Electronic Transactions on Numerical Analysis.
Volume 34, pp. 125-135, 2009.

ISSN 1068-9613.

ETNA
Kent State University
http://etna.math.kent.edu

CHEBYSHEV SEMI-ITERATION IN PRECONDITIONING FOR PROBLEMS INCLUDING THE MASS MATRIX*

ANDY WATHEN† AND TYRONE REES†

Dedicated to Víctor Pereyra on the occasion of his 70th birthday

Abstract. It is widely believed that Krylov subspace iterative methods are better than Chebyshev semi-iterative methods. When the solution of a linear system with a symmetric and positive definite coefficient matrix is required, the Conjugate Gradient method will compute the optimal approximate solution from the appropriate Krylov subspace, that is, it will implicitly compute the optimal polynomial. Hence a semi-iterative method, which requires eigenvalue bounds and computes an explicit polynomial, must, for just a little less computational work, give an inferior result. In this manuscript, we identify a specific situation in the context of preconditioning where finite element mass matrices arise as certain blocks in a larger matrix problem when the Chebyshev semi-iterative method is the method of choice, since it has properties which make it superior to the Conjugate Gradient method. In particular, the Chebyshev method gives preconditioners which are linear operators, whereas corresponding use of conjugate gradients would be nonlinear. We give numerical results for two example problems, the Stokes problem and a PDE control problem, where such nonlinearity causes poor convergence.

Key words. Iteration, linear systems, preconditioning, finite elements, mass matrix

AMS subject classifications. 6F10, 65N20

1. Introduction.

Suppose we are interested in solving a system of linear equations

$$\mathcal{A}\mathbf{x} = \mathbf{b} \tag{1.1}$$

in the situation where $\mathcal{A} \in \mathbb{R}^{n\times n}$ is sparse and n is large. Such problems arise ubiquitously in the numerical solution of partial differential equation problems as well as other areas. One approach is to use iterative methods, and leading contenders are methods of Krylov subspace type. These require matrix vector products with $\mathcal{A}$ and compute a sequence of iterates $\{\mathbf{x}_k\}$ from a starting guess $\mathbf{x}_0$ which belong to nested Krylov subspaces

$$\mathbf{x}_k \in \mathbf{x}_0 + \mathrm{span}\{\mathbf{r}_0, \mathcal{A}\mathbf{r}_0, \mathcal{A}^2\mathbf{r}_0, \ldots, \mathcal{A}^{k-1}\mathbf{r}_0\}$$

for $k = 1, 2, \ldots, n$, where $\mathbf{r}_k = \mathbf{b} - A\mathbf{x}_k$ is the residual; see, for example, [8, 14, 26]. The most well-known such method is the method of Conjugate Gradients due to Hestenes and Stiefel [16], which is applicable in the case that $\mathcal{A}$ is symmetric and positive definite. Hereafter, we denote this method by the abbreviation CG. For indefinite symmetric matrices, the MINRES method of Paige and Saunders [20] is the Krylov subspace method of choice and this is the method we employ in our examples in this paper.

Most often, such Krylov subspace methods are used in conjunction with a preconditioner $\mathcal{P}$ [6]. The preconditioner should be such that an appropriate Krylov subspace method applied to $\mathcal{P}^{-1}\mathcal{A}$ or $\mathcal{A}\mathcal{P}^{-1}$, or if it is useful to preserve symmetry to $\mathcal{M}^{-1}\mathcal{A}\mathcal{M}^{-T}$, where $\mathcal{P} = \mathcal{M}\mathcal{M}^T$, will give a sequence of iterates which converges rapidly. Even in the symmetric case it is not necessary to form any factorization of $\mathcal{P}$ in practice and, in fact, in all cases all that is needed for a given vector $\mathbf{r}$ is to be able to solve $\mathcal{P}\mathbf{z} = \mathbf{r}$ for $\mathbf{z}$. For this reason, $\mathcal{P}$ does not have to be known explicitly as a matrix, but it must be a *linear* operator; else the preconditioned operator $\mathcal{P}^{-1}\mathcal{A}$ (or any of the other forms as above) to which the

* Received October 2, 2008. Accepted February 11, 2009. Published online August 25, 2009. Recommended by Godela Scherer.

†Oxford University Computing Laboratory, Parks Road, Oxford, UK. ({Andy.Wathen,Tyrone.Rees}@comlab.ox.ac.uk)

ETNA
Kent State University
http://etna.math.kent.edu

Krylov subspace method is applied is also not a linear operator. We comment that there are well-known nonlinear Conjugate Gradient methods such as that of Fletcher and Reeves [9] (see also Powell [21]) in the optimization literature, but generally in the context of solving a system of linear equations it would seem desirable to maintain linearity by using a linear preconditioner $\mathcal{P}$.

In this paper we explore a practical implication of this simple observation as it relates to nested iterations when just part of the preconditioner is nonlinear. We give two practical numerical examples, both for symmetric and indefinite matrices $\mathcal{A}$ of saddle-point form [4]. The first arises from the mixed finite element approximation of the Stokes problem in computational fluid dynamics and the second from a problem of PDE-constrained optimization. In both cases, we employ the Minimal Residual (MINRES) method of Paige and Saunders [20] rather than CG because this is the method of choice for symmetric and indefinite systems. The issue of linear versus nonlinear preconditioning is as relevant for this iterative method as for CG.

2. Preconditioning, Krylov subspace methods and Chebyshev semi-iteration. The MINRES method is a Krylov subspace method based on the Lanczos algorithm. To use MINRES with a preconditioner $\mathcal{P} = HH^T$, which must be symmetric positive definite, we solve the (symmetric) system

$$H^{-1}\mathcal{A}H^{-T}y = H^{-1}b,\ y = H^T x, \tag{2.1}$$

which is equivalent to (1.1). The matrix H is not required – all that is needed for a practical algorithm is a procedure for evaluating the action of $\mathcal{P}^{-1}$; see, for example, Algorithm 6.1 in [8]. In some cases, an obvious preconditioning procedure may be a nonlinear operator $\mathcal{Q} \cong \mathcal{P}^{-1}$. The theoretical framework for guaranteeing a minimum convergence rate for the appropriate Krylov subspace is at least more complicated and may no longer be valid with use of such a nonlinear procedure, although in practice it is possible that such a procedure may give good results on some examples.

There are flexible Krylov subspace methods which allow for a different preconditioner at each iteration; however, the performance (convergence) of these methods is more complex, and it might be considered generally desirable to stay within the standard linear convergence framework where possible. The most well-known of the flexible methods is the flexible GMRES method for nonsymmetric matrices [25], though there has been significant research in this area; for example, see [29] and references therein, which also include work on symmetric matrices. Simoncini and Szyld present a quite general analysis of Krylov subspace iteration with preconditioning which is also a Krylov method for the *same* matrix. In this case one can view the iterates as lying in some higher-dimensional Krylov subspace and often can establish convergence. The situation we consider here arises in several practical applications (as illustrated by our examples) and is different in that only a part of the preconditioner is an iteration for a *different* matrix.

Any Krylov subspace method (including CG) computes iterates $\mathbf{x}_k$ of the form

$$\mathbf{x}_k = \mathbf{x}_0 + q_{k-1}(\mathcal{A})\mathbf{r}_0, \tag{2.2}$$

where q_{k-1} is a polynomial of degree $k-1$. Often this property is described in terms of the residuals $\mathbf{r}_k = \mathbf{b} - \mathcal{A}\mathbf{x}_k$ rather than the iterates as

$$\mathbf{r}_k = p_k(\mathcal{A})\mathbf{r}_0,$$

where p_k is a polynomial of degree k satisfying $p_k(0) = 1$. This is easily seen from (2.2) by multiplication of each side by $\mathcal{A}$ and subtraction from $\mathbf{b}$:

$$\mathbf{r}_k = \mathbf{b} - \mathcal{A}\mathbf{x}_k = \mathbf{b} - \mathcal{A}\mathbf{x}_0 - \mathcal{A}q_{k-1}(\mathcal{A})\mathbf{r}_0 = \mathbf{r}_0 - \mathcal{A}q_{k-1}(\mathcal{A})\mathbf{r}_0 = p_k(\mathcal{A})\mathbf{r}_0,$$

ETNA
Kent State University
http://etna.math.kent.edu

where $p_k(z) = 1 - zq_{k-1}(z)$ clearly satisfies $p_k(0) = 1$. This procedure is clearly reversible when $\mathcal{A}$ is invertible, hence the equivalence of these statements. Now suppose that the same Krylov subspace method is used on the same problem with a different starting vector $\widehat{\mathbf{x}}_0$; we will compute iterates $\widehat{\mathbf{x}}_k$ satisfying

$$\widehat{\mathbf{x}}_k = \widehat{\mathbf{x}}_0 + \widehat{q}_{k-1}(\mathcal{A})\widehat{\mathbf{r}}_0,$$

where $\widehat{\mathbf{r}}_0 = \mathbf{b} - \mathcal{A}\widehat{\mathbf{x}}_0$. Even for the first iterate we will have

$$\begin{aligned}\mathbf{x}_1 &= \mathbf{x}_0 + \gamma\mathcal{A}\mathbf{r}_0 = \mathbf{x}_0 + \gamma\mathcal{A}\mathbf{b} - \gamma\mathcal{A}^2\mathbf{x}_0\\ \widehat{\mathbf{x}}_1 &= \widehat{\mathbf{x}}_0 + \widehat{\gamma}\mathcal{A}\widehat{\mathbf{r}}_0 = \widehat{\mathbf{x}}_0 + \widehat{\gamma}\mathcal{A}\mathbf{b} - \widehat{\gamma}\mathcal{A}^2\widehat{\mathbf{x}}_0\end{aligned}$$

so that, for example,

$$\mathbf{x}_1 + \widehat{\mathbf{x}}_1 = \mathbf{x}_0 + \widehat{\mathbf{x}}_0 + (\gamma + \widehat{\gamma})\mathcal{A}\mathbf{b} - \mathcal{A}^2(\gamma\mathbf{x}_0 + \widehat{\gamma}\widehat{\mathbf{x}}_0).$$

Correspondingly if $\overline{\mathbf{x}}_0 = \mathbf{x}_0 + \widehat{\mathbf{x}}_0$ is chosen as starting vector, we get

$$\overline{\mathbf{x}}_1 = \overline{\mathbf{x}}_0 + \overline{\gamma}\mathcal{A}\mathbf{b} - \overline{\gamma}\mathcal{A}^2\overline{\mathbf{x}}_0.$$

It is easy to see that $\overline{\mathbf{x}}_1 \neq \mathbf{x}_1 + \widehat{\mathbf{x}}_1$, whatever the values of the constants γ. This is a simple demonstration that any Krylov subspace method is nonlinear; this fact is in some sense well-known, but is sometimes overlooked. The above demonstrates nonlinearity with respect to the starting vector, but in a similar manner it is easy to show nonlinearity with respect to the right hand side vector; in correspondence with the above, if $\mathcal{A}\mathbf{x} = \mathbf{b}$ and $\mathcal{A}\widehat{\mathbf{x}} = \widehat{\mathbf{b}}$ are solved by a Krylov subspace method for the same starting vector $\mathbf{x}_0$ then $\mathbf{x}_1 + \widehat{\mathbf{x}}_1$ is not the first iterate for $\mathcal{A}(\mathbf{x} + \widehat{\mathbf{x}}) = \mathbf{b} + \widehat{\mathbf{b}}$. The sophistication of CG is that it implicitly computes automatically the best polynomial both with respect to the eigenvalues of $\mathcal{A}$ but also dependent on the components of the initial residual in the eigenvector directions; see, for example, [3, p. 560], [22, Section 2.5]. In summary, for CG we have $\mathbf{r}_k = p(A, \mathbf{r}_0)\mathbf{r}_0$, i.e., the polynomial p depends on $\mathbf{r}_0$.

The Chebyshev semi-iterative method was developed by Golub and Varga [12], [13, Section 10.1.5], and also Young [15]. This method, by contrast to CG, implicitly computes the same shifted and scaled Chebyshev polynomials of each degree k independently of the initial guess and right hand side vector provided the same spectral bounding parameters are used. Precisely, for the Chebyshev method applied to accelerate Richardson iteration

$$\mathbf{x}_k = (I - \mathcal{A})\mathbf{x}_{k-1} + \mathbf{b}$$

we have the Chebyshev iterates $\{\mathbf{y}_k\}$ satisfying

$$\mathbf{x} - \mathbf{y}_k = s_k(\mathcal{A})(\mathbf{x} - \mathbf{x}_0),$$

where s_k is a Chebyshev polynomial of degree k shifted and scaled so that $s_k(0) = 1$. Since s_k does not depend on $\mathbf{x}_0$ nor on $\mathbf{b}$, it is clear that $\mathbf{y}_k$ is the result of applying a fixed linear operator for the approximate solution of (1.1). This is true for each k. Of course, using different values $k_1 \neq k_2$ leads to different linear operators, as does varying the spectral bounding parameters. For larger k the approximation will be better, but in terms of using a fixed number of Chebyshev semi-iterative steps as a preconditioner, the important point we emphasize is that this preconditioner $\mathcal{P}$ is a linear operator. Thus $\mathbf{r}_k = p(A)\mathbf{r}_0$, where the polynomial p does not depend on $\mathbf{r}_0$. The same holds true if a different splitting than Richardson is used; we will employ a Jacobi splitting in the examples below.

ETNA
Kent State University
http://etna.math.kent.edu

The basic point here is that although the Chebyshev method computes a sequence of polynomials in the matrix as a Krylov subspace method does, these polynomials are fixed by the parameter estimates of the spectral interval or region which are employed, i.e., the coefficients do not depend on the starting guess nor the right hand side, unlike with CG. That means for any particular choices of the spectral bounding parameters that a fixed number of steps of the Chebyshev semi-iteration is a fixed linear operator $\mathcal{P}$, and so can be reliably used as a preconditioner for a linear system or, in fact, as a part of a preconditioner — this is how we employ it in the examples below.

This is in contrast to even a fixed number of iterative steps of a Krylov subspace method, which is always a nonlinear operator and so does not fit into the theory of (linear) preconditioning for linear systems of equations. It is reasonable that if sufficiently many iterations of a Krylov subspace method (or any other convergent iterative method) are employed so that we essentially have convergence to the exact solution, then the corresponding operator is linear, namely $\mathcal{A}^{-1}$. Of course, if an inner convergence tolerance is used for *any* method, then this will give different preconditioners at each outer iteration, in which case a flexible outer iterative method should be employed. In any particular computation, an appropriate flexible method with an inner iteration could perform well; however, as mentioned above, there is no guarantee with such an approach because of the limited theoretical understanding of flexible methods. By contrast, our approach is completely covered by well-known theory, i.e., there is a guaranteed upper bound on the work required. We emphasize that we are not considering inner/outer iterations here, but a fixed linear preconditioning operator.

The Chebyshev semi-iteration has been previously explored as a preconditioner in the literature [2, 10, 18, 19, 24]. In particular, it has recently been used successfully as a preconditioner by Golub, Ruiz and Touhami [11, 30] in the case of solving a system with multiple right hand sides. Some additional theoretical results are also to be found in the report by Arioli and Ruiz [1].

3. Finite elements: the mass matrix. The major issue with Chebyshev methods is getting good a priori estimated bounds for the eigenvalues. Fortunately there are practical situations where such bounds are analytically known. One such is for the (consistent) mass matrix that arises in finite element computations.

Suppose that finite element basis functions $\{\phi_j, j = 1, \ldots, N\}$ are used for some problem on a domain Ω, then the consistent mass matrix is just the Gramm matrix

$$Q = \{q_{i,j}, i, j = 1, \ldots, N\}, \quad q_{i,j} = \int_\Omega \phi_i \phi_j,$$

which is symmetric and positive definite because the basis will always be chosen to be linearly independent. More than twenty years ago, the first author established analytic bounds for the eigenvalues of the matrix $\operatorname{diag}(Q)^{-1}Q$, or equivalently for the generalized eigenvalues λ satisfying

$$\det(Q - \lambda D) = 0,$$

where $D = \operatorname{diag}(Q)$ [31]. For example, for any conforming mesh of tetrahedral ($\boldsymbol{P}_1$) elements in three dimensions, the result is

$$\frac{1}{2} \le \lambda \le \frac{5}{2},$$

and for a mesh of rectangular bi-linear ($\boldsymbol{Q}_1$) elements in two dimensions

$$\frac{1}{4} \le \lambda \le \frac{9}{4}. \tag{3.1}$$

ETNA
Kent State University
http://etna.math.kent.edu

For other elements, see [31]; the matrix `wathen.m` in the test set of matrices in `matlab` assembled by Higham [17] is precisely such a mass matrix for the 'serendipity' finite element. The bounds are found to be as tight as possible in practical computations in that there are eigenvalues equal to both the upper and lower bounds.

Such a priori bounds are precisely what are required for Chebyshev semi-iteration. Chebyshev polynomials shifted from the interval $[-1,1]$ to $[\alpha,\beta]$ and scaled so that their intercept is unity will ensure that the Chebyshev iterates $\{\mathbf{y}_k\}$ satisfy

$$\|\mathbf{x}-\mathbf{y}_k\|_2 \le 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|\mathbf{x}-\mathbf{x}_0\|_2,$$

where $\kappa=\beta/\alpha$. For example, for the $\boldsymbol{Q}_1$ element with the required Jacobi splitting the result is

$$\|\mathbf{x}-\mathbf{y}_k\|_2 \le 2(\frac{1}{2})^k\|\mathbf{x}-\mathbf{x}_0\|_2$$

since $\kappa=9$. The usual convergence result that is quoted for CG iterates $\{\mathbf{x}_k\}$ is based on precisely these same Chebyshev polynomials and is

$$\|\mathbf{x}-\mathbf{x}_k\|_{\mathcal{A}} \le 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k \|\mathbf{x}-\mathbf{x}_0\|_{\mathcal{A}}, \tag{3.2}$$

where $\|\mathbf{z}\|_{\mathcal{A}}^2=\mathbf{z}^T\mathcal{A}\mathbf{z}$; see, for example, [8, Theorem 2.4], [14, Theorem 3.1.1]. In all the computations here which compare CG and Chebyshev for systems with Q, the diagonal scaling is used with both methods.

Figure 3.1 shows how little in general is lost in using the Chebyshev method rather than CG for such mass matrices. We give results for the first 20 iterations for both methods applied to a diagonally preconditioned $\boldsymbol{Q}_1$ mass matrix corresponding to a mesh size of $h=2^{-5}$ with a right hand side that is a random, normally distributed vector generated by `randn` in `matlab`. The easily computable quantity for monitoring convergence in each case is the residual $\mathbf{r}_k=\mathbf{b}-\mathcal{A}\mathbf{x}_k$ for CG, respectively $\mathbf{r}_k=\mathbf{b}-\mathcal{A}\mathbf{y}_k$ for Chebyshev; hence, we show the values of $\|\mathbf{r}_k\|_2$ in Figure 3.1(a) and the values of $\|\mathbf{r}_k\|_{\mathcal{A}^{-1}}$, the quantity that is actually minimized by CG, in Figure 3.1(b).

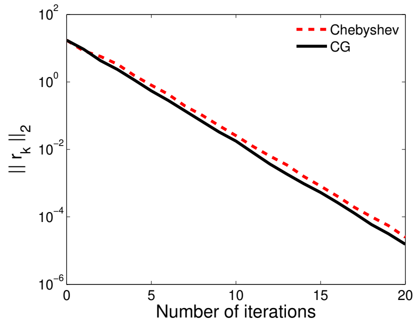
The behavior seen in Figure 3.1 is as we might expect from the theory. Although the conjugate gradient method in general shows superlinear convergence, in this particular case the spectrum of the preconditioned system is essentially uniformly distributed (and the right hand side is random), which corresponds to the situation that is considered to get the error bound (3.2). Thus, since CG on these matrices does not exhibit superlinear convergence, the linear convergence slope must be the same as that seen in the Chebyshev case.

4. Numerical Examples. Problems with constraints lead to saddle-point systems — an important class of symmetric (and nonsymmetric) indefinite matrices. The general structure is

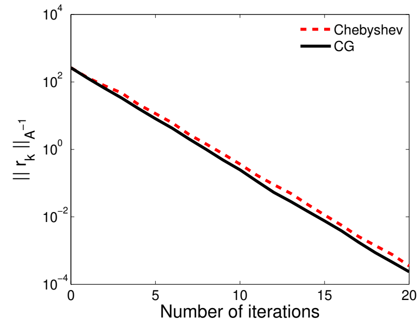
$$\mathcal{A}\mathbf{x}=\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix}\begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix}=\begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}, \tag{4.1}$$

where A may either be symmetric (giving the classical saddle-point system) or non-symmetric (giving a generalized saddle-point system). For a comprehensive survey of solution methods for saddle-point systems, see [4]. We consider only symmetric A here; in this situation, $\mathcal{A}$ is symmetric and indefinite, and the solver of choice would be the MINRES method of Paige and Saunders [20].

ETNA
Kent State University
http://etna.math.kent.edu



(a) The residuals in the 2-norm



(b) The residuals in the $\mathcal{A}^{-1}$-norm

FIGURE 3.1. *Comparison of convergence of* CG *and Chebyshev semi-iteration.*

4.1. Example 1. One of the more important PDE examples of a saddle-point system is the Stokes problem:

$$\nabla^2 u + \nabla p = f$$
$$\nabla \cdot u = 0;$$

see, for example, [8, Chapters 5 and 6]. This problem arises as the most common model for the slow flow of an incompressible fluid. This problem is self-adjoint, and most discretizations — including conforming mixed finite elements in any domain $\Omega \subset \mathbb{R}^d$ — lead to a symmetric matrix block A that is usually a $d \times d$ block diagonal matrix with diagonal blocks that are just discrete Laplacians.

Silvester and Wathen [28] proved that if $\widehat{A}$ is a spectrally equivalent approximation of the Laplacian, such as a multigrid cycle, and Q is the mass matrix as above (for the pressure space), then a block diagonal preconditioner of the form

$$\mathcal{P} = \begin{bmatrix} \widehat{A} & 0 \\ 0 & Q \end{bmatrix} \tag{4.2}$$

leads to optimal convergence of the MINRES iterative method for any (inf-sup) stable mixed finite element discretization. That is, the solution of (4.1) will be achieved in a number of MINRES iterations which is bounded independently of the number of unknowns in the finite element discretization.

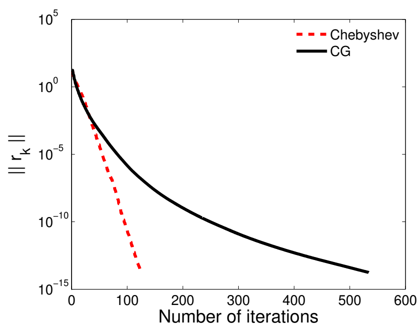
At each MINRES iteration it is necessary to solve a system of equations with coefficient matrix $\mathcal{P}$. Since a multigrid cycle is a simple stationary iteration, it is a linear operator — although certainly not known in general in the form of a matrix! By using exactly the same number of cycles (here, just one V-cycle) with the same number of pre- and post-smoothing steps at every application, this part of the preconditioner is a fixed linear operator. This is true even for the Algebraic Multigrid (AMG) procedure that we employ in our example computations. For the other part of the preconditioner involving the solution of linear systems with Q, it is advantageous to use the results of the previous section. Now the issue addressed in this paper arises: use of CG (with any preconditioner) for these Q systems will result in a nonlinear preconditioner even if a fixed number of iterations is employed, whereas a fixed number of steps of a Chebyshev method for Q with Jacobi splitting (i.e., with preconditioner $D = \mathrm{diag}(Q)$) is a linear operator, and so it preserves the linearity of $\mathcal{P}$.

Some simple numerical results illustrate the issue and show the clear advantage of the Chebyshev method in this situation. For clarity in our nomenclature, we let $\lambda_1 \leq \lambda_2 \leq \cdots \leq$

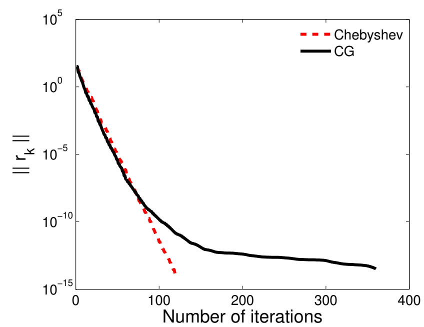
ETNA
Kent State University
http://etna.math.kent.edu

λ_m denote the eigenvalues of the symmetrically scaled matrix $D^{-\frac{1}{2}}QD^{-\frac{1}{2}}$ and $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m$ denote the corresponding eigenvectors.

In Figure 4.1, we plot the value of the Euclidean norm of the residual versus the iteration number in MINRES. In both cases we use $\boldsymbol{Q}_2$–$\boldsymbol{Q}_1$ mixed finite elements and the Stokes system is of size 2467×2467. The (1,1) block of (4.2) is given by a single AMG V-cycle using `HSL` package `HSL_MI20` applied via a `matlab` interface [5]. The (2,2) block is approximated using a fixed number of steps of either CG or Chebyshev semi-iteration, as described above, with diagonal scaling for both methods. In both cases the velocity part of the right hand side is given by the driven cavity flow problem in IFISS [7], whereas the pressure part, $\mathbf{g}$, is given by $\mathbf{v}_{(m+1)/2}$ and $\mathbf{v}_3 + \mathbf{v}_{(m+1)/2}$ in Figures 4.1(a) and 4.1(b), respectively. The pressure part of the right hand side is in this case not relevant to the physical problem, but it enables easy description of our particular example and gives an initial residual which must correspond to some starting guess for the correct physical right hand side. We use one CG iteration with starting vector $\mathbf{v}_m$ in Figure 4.1(a) and two CG iterations with starting vector $\mathbf{v}_1$ in Figure 4.1(b). On the same plots are shown the results with the same number of Chebyshev iterations with the exact spectral bounding parameters (3.1) for the $\boldsymbol{Q}_1$ pressure element used here.



(a) One step of CG/Chebyshev, $\mathbf{g} = \mathbf{v}_{(m+1)/2}$, $\mathbf{x}_0 = \mathbf{v}_m$



(b) Two steps of CG/Chebyshev, $\mathbf{g} = \mathbf{v}_3 + \mathbf{v}_{(m+1)/2}$, $\mathbf{x}_0 = \mathbf{v}_1$

FIGURE 4.1. *Convergence of* MINRES *when using fixed number of steps of* CG *and Chebyshev semi-iteration in the preconditioner.*

4.2. Example 2. Our second example also involves the saddle point system (4.1), but as it arises in the context of PDE-constrained optimization. Consider the (distributed) optimal control problem

$$\min_{u,f} \frac{1}{2}||u - \hat{u}||_2^2 + \beta||f||_2^2 \tag{4.3}$$

$$\text{subject to} \quad -\nabla^2 u = f \text{ in } \Omega \tag{4.4}$$

$$\text{with} \quad u = g \text{ on } \partial\Omega, \tag{4.5}$$

where Ω is some bounded domain, g and $\hat{u}$ are prescribed functions, and β is a regularization parameter. It can be shown that upon discretization, this problem is equivalent to solving the saddle point system

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{c} \\ \mathbf{d} \end{bmatrix}, \tag{4.6}$$

ETNA
Kent State University
http://etna.math.kent.edu

where $A = \begin{bmatrix} 2\beta Q & 0 \\ 0 & Q \end{bmatrix}$, $B = [-Q \quad K]$ with Q and K denoting the mass and stiffness matrices, respectively [23, 27].

Rees, Dollar and Wathen [23] showed that if we use MINRES to solve this system then an effective preconditioner is of the form

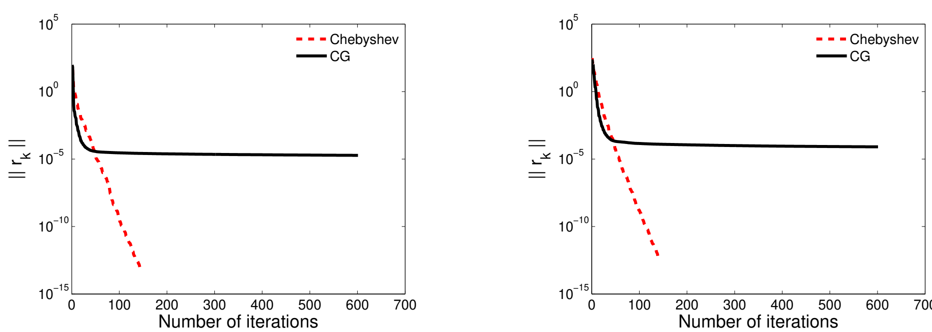
$$\mathcal{P} = \begin{bmatrix} 2\beta\widetilde{Q} & 0 & 0 \\ 0 & \widetilde{Q} & 0 \\ 0 & 0 & \widetilde{K}Q^{-1}\widetilde{K}^T \end{bmatrix}, \tag{4.7}$$

where $\widetilde{Q}$ and $\widetilde{K}$ are approximations to the mass and stiffness matrices. As in the first example, we can use a fixed number of multigrid iterations, say, for $\widetilde{K}$. The operator $\widetilde{Q}$ needs to be an approximation to the mass matrix which preserves linearity of $\mathcal{P}$; therefore CG is unsuitable, but, because of the results in Section 3, a fixed number of steps of the Chebyshev semi-iteration with Jacobi splitting should perform well.

Figure 4.2 illustrates the situation. Here we take $\Omega = [0,1]^2$ and discretize the problem using $\boldsymbol{Q}_1$ finite elements with mesh size 2^{-5} (making $\mathcal{A}$ a 2883×2883 system). We take the regularization parameter $\beta = 10^{-2}$. We take $\hat{u}$ such that

$$\hat{u} = \begin{cases} (2x-1)^2(2y-1)^2 & \text{if } (x,y) \in [0, \frac{1}{2}]^2 \\ 0 & \text{otherwise.} \end{cases}$$

For $\widetilde{K}$ we again use one AMG V-cycle using `HSL` package `HSL_MI20` applied via a `matlab` interface [5]. $\widetilde{Q}$ is one step of either diagonally scaled CG or Chebyshev semi-iteration, and in both cases the vector $\mathbf{d}$ in the right hand side is that given by Example 1 in [23]. In Figure 4.2(a), the vector $\mathbf{c}$ is given by $[\beta\mathbf{v}_{(m+1)/2} \; \mathbf{v}_3]^T$ and the starting vectors for both CG and Chebyshev are $\mathbf{v}_{(m+3)/2}$ for the (1,1) block and $\mathbf{v}_2$ for the (2,2) block. In Figure 4.2(b), $\mathbf{c} = [\mathbf{v}_m \; \mathbf{v}_3]^T$ and the initial vectors are $\mathbf{v}_{(m+1)/2}$ and $\mathbf{v}_{(m+3)/2}$.



(a) One step of CG/Chebyshev, $\mathbf{c} = (\beta\mathbf{v}_{(m+1)/2} \; \mathbf{v}_3)^T$, $\mathbf{x}_0^1 = \mathbf{v}_{(m+3)/2}$, $\mathbf{x}_0^2 = \mathbf{v}_2$

(b) One step of CG/Chebyshev, $\mathbf{c} = (\mathbf{v}_m \; \mathbf{v}_3)^T$, $\mathbf{x}_0^1 = \mathbf{v}_{(m+1)/2}$, $\mathbf{x}_0^2 = \mathbf{v}_{(m+3)/2}$

FIGURE 4.2. *Convergence of* MINRES *when using fixed number of steps of* CG *and Chebyshev semi-iteration in the preconditioner.*

In both these examples, we see failure in the convergence of the outer MINRES iteration when we use CG, presumably because of the nonlinear nature of the preconditioner. Again, these examples are artificial, but they serve to illustrate behavior that may occur in a practical application. The Chebyshev method is covered by the linear theory and so MINRES convergence in this case is as expected.

ETNA
Kent State University
http://etna.math.kent.edu

We have shown that a small fixed number of iterations of the Chebyshev semi-iteration can behave better than CG, but it remains to be seen how effective this actually is as a preconditioner. In Table 4.1, we give iteration counts and timings to solve the system (4.6) using (4.7) as a preconditioner for the described two-dimensional problem, where $\widetilde{Q}$ represents five steps of Chebyshev semi-iteration, ten steps of Chebyshev semi-iteration, $\mathtt{diag}(Q)$, the lumped mass matrix, or a sparse direct solve (`backslash` in `matlab`). The number of iterations is given in brackets after the CPU time, and tests were done using `matlab` version 7.5.0 on a machine with a dual processor AMD Opteron 244 (1.8GHz).

Here, again, we see that the sparse direct solver gives the smallest iteration counts, as one would expect, but the time taken to solve the system increases superlinearly as the problem size increases. The most efficient preconditioner out of the ones considered here is ten steps of the Chebyshev semi-iteration. In this case, since the multigrid solves in the preconditioner are relatively more expensive, it pays to have a more accurate approximation to the mass matrix — which can be done comparatively cheaply — giving a faster solution time overall. In Table 4.2, the results are given for the corresponding problem in three space dimensions.

TABLE 4.1

Comparison of times and iterations to solve (4.6) for different mesh sizes (h) (with $3n$ unknowns) to a tolerance of 10^{-6} for MINRES *with (4.7) as a preconditioner, where $\widetilde{Q}$ represents five steps of Chebyshev semi-iteration, ten steps of Chebyshev iteration, $\mathtt{diag}(Q)$, the lumped mass matrix, and a sparse direct solve (*`backslash` *in* `matlab`*).*

h	$3n$	Chebyshev (5)	Chebyshev (10)	Diagonal	Lumped	`backslash`
2^{-2}	27	0.15 (11)	0.12 (6)	0.14 (17)	0.16 (18)	0.12 (5)
2^{-3}	147	0.17 (12)	0.15 (8)	0.23 (28)	0.24 (28)	0.13 (7)
2^{-4}	675	0.23 (12)	0.19 (8)	0.37 (30)	0.32 (23)	0.17 (7)
2^{-5}	2883	0.47 (12)	0.36 (8)	0.74 (27)	0.60 (20)	0.35 (7)
2^{-6}	11907	1.4 (11)	1.12 (8)	2.42 (26)	1.70 (17)	1.31 (7)
2^{-7}	48387	5.5 (11)	4.43 (8)	9.05 (24)	6.41 (16)	5.73 (7)
2^{-8}	195075	22.9 (10)	17.8 (7)	38.0 (23)	28.5 (16)	43.9 (7)
2^{-9}	783363	111 (10)	84.2 (7)	102 (14)	115 (15)	1956 (7)

TABLE 4.2

Comparison of times and iterations to solve the three-dimensional problem corresponding to (4.6) for different mesh sizes (h) (with $3n$ unknowns) to a tolerance of 10^{-6} for MINRES *with (4.7) as a preconditioner, where $\widetilde{Q}$ represents five steps of Chebyshev semi-iteration, ten steps of Chebyshev iteration, fifteen steps of Chebyshev iteration, $\mathtt{diag}(Q)$, the lumped mass matrix, and a sparse direct solve (*`backslash` *in* `matlab`*).*

h	$3n$	Cheb. (5)	Cheb. (10)	Cheb. (15)	Diagonal	Lumped	`backslash`
2^{-2}	81	0.32 (11)	0.14 (9)	0.16 (7)	0.14 (12)	0.15 (12)	0.13 (5)
2^{-3}	1029	0.37 (18)	0.27 (11)	0.24 (8)	0.47 (33)	0.51 (38)	0.22 (5)
2^{-4}	10125	3.13 (18)	2.10 (11)	1.60 (8)	2.40 (20)	2.85 (23)	3.64 (7)
2^{-5}	89373	29.3 (18)	16.2 (9)	15.0 (8)	18.9 (16)	21.8 (18)	94.4 (7)
2^{-6}	750141	214 (15)	169 (11)	132 (8)	136 (13)	173 (16)	— (–)

5. Conclusions. In the context of preconditioning for Krylov subspace iterative methods for solving linear systems, the use of a Krylov subspace method in applying the preconditioner — or part of the preconditioner — leads necessarily to a nonlinear preconditioner. There are important situations where the Chebyshev semi-iterative method is essentially as effective as Conjugate Gradients, and it leads to a linear preconditioner provided that a fixed number of iterations are used. We have illustrated this by giving two examples where the consistent mass matrix is desired as part of a preconditioner and so this issue is important.

Acknowledgments. We thank Godela Scherer for her patience and the two anonymous referees for their comments, which have improved the content of this paper.

REFERENCES

[1] M. ARIOLI AND D. RUIZ, *A Chebyshev-based two-stage iterative method as an alternative to the direct solution of linear systems*, Tech. Report, STFC, Rutherford Appleton Laboratory, 2002.

[2] S. ASHBY, T. MANTEUFFEL, AND J. OTTO, *A comparison of adaptive Chebyshev and least squares polynomial preconditioning for Hermitian positive definite linear systems*, SIAM J. Sci. Comput., 13 (1992), pp. 1–29.

[3] O. AXELSSON, *Iterative Solution Methods*, Cambridge University Press, New York, 1994.

[4] M. BENZI, G. H. GOLUB, AND J. LIESEN, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), pp. 1–137.

[5] J. BOYLE, M. D. MIHAJLOVIC, AND J. A. SCOTT, *HSL_MI20: an efficient AMG preconditioner*, Tech. Report RAL-TR-2007-021, Department of Computational and Applied Mathematics, Rutherford Appleton Laboratory, 2007.

[6] P. CONCUS, G. GOLUB, AND D. O'LEARY, *A generalized conjugate gradient method for the numerical solution of elliptic partial-differential equations*, in Proceedings of the Symposium on Sparse Matrix Computations, J. R. Bunch and D. J. Rose, eds., Academic Press, New York, 1976, pp. 309–332.

[7] H. ELMAN, A. RAMAGE, AND SILVESTER. D.J., *Algorithm 886: IFISS, A Matlab toolbox for modelling incompressible flow*, ACM Trans. Math. Software, 33 (2007), 14 (18 pages).

[8] H. ELMAN, D. SILVESTER, AND A. WATHEN, *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2005.

[9] R. FLETCHER AND C. REEVES, *Function minimization by Conjugate Gradients*, Comput. J., 7 (1964), pp. 149–154.

[10] G. GOLUB AND M. KENT, *Estimates of eigenvalues for iterative methods*, Math. Comp., 53 (1989), pp. 249–263.

[11] G. GOLUB, D. RUIZ, AND A. TOUHAMI, *A hybrid approach combining a Chebyshev filter and conjugate gradient for solving linear systems with multiple right-hand sides*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 774–795.

[12] G. GOLUB AND R. VARGA, *Chebyshev semi iterative methods, successive overrelaxation iterative methods and second order Richardson iterative methods*, Numer. Math., 3 (1961), pp. 147–156.

[13] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, third ed., The Johns Hopkins University Press, Baltimore, 1996.

[14] A. GREENBAUM, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.

[15] L. A. HAGEMAN AND D. M. YOUNG, *Applied Iterative Methods*, Academic Press, New York, 1981.

[16] M. R. HESTENES AND E. STIEFEL, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.

[17] N. HIGHAM, *Algorithm 694: A collection of test matrices in MATLAB*, ACM Trans. Math. Software, 17 (1991), pp. 289–305.

[18] O. JOHNSON, C. MICCHELLI, AND G. PAUL, *Polynomial preconditioners for conjugate gradient calculations*, SIAM J. Numer. Anal., 20 (1983), pp. 362–376.

[19] D. O'LEARY, *Yet another polynomial preconditioner for the conjugate gradient algorithm*, Linear Algebra Appl., 29 (1991), pp. 377–388.

[20] C. C. PAIGE AND M. A. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

[21] M. POWELL, *An efficient method for finding the minimum of a function of several variables without calculating derivatives*, Computer J., 7 (1964), pp. 152–162.

[22] A. RAMAGE, *Preconditioned Conjugate Gradient Methods for Galerkin Finite Element Equations*, Ph.D. thesis, University of Bristol, 1990.

[23] T. REES, H. DOLLAR, AND A. WATHEN, *Optimal solvers for PDE constrained optimization*, SIAM J. Sci. Comput., to appear, 2009.

[24] Y. SAAD, *Practical use of polynomial preconditioning for the conjugate gradient method*, SIAM J. Sci. Comput., 6 (1985), pp. 865–881.

[25] ———, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM. J. Sci. Comput., 14 (1993), pp. 461–469.

[26] ———, *Iterative methods for sparse linear systems*, PWS Publishing, Boston, 1996. Second edition, SIAM, Philadelphia, 2003.

[27] J. SCHÖBERL AND W. ZULEHNER, *Symmetric indefinite preconditioners for saddle point problems with applications to PDE-constrained optimization problems*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 752–

ETNA
Kent State University
http://etna.math.kent.edu

773.

[28] D. SILVESTER AND A. WATHEN, *Fast iterative solution of stabilised Stokes systems Part II: Using general block preconditioners*, SIAM J. Numer. Anal., 31 (1994), pp. 1352–1367.

[29] V. SIMONCINI AND D. SZYLD, *Flexible inner-outer Krylov subspace methods*, SIAM J. Numer. Anal., 40 (2003), pp. 2219–2239.

[30] A. TOUHAMI, *Utilisation des Filtres de Tchebycheff et Construction de Préconditioners Spectraux pour l'Acceleration des Méthods de Krylov*, Ph.D. thesis, INPT-ENSEEIHT, Toulouse, France, 2005.

[31] A. J. WATHEN, *Realistic eigenvalue bounds for the Galerkin mass matrix*, IMA J. Numer. Anal., 7 (1987), pp. 449–457.