall time to solution is almost linear in the number of nodes.

REFERENCES

1. Giles MB, Pierce NA. *An Introduction to the Adjoint Approach to Design*. Kluwer Academic Publishers: Dordrecht, 2000.
2. Kay D, Loghin D, Wathen A. A preconditioner for the steady-state Navier–Stokes equations. *SIAM Journal on Scientific Computing* 2002; **24**(1):237–256.
3. Elman HC, Loghin D, Wathen AJ. Preconditioning techniques for Newton's method for the incompressible Navier–Stokes equations. *BIT* 2003; **43**(5):961–974.
4. Gunzburger MD. *Finite Element Methods for Viscous Incompressible Flows*. Academic Press: New York, 1989.
5. Trottenberg U, Oosterlee C, Schuller A. *Multigrid*. Academic Press: New York, 2001.
6. Meyer A. Projection techniques embedded in the pcgm for handling hanging nodes and boundary restrictions. In *Engineering Computational Technology*, Topping BHV, Bittnar Z (eds). Saxe-Cobourg Publications: Edinburgh, 2002; 147–165.
7. Ramage A. A multigrid preconditioner for stabilised discretisations of advection–diffusion problems. *Journal of Computational and Applied Mathematics* 1999; **110**:187–203.