BIT Numerical Mathematics **43:** 231–244, 2003.
© 2003 *Kluwer Academic Publishers. Printed in The Netherlands*

APPROXIMATE INVERSE PRECONDITIONING FOR SHIFTED LINEAR SYSTEMS *

MICHELE BENZI[1] and DANIELE BERTACCINI[2]†

[1] *Department of Mathematics and Computer Science, Emory University*
Atlanta, GA 30322, USA. email: benzi@mathcs.emory.edu

[2] *Dipartimento di Matematica "Guido Castelnuovo", Università di Roma "La Sapienza"*
00185 Roma, Italy. email: bertaccini@mat.uniroma1.it

Abstract.

In this paper we consider the problem of preconditioning symmetric positive definite matrices of the form $A_\alpha = A + \alpha I$ where $\alpha > 0$. We discuss how to cheaply modify an existing sparse approximate inverse preconditioner for A in order to obtain a preconditioner for A_α. Numerical experiments illustrating the performance of the proposed approaches are presented.

AMS subject classification (2000): 65F10, 65N22, 15A18.

Key words: Preconditioning, shifted linear systems, sparse approximate inverses.

1 Introduction.

In this paper we consider the efficient construction of preconditioners for shifted matrices of the form

$$A_\alpha = A + \alpha I,$$

where A is a symmetric positive definite (SPD) matrix of order n and $\alpha > 0$ is a positive shift. We assume that a preconditioner P is initially computed for the matrix A, or possibly for $A + \alpha I$ for some initial value of $\alpha > 0$. The question is then how to compute a preconditioner P_α for subsequent values of the shift α. The goal is to obtain an overall solution procedure that is cheaper, in terms of total solution costs, than either reusing the same preconditioner over and over again (without modifications) or recomputing a new preconditioner from scratch for each new value of α. Reusing the same preconditioner each time often leads to slow convergence, whereas recomputing a preconditioner each time is both costly and wasteful. Clearly, there is a broad range of possibilities within these two extremes. It should be possible to modify an existing preconditioner at a cost much lower than recomputing a preconditioner from scratch; even if the resulting preconditioner can be expected to be less effective than a brand new one in terms of iteration count, the overall cost should be considerably reduced.

*Received June 2002. Revised December 2002. Communicated by Lars Eldén.

†This work was partially supported by NSF grant DMS-0207599, by INdAM-GNCS, and by the MURST project "Progetto Giovani Ricercatori anno 2000".

When A is a symmetric M-matrix and the preconditioner an incomplete Cholesky factorization, this problem has been studied by Meurant [15]. In this paper we consider general SPD matrices and sparse approximate inverse preconditioners in factorized form, focusing on the SAINV technique introduced in [2]. We recall here that the SAINV preconditioner is well defined for a general SPD matrix (not just for M-matrices), and that it is well-suited for parallel implementation, since its application requires only matrix–vector products. Although there exists a close relationship between incomplete Cholesky preconditioners and factored approximate inverses [4, 8], our approach is different from the one used by Meurant in [15]. Furthermore, we provide some theoretical analysis in support of our modification strategies.

The paper is organized as follows. In section 2 we mention a few situations that lead to shifted linear systems. In section 3 we briefly recall the SAINV preconditioner. In section 4 we describe our proposed approaches and present some underlying theoretical results. Section 5 is devoted to numerical experiments. We present further possible improvements and concluding remarks in section 6.

2 Motivation.

The solution of shifted linear systems is an important problem that arises in several contexts in scientific computing. Perhaps the most natural example is the solution of parabolic partial differential equations by implicit methods. Consider for instance a simple diffusion problem of the form

$$\frac{\partial u}{\partial t} = (\nabla \cdot D\nabla)\, u + f \tag{2.1}$$

on a plane region with Dirichlet boundary conditions and an initial condition $u(x, 0) = u_0(x)$. Finite difference discretization in space with stepsize h and an implicit (backward Euler) time discretization with time step τ results in a sequence of linear systems

$$\Big(I + \frac{\tau}{h^2} A\Big)\, u^{m+1} = u^m + \tau f^{m+1}, \quad m = 0, 1, 2, \ldots, M, \tag{2.2}$$

where A is SPD. Typically the time step τ will not be constant, but it will change adaptively from time to time. Upon multiplication of (2.2) by $\alpha = h^2/\tau$, one obtains a sequence of linear systems of the form

$$(\alpha I + A)x_\alpha = b_\alpha \tag{2.3}$$

for many different values of α. For example, if τ is of the same order of magnitude as h, we see that the shift $\alpha = \mathrm{O}(h)$ is a "small" number, relative to the entries of A.

More generally, sequences of systems of the form (2.3) occur in the numerical solution of discretized nonlinear systems of ordinary and partial differential equations with implicit methods; see, e.g., [1, 6, 7, 12] and the references therein. Shifted linear systems also occur in other contexts, such as regularization of ill-posed least squares problems, trust region methods in nonlinear optimization, and elsewhere.

3 The stabilized AINV preconditioner.

The SAINV (for Stabilized AINV) preconditioner [2, 13] is a robust variant of the AINV preconditioner [4] that is guaranteed to be well defined, in exact arithmetic, for general SPD matrices. The algorithm is based on a conjugate Gram–Schmidt (or A-orthogonalization) process. We start by recalling that since A is SPD, it defines an inner product on $\mathbb{R}^n$ via

$$\langle x, y\rangle_A := x^T Ay \quad \text{for all } x, y \in \mathbb{R}^n. \tag{3.1}$$

Given a set of n linearly independent vectors $v_1, v_2, \dots, v_n \in \mathbb{R}^n$, we can build an A-orthogonal (or A-conjugate) set of vectors $z_1, z_2, \dots, z_n \in \mathbb{R}^n$ by a conjugate Gram–Schmidt process, i.e., a Gram–Schmidt process with respect to the inner product (3.1). Written as a *modified* Gram–Schmidt process, the (right-looking) algorithm starts by setting $z_i = v_i$ and then performs the following nested loop:

$$z_i \leftarrow z_i - \frac{\langle z_j, z_i\rangle_A}{\langle z_j, z_j\rangle_A} z_j, \tag{3.2}$$

where $j = 1, 2, \dots, n-1$ and $i = j+1, \dots, n$. Letting

$$Z = [z_1, z_2, \dots, z_n],$$

we have

$$Z^T AZ = D = \text{diag }(d_1, d_2, \dots, d_n), \tag{3.3}$$

where

$$d_j = \langle z_j, z_j\rangle_A = z_j^T Az_j > 0, \quad 1 \le j \le n.$$

If we set $v_i = e_i$ (the i-th unit basis vector) for $1 \le i \le n$, then $Z^T = L^{-1}$ where L is the unit lower triangular factor in the root-free Cholesky factorization $A = LDL^T$; the matrix D is exactly the same here and in (3.3). Indeed, it is clear from (3.2) that the vector z_i is modified only above position i (for $2 \le i \le n$), therefore Z is unit upper triangular and by virtue of (3.3) and the uniqueness of the LDL^T factorization, it must be $Z^T = L^{-1}$.

The SAINV preconditioner is constructed by carrying out the updates in the A-conjugation process (3.2) incompletely. Given a drop tolerance $0 < \varepsilon < 1$, the entries of z_i are scanned after each update and entries that are smaller than ε in absolute value are discarded. We denote by $\tilde{z}_i$ the sparsified vectors, and we set

$$\tilde{Z} = [\tilde{z}_1, \tilde{z}_2, \dots, \tilde{z}_n].$$

In alternative, a *relative* drop tolerance can be used; for example, ε can be replaced by $\varepsilon||a_i||_2$, where a_i is the i-th column of A. It is often advantageous to symmetrically scale A so that A has unit diagonal; this tends to improve the conditioning of the matrix and it allows for the use of an absolute drop tolerance ε. Whatever the scaling or the drop strategy used, the incomplete A-orthogonalization process results in a sparse matrix $\tilde{Z} \approx L^{-T}$, that is, we have an incomplete inverse factorization of A of the form

$$A^{-1} \approx \tilde{Z}\tilde{D}^{-1}\tilde{Z}^T,$$

where $\tilde{D}$ is diagonal with entries $\tilde{d}_i = \tilde{z}_i^T A \tilde{z}_i > 0$. This is a factored sparse approximate inverse that can be used as a preconditioner for the conjugate gradient algorithm applied to $Ax = b$. The preconditioner is guaranteed to be positive definite (since $\tilde{Z}$ is nonsingular and $\tilde{d}_i > 0$ for all i) and is easily applied in parallel, since its application only requires matrix–vector products. The preconditioner has been successfully used in solving a wide range of challenging problems; see [2, 3, 13].

Because computing a sparse approximate inverse preconditioner is relatively expensive (generally more so than computing an incomplete Cholesky factorization), the potential savings from using cheap modifications of an existing SAINV preconditioner on a sequence of shifted linear system can in principle be very significant.

4 The proposed approaches.

Consider a family of linear systems

$$A_{\alpha_j} x_j = b_j, \quad A_{\alpha_j} = A + \alpha_j I, \quad \alpha_j \in [0, \alpha_{\max}], \quad j = 0, 1, \ldots, s, \tag{4.1}$$

where A is a large, sparse, possibly ill-conditioned SPD matrix, b_j are given right-hand side vectors, and x_j are the corresponding solution vectors. The linear systems (4.1) may be given simultaneously or sequentially; the latter case occurs, for instance, when the right-hand side b_j depends on the previous solution x_{j-1}, as in (2.2).

For simplicity of notation, we will consider a generic shift α and drop the subscript. Assume now that A has been normalized in such a a way that the largest entry in A is equal to 1. Then clearly if α is "large enough" there is no need to use any preconditioning whatsoever. Indeed, denoting by $\lambda_{\min}$ and $\lambda_{\max}$ the extremal eigenvalues of A, we have that

$$\kappa_2(A + \alpha I) = \frac{\lambda_{\max} + \alpha}{\lambda_{\min} + \alpha} \leq \frac{\lambda_{\max}}{\alpha} + 1 \tag{4.2}$$

and, in practice, preconditioning is no longer necessary (or beneficial) as soon as $\lambda_{\max}/\alpha$ is small enough. Note that, for diagonally dominant problems, $\lambda_{\max} \leq 2$ and, for our normalized examples (see Section 5), we have that $\lambda_{\max}$ is always less than four. In practice, we found that preconditioning is no longer beneficial as soon as α is of the order of 10^{-1}. In some cases this might already be true for even smaller values of α, depending on the distribution of the eigenvalues. At the other extreme, continuity suggests that there is a value of α under which one might as well reuse the preconditioner computed for the original A. However, in our experiments we found cases where reusing a preconditioner for A gives poor results already for α as small as $\mathcal{O}(10^{-5})$. Hence, there is a fairly broad range of values of α where modification strategies are of potential benefit.

4.1 Order k modified preconditioners.

Consider a preconditioner of the form

$$P^{-1} = \tilde{Z}\,\tilde{D}^{-1}\,\tilde{Z}^T \approx A^{-1}, \tag{4.3}$$

computed with the SAINV approach outlined in section 3. Letting $\tilde{L} = \tilde{Z}^{-T}$, we can write $A \approx P = \tilde{L}\tilde{D}\tilde{L}^T$. Hence, P can be regarded as an approximate LDL^T factorization of A. For theoretical analysis purposes, it will be sometimes useful to work with P rather than P^{-1}, although in practice only P^{-1} (more precisely, the factors $\tilde{Z}$ and $\tilde{D}$) is computed and operated with.

Let now $E = E^T$ be a generic symmetric matrix, and consider modified SAINV preconditioners of the form

$$P_\alpha^{-1} = \tilde{Z}\,(\tilde{D} + \alpha E)^{-1}\,\tilde{Z}^T. \tag{4.4}$$

In order for this preconditioner to be well defined and SPD, we need the matrix $\tilde{D} + \alpha E$ to be SPD. Moreover, the matrix E must be cheap to compute, and such that linear systems with coefficient matrix $\tilde{D} + \alpha E$ can be easily solved. With these constraints, we would like to find a matrix E such that P_α^{-1} given by (4.4) is a good preconditioner for A_α. To guide us in this choice, we assume that $P^{-1} = ZD^{-1}Z^T = A^{-1}$ (the exact inverse); hence, $P_\alpha^{-1} = Z\,(D + \alpha E)^{-1}\,Z^T$.

Consider now the difference

$$P_\alpha - A_\alpha = Z^{-T}\,(D + \alpha E)\,Z^{-1} - (A + \alpha I) = \alpha(LEL^T - I). \tag{4.5}$$

Taking $E = L^{-1}L^{-T} = Z^TZ$ in (4.4) would result in the exact inverse $P_\alpha^{-1} = A_\alpha^{-1}$. Note that Z^TZ is SPD and its (i,j) entry is given by $z_i^Tz_j$. Of course, this would not be a practical choice, quite apart from the fact that we don't know the exact Z in practice, but only a sparse approximation $\tilde{Z}$. However, (4.5) suggests that we use some simple approximation to $E = Z^TZ$ in order to generate successive approximations to A_α^{-1}. A possible approach would be to set up a constrained minimization problem of the form

$$\min_{E \in \mathcal{S}} \|I - LEL^T\|_F,$$

where $\mathcal{S}$ is a set of matrices E such that $D + \alpha E$ is SPD and "easy to invert". Here $||\cdot||_F$ denotes the Frobenius norm. Note that this problem, although quite expensive to solve, would have to be dealt with only once, since there is no dependency on α. However, in the context of approximate inverse preconditioning we do not have access to L, but rather to (an approximation of) L^{-1}. A more viable approach, which uses only information available from the already computed factor Z, is the following. For $k \geq 1$ define the *order k modified preconditioner* as

$$P_\alpha^{-1} := Z(D + \alpha E_k)^{-1}Z^T, \tag{4.6}$$

where E_k is the symmetric positive definite band matrix given by

$$E_k = Z_k^TZ_k, \tag{4.7}$$

and Z_k is obtained by extracting the $k-1$ upper diagonals from Z if $k > 1$ or $E_1 = \text{diag}(Z^TZ)$ if $k = 1$. Thus, E_1 corresponds to the *order 1 preconditioner*, while the symmetric tridiagonal band matrix E_2 corresponds to the *order 2 preconditioner*. It is worth to note that E_k is always positive definite since

Z_k is a unit upper triangular matrix and therefore nonsingular. Therefore, the modified approximate inverse is guaranteed to be positive definite. To complete the hierarchy of approximations, we define the *order* -1 *preconditioner* by letting $E = 0$ (which corresponds to just using $P^{-1} = A^{-1}$ as an approximation of A_α^{-1}) and the *order 0 preconditioner* by letting $E = I$. This approach is motivated by the observation that under suitable assumptions, the entries along the rows of Z decay away from the main diagonal [9, 14, 5]; hence, banded approximations of Z tend to contain most of the large entries in Z.

Typically, only small values of k are viable. To form E_k we need to get the entries in the first k diagonals in the upper triangular part of Z and to compute the product $Z_k^T Z_k$. This needs to be done only once, as the entries of Z are independent of α. Applying the order k preconditioner $Z(D+\alpha E_k)^{-1}Z^T$ within a step of the conjugate gradient algorithm requires, besides multiplication by Z_k and its transpose, the solution of banded linear systems of the form

$$(D + \alpha E_k)u = v. \tag{4.8}$$

For $k \geq 2$, a banded Cholesky factorization of the matrix $D + \alpha E_k$ must be computed (for each new value of α), and the linear system (4.8) is solved by forward and backward substitution. These are, again, $\mathcal{O}(kn)$ operations for the forward/backward substitutions per iteration and $\mathcal{O}(k^2 n)$ for the factorization (before starting the iteration). On a parallel computer, a parallel band solver would be required; see, e.g., [10].

In the numerical experiments we consider preconditioners of order 0, 1 and 2. However, there may be specific cases where higher order preconditioners are well-suited. In practice, we found that the best results are often obtained with the preconditioners of order 0 and 1. In the following, we analyze the order 0 preconditioner in some detail. Higher order preconditioners can be analyzed along similar lines.

4.2 Theoretical justification.

We will consider the following expressions:

$$P_\alpha - (A + \alpha\, I), \quad P_\alpha^{-1}(A + \alpha\, I) - I. \tag{4.9}$$

It is worth repeating that for the purpose of this analysis, we are assuming that Z and D are computed exactly; that is, $ZD^{-1}Z^T = A^{-1}$. Also, A has been normalized so that its largest entry (which is necessarily on the main diagonal) is equal to 1.

THEOREM 4.1. *Let $A = LDL^T$ be a normalized SPD matrix of order n, and let $A_\alpha = A + \alpha I$, $\alpha \in (0, \alpha_{\max})$. Assume that, fixed $\delta \in (0,1)$ suitably small, s of the eigenvalues λ_i of the matrix $L\,L^T - I$ satisfy $|\lambda_i| \leq \delta$, with $n - s = k \ll n$. Also, let $P_\alpha^{-1} = Z(D+\alpha I)^{-1}Z^T$ where $Z = L^{-T}$. Then, there exist matrices F, Δ and a constant $0 < c_1 < 1$ such that*

$$P_\alpha^{-1}(A + \alpha\, I) = I + F + \Delta, \quad ||F||_2 \leq c_1 \delta ||Z||_2^2,$$

and $\operatorname{rank}(\Delta) = \operatorname{rank}(\Delta_1) = k \ll n$. *Moreover,* k *and* c_1 *do not depend on* α.

PROOF. Consider the difference

$$P_\alpha - A_\alpha = Z^{-T}\,(D + \alpha I)\,Z^{-1} - (A + \alpha I) = \alpha(L\,L^T - I). \tag{4.10}$$

Clearly, how well P_α approximates A_α depends on the size of α and on how far the symmetric matrix LL^T is from the identity matrix.

Let us consider now the preconditioned matrix $P_\alpha^{-1}A_\alpha$ and the identity

$$A_\alpha = P_\alpha + \alpha(I - LL^T).$$

Let the eigenvalues $\lambda_1, \ldots, \lambda_n$ of $K = I - LL^T$ be such that

$$0 \le |\lambda_1| \le \ldots \le |\lambda_s| \le \delta < |\lambda_{s+1}| \le \ldots \le |\lambda_n|, \quad K = U\mathrm{diag}\,(\lambda_1, \ldots, \lambda_n)U^T,$$

where U is an orthogonal matrix which diagonalizes K. Moreover, let F_1 and Δ_1 be such that $F_1 + \Delta_1 = I - L\,L^T$, with

$$F_1 = U\mathrm{diag}\,(\lambda_1, \ldots, \lambda_s, 0, \ldots, 0)U^T, \quad \Delta_1 = U\mathrm{diag}\,(0, \ldots, 0, \lambda_{s+1}, \ldots, \lambda_n)U^T.$$

We have that

$$||F_1||_2 \le \delta, \quad \operatorname{rank}(\Delta_1) \le n - s = k$$

and the preconditioned matrix $P_\alpha^{-1}(A + \alpha I)$ can be written as

$$P_\alpha^{-1}A_\alpha = I + \alpha P_\alpha^{-1}(I - LL^T) = I + \alpha(P_\alpha^{-1}F_1 + P_\alpha^{-1}\Delta_1) = I + F + \Delta, \tag{4.11}$$

where $\operatorname{rank}(\Delta) = \operatorname{rank}(\Delta_1) = n - s = k \ll n$.

To prove the upper bound for the 2-norm of $F = \alpha P_\alpha^{-1}F_1$, let us consider first $||P_\alpha^{-1}||_2$. We have:

$$\begin{aligned} ||P_\alpha^{-1}||_2 &= ||Z(D + \alpha I)^{-1}Z^T||_2 \le ||Z||_2^2||(D + \alpha I)^{-1}||_2 \\ &= ||Z||_2^2 \cdot \max_i \left\{|d_i + \alpha|^{-1}\right\}. \end{aligned} \tag{4.12}$$

By virtue of the well-known properties of the Rayleigh quotient, the diagonal elements $d_i = z_i^T A z_i$, $i = 1, \ldots, n$, of the matrix D are bounded as follows:

$$\lambda_{\min}(A)||z_i||_2^2 \le d_i \le \lambda_{\max}(A)||z_i||_2^2, \quad ||z_i||_2 \ge 1, \quad i = 1, \ldots, n. \tag{4.13}$$

Therefore, by setting

$$c_1 = \left(\frac{\lambda_{\min}(A)}{\alpha_{\max}} + 1\right)^{-1}, \tag{4.14}$$

from (4.12) and (4.13), we have

$$||P_\alpha^{-1}||_2 \le ||Z||_2^2 \frac{1}{\lambda_{\min}(A) + \alpha} \le c_1 \frac{||Z||_2^2}{\alpha}, \tag{4.15}$$

where, by (4.14) and the hypotheses,

$$0 < c_1 = \left(\frac{\lambda_{\min}(A)}{\alpha_{\max}} + 1\right)^{-1} < 1, \quad 0 < \frac{\lambda_{\min}(A)}{\alpha}.$$

Therefore, $c_1 < 1$ does not depend on α and it can be $c_1 \ll 1$ if $\alpha_{\max} \ll \lambda_{\min}(A)$, say. Finally, from (4.15), we have

$$||F||_2 = \alpha ||P_\alpha^{-1} F_1||_2 \leq \alpha c_1 \delta \frac{||Z||_2^2}{\alpha} = c_1 \delta ||Z||_2^2. \tag{4.16}$$

Notice that the dependence on the shift parameter α disappears in (4.11) (but not in (4.10)). □

We observe that the same argument as above can be used to establish similar results for order $k > 0$ preconditioners. Furthermore, as observed by one of the referees, this result can be extended to cover the more general case of shifted matrices of the type $A + \alpha E$, where E is an arbitrary SPD matrix.

As is well known, the preconditioned conjugate gradient (PCG) method converges rapidly when the preconditioned matrix differs from the identity by a matrix that can be written as the sum of a matrix of small norm and one of small rank. Hence, it follows from Theorem 4.1 that if $||Z||_2$ is not too large and/or c_1 is small, rapid convergence can be expected. In general, however, $||Z||_2$ could be arbitrarily large.

One case where we can expect the underlying preconditioners to be effective is when the entries of Z decay fast enough from the main diagonal. To this end, in [5] the authors prove exponential decay bounds for the entries of $ZD^{-1/2}$, $Z = (z_{i,j})$ for A symmetric and positive definite.

Theorem 4.2. *Let A be SPD and normalized, $A^{-1} = ZD^{-1}Z^T$. Then for $j > i$ we have*

$$|z_{i,j}| \leq \sqrt{d_j}\, c_2\, t^{j-i}, \text{where } c_2 = \frac{1-t^n}{1-t} \max\left\{\lambda_{\min}^{-1}(A), \frac{(1+\sqrt{\kappa})^2}{2\lambda_{\min}(A)\kappa}\right\}, \tag{4.17}$$

$$t = \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{\frac{1}{n}}, \ \kappa = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}, \ 1 \leq i < j \leq n.$$

Proof. Follows from Theorem 4.1 in [5]. □

For a banded A, the upper bound on the entries of Z can be improved considerably. In particular when the bandwidth and the condition number of A are not too large, then the entries of Z are bounded in a rapidly decaying fashion away from the main diagonal along rows. Thus $||Z||_2^2 \leq c_3$, where c_3 is of the order of unity. By (4.15), (4.17), defining $c_4 = c_1 c_3$, we have $||P_\alpha^{-1}||_2 \leq c_4/\alpha$. Therefore, by (4.11), we have $||F||_2 \leq c_4\delta$ and the influence of the term $\alpha(I - LL^T)$ can be moderate in (4.2) for large values of α as well. We stress that we will consider α "large" for the given matrix if there is no need to precondition the corresponding shifted linear system.

Corollary 4.3. *Let A be a normalized SPD diagonally dominant matrix. Then the preconditioned matrix $P_\alpha^{-1}A_\alpha$ has clustered spectrum.*

Proof. If A is normalized so that $|a_{i,j}| \leq 1$, then $|l_{i,j}| \leq 1$. Moreover, by using the bound in (4.17) and the diagonal dominance in Theorem 4.1, it is easy

Table 5.1: Test results for NOS5. $n = 468$, $nnz = 5172$, $\kappa \approx 1.1e+04$, $zfill = 2618$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	% fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	40	2.37	–	–	–	–	–	–	–	–
1.49e-5	1	40	2.37	40	1.2	40	**1.07**	40	**1.07**	40	**1.07**
2.38e-4	.97	30	2.08	30	.9	30	**.81**	30	**.81**	30	**.81**
1.5e-3	.84	23	1.69	19	.57	19	**.52**	19	**.52**	20	.55
2.4e-1	.33	7	**.68**	51	1.51	55	1.47	53	1.41	61	1.63
2.4e-1*	.33	7	.68	13	**.24**	13	**.24**	13	**.24**	43	.78

to see that the entries of LL^T decay fast away from the main diagonal. Therefore $k = 0$, $\Delta_1 = \Delta = 0$ in the decomposition (4.11), and $LL^T - I$ can be regarded as a small norm approximation of the null matrix in (4.10). □

5 Numerical experiments.

In this section we present the results of some numerical tests with a Matlab implementation of the proposed techniques. We limit ourselves to modified preconditioners of order 0, 1 and 2. These low-order modifications are compared with the "full" SAINV preconditioner (i.e., the SAINV preconditioner is recomputed from scratch for each different α) and with the "order -1" preconditioner, which is just the preconditioner computed for $\alpha = 0$.

Since diagonally dominant matrices are easily handled (as expected from Corollary 4.3), we focus for the most part on SPD matrices that are not diagonally dominant. The test matrices are all available from the Matrix Market [16] with the exception of GEO, which is part of a problem from [11], and of DISCDIFF, which is from [15, p. 425]. This is a finite difference discretization of the diffusion equation (2.1) on $[0,1] \times [0,1]$ with discontinuous coefficients. The diffusivity coefficient D is 1000 in $[1/4, 3/4] \times [1/4, 3/4]$ and 1 elsewhere. Except for DISCDIFF and 1138BUS (also used in [15]), the test problems used here are generally more difficult than those considered by Meurant in [15].

In all the experiments the initial guess for the conjugate gradient iteration was the null vector, while the stopping criterion was $||r_k||_2 < 10^{-6}||r_0||_2$, where r_k denotes the true residual after k iterations. In the tables, a "†" indicates no convergence within 1000 iterations. In all tests, the SAINV preconditioner was computed with drop tolerance $\varepsilon = 0.1$. No attempt was made to tune ε for optimal performance. The matrices were normalized so that $\max_i\{a_{ii}\} = 1$, and the original ordering was used. The values used for the shift α are similar to the (small) values used by Meurant [15].

In the tables we denote the modified SAINV preconditioner of order k by SAINV$_k$, for $k = 2, 1, 0$. The full SAINV preconditioner is denoted by "full SAINV" and the order -1 preconditioner by "SAINV(A)". For each preconditioner we report (under "It") the number of PCG iterations for some significant values of α, similar to those used by Meurant [15]. Under "% fill" we report the amount of fill in the approximate inverse factor $\tilde{Z}$, normalized with respect to the number of nonzeros in $\tilde{Z}$ corresponding to $\alpha = 0$. The total number of floating point operations (in Megaflops) is reported under "Mf", and it includes both the work for computing the preconditioner and the work for the iteration phase. Note that for the modified preconditioners, the former is always negligible. In the caption of each table we report the order n of the matrix A, the number nnz of nonzero entries in the lower triangular part of A, an estimate κ of its condition number, and the number $zfill$ of nonzeros in the $\tilde{Z}$ factor computed for $\alpha = 0$.

In some of the tables, we have in addition experimented with using $\tilde{Z} = I$ in (4.4) for some of the (relatively large) values of α, denoted by the superscript $*$. The motivation is that for some of the matrices the decay of the entries of Z is very slow (or absent). For these problems, the use of modified preconditioners reusing the $\tilde{Z}$ computed for $\alpha = 0$ is not suitable for larger values of α. On the other hand, setting $\tilde{Z} = I$ works quite well in several cases.

Table 5.2: Test results for 1138BUS. $n = 1138$, $nnz = 2596$, $\kappa \approx 3.84e + 04$, $zfill = 5462$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	%fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	61	9.21	–	–	–	–	–	–	–	–
1.49e-5	.95	47	8.27	43	2.25	43	**1.89**	45	1.98	46	2.03
2.38e-4	.72	25	6.05	35	1.84	35	**1.55**	45	1.98	70	3.08
1.5e-3	.50	17	4.37	49	2.56	50	**2.2**	54	2.31	120	5.26
2.4e-1	.22	6	**2.24**	79	4.1	85	3.73	86	3.77	623	27.2
2.4e-1*	.22	6	2.24	18	.64	21	**.56**	21	**.56**	703	18.5

Table 5.3: Test results for GEO. $n = 729$, $nnz = 19027$, $\kappa \approx 3.08e + 08$, $zfill = 10459$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	%fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	121	21.77	–	–	–	–	–	–	–	–
1.49e-5	1.01	104	20	104	10	104	**9.28**	104	**9.28**	104	**9.28**
2.38e-4	.99	93	18.7	88	8.48	89	7.95	88	**7.86**	88	**7.86**
1.5e-3	.96	64	15.1	64	6.19	64	**5.73**	64	**5.73**	64	**5.73**
2.4e-1	.26	11	**3.15**	47	4.56	54	4.84	58	5.2	61	5.41
2.4e-1*	.26	11	3.15	25	**1.28**	25	**1.28**	25	**1.28**	27	1.38

For each value of $\alpha > 0$, the best results in terms of operation count are reported in boldface. There are several observations that can be made on the basis of these numerical experiments. First of all, there seems to be little reason to use the order 2 preconditioner, SAINV$_2$. The best results overall are obtained with

Table 5.4: Test results for BCSSTK09. $n = 1083$, $nnz = 9760$, $\kappa \approx 6.8e + 07$, $zfill = 8215$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	% fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	88	17.1	–	–	–	–	–	–	–	–
1.49e-5	1	87	17	87	8	87	**7.25**	87	**7.25**	87	**7.25**
2.38e-4	1	78	16.3	81	7.52	81	**6.76**	81	**6.76**	81	**6.76**
1.5e-3	1	56	14.4	56	5.2	56	**4.7**	56	**4.7**	56	**4.7**
2.4e-1	.34	12	4.5	18	1.72	18	**1.54**	19	1.63	19	1.63

Table 5.5: Test results for NOS3. $n = 960$, $nnz = 15844$, $\kappa \approx 3.7e + 04$, $zfill = 5860$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	79	11.6	–	–	–	–	–	–	–	–
1.49e-5	1	78	11.54	78	6	78	**5.25**	78	**5.25**	78	**5.25**
2.38e-4	1	73	11.2	73	5.61	73	**4.91**	73	**4.91**	73	**4.91**
1.5e-3	1	63	10.5	63	4.85	63	**4.25**	63	**4.25**	63	**4.25**
2.4e-1	.52	12	4.18	14	1.13	15	**1.04**	16	1.11	17	1.18

the order 1 preconditioner SAINV$_1$, with SAINV$_0$ often being equally valid. For very small values of α it is usually the case that good results are obtained by simply reusing the preconditioner already computed for A, denoted by SAINV(A) in the tables. Note, however, that there are exceptions; see the results for NOS6 in Table 5 and DISCDIFF in Table 5.

The modified preconditioners performed especially well for problems 1138BUS, DISCDIFF, NOS6, BCSSTK07 and BCSSTK27, in some cases allowing for large savings in total solution costs. This is especially encouraging for DISCDIFF, which is the most physically meaningful of the test problems considered here. Similar results were also observed for other diffusion-type problems.

The modified preconditioners also performed well for NOS5, GEO, BCSSTK09 and NOS3; in these cases, however, the "unmodified" preconditioner SAINV(A) also performed well. In any case, the modified preconditioners were not worse than the unmodified one.

6 Conclusions.

In this paper we have considered preconditioner modification strategies for shifted linear systems. We targeted general SPD matrices and the sparse approximate inverse preconditioner SAINV. This is not an easy problem in general, since the inverse of $A+\alpha I$ can be quite different from A^{-1} if A is ill-conditioned.

Table 5.6: Test results for BCSSTK07. $n = 420$, $nnz = 7860$, $\kappa \approx 3.5e + 09$, $zfill = 5779$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	70	6.6	–	–	–	–	–	–	–	–
1.49e-5	.86	32	3.2	29	1.4	29	**1.3**	31	1.39	32	1.43
2.38e-4	.45	14	1.82	39	1.9	41	1.82	35	**1.56**	38	1.7
1.5e-3	.29	10	**1.2**	91	4.3	104	4.59	92	4	77	3.4
1.5e-3*	.29	10	1.2	42	**.96**	42	**.96**	42	**.96**	73	1.65
2.4e-1	.09	6	**.5**	6	1.27	999	43.8	†	†	541	23.8
2.4e-1*	.09	6	.5	15	**.35**	15	**.35**	15	**.35**	217	4.9

Table 5.7: Test results for NOS6. $n = 675$, $nnz = 1965$, $\kappa \approx 7.7e + 06$, $zfill = 2358$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	%fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	25	2.4	–	–	–	–	–	–	–	–
1.49e-5	.73	21	1.8	30	.88	34	**.8**	37	.9	92	2.2
2.38e-4	.67	20	1.7	33	**.97**	40	**.97**	42	1	229	5.5
1.5e-3	.67	19	1.6	32	**.9**	37	**.9**	40	.97	309	7.4
2.4e-1	.4	9	1.1	22	**.67**	29	.7	29	.7	420	10

Indeed, note that

$$\frac{d}{d\alpha}(A + \alpha I)^{-1}_{|\alpha=0} = -A^{-2},$$

showing that the inverse of $A + \alpha I$ can be very sensitive around $\alpha = 0$ when A^{-2} has large entries, as it is to be expected if A is ill-conditioned.

We found that simply shifting the pivots in the SAINV approximate inverse factorization leads to surprisingly good results in many cases. We gave a theoretical justification of this fact for matrices that exhibit rapid decay away from the main diagonal in the inverse Cholesky factor.

It is likely that these techniques can be further improved upon by including modifications to the Z factor. For instance, if the decay rate of the elements of Z is very slow or there is no decay at all, we can consider modified order k preconditioners where the Z factor in (4.6) now varies with α.

Indeed, notice that for $\alpha \to \infty$, the conjugate gradient method preconditioned with (4.6) can converge faster if $\tilde{Z}_\alpha$ is used in (4.3) instead of a fixed $\tilde{Z}$, where $\tilde{Z}_\alpha$ is such that

$$\lim_{\alpha\to\infty} \tilde{Z}_\alpha = I.$$

This follows by using $\tilde{Z}_\alpha$ instead of Z in (4.9). As shown in some of our numerical experiments, simply using $\tilde{Z}_\alpha = I$ gives good results for relatively large values of α.

Table 5.8: Test results for DISCDIFF. $n = 900$, $nnz = 4380$, $\kappa \approx 3.5e + 05$, $zfill = 2856$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	%fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	59	5	–	–	–	–	–	–	–	–
1.49e-5	1	46	4.6	46	1.86	46	**1.4**	46	**1.4**	46	**1.4**
2.38e-4	1	18	3.7	19	.8	21	**.67**	21	**.67**	21	**.67**
1.5e-3	1	12	3.5	17	.7	21	**.66**	21	**.66**	24	.76
2.4e-1	.33	7	1.5	16	**.6**	22	.7	22	.7	127	3.95

Table 5.9: Test results for BCSSTK27. $n = 1224$, $nnz = 56126$, $\kappa \approx 3.5e + 06$, $zfill = 16451$.

α	full SAINV			SAINV$_2$		SAINV$_1$		SAINV$_0$		SAINV(A)	
	%fill	It	Mf	It	Mf	it	Mf	it	Mf	it	Mf
0	1	75	37.6	–	–	–	–	–	–	–	–
1.49e-5	.99	73	37	73	15	73	**14.2**	73	**14.2**	74	14.4
2.38e-4	.89	53	30.4	57	11.8	57	**11.1**	59	11.5	60	11.7
1.5e-3	.66	35	21.4	40	8.3	41	**8**	64	12.4	81	15.8
2.4e-1	.15	10	**5.4**	162	33.2	257	49.7	830	160	344	66.5
2.4e-1*	.15	10	5.4	22	**3**	22	**3**	22	**3**	330	43.6

For example, we could define $\tilde{Z}_\alpha$ as follows:

$$\tilde{Z}_\alpha = \begin{cases} (1-\alpha)\tilde{Z} + \alpha I & 0 \le \alpha < \beta, \\ I & \alpha \ge \beta, \end{cases}$$

where $\tilde{Z}$ is the matrix computed for (4.3). The threshold parameter β can be determined for specific classes of normalized matrices A by considering (4.2). For example, a reasonable choice could be $\beta = c \cdot 10^{-1}$, where c is a constant of the order of unity.

Finally, we mention briefly another possible approach based on the interpolation of two (or more) matrices $\tilde{Z}_{\alpha_r}$ related to the matrices A_{α_r} for different values of the shift parameter. For example, if $\tilde{Z}_1$, $\tilde{Z}_2$ are unit upper triangular matrices such that the following approximate inverse decompositions hold:

$$(A + \alpha_1 I)^{-1} \approx \tilde{Z}_1 \tilde{D}_1^{-1} \tilde{Z}_1^T, \quad (A + \alpha_2 I)^{-1} \approx \tilde{Z}_2 \tilde{D}_2^{-1} \tilde{Z}_2^T,$$

where $0 \le \alpha_1 < \alpha_2 < 1$, α_1, α_2 suitably chosen, we can use an order k preconditioner (4.6) with $Z = \tilde{Z}_\alpha$, where $\tilde{Z}_\alpha$ is defined as follows:

$$\tilde{Z}_\alpha = \begin{cases} \dfrac{\alpha_2 - \alpha}{\alpha_2 - \alpha_1}\tilde{Z}_1 + \dfrac{\alpha - \alpha_1}{\alpha_2 - \alpha_1}\tilde{Z}_2 & \alpha_1 \le \alpha \le \alpha_2, \\ I & \alpha > \alpha_2. \end{cases}$$

Such a preconditioner can be used to precondition matrices of the form $A + \alpha I$ where $\alpha \in (\alpha_1, \alpha_2)$.

Acknowledgements.

The second author would like to thank the Department of Mathematics and Computer Science at Emory University for the support, warm hospitality and excellent computing facilities offered in February 2002, when part of this work was completed. The authors would like to thank the referees and Daniel Szyld for helpful comments and useful suggestions which have improved the presentation.

REFERENCES

1. U. R. Ascher, R. M. M. Matteij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, SIAM, Philadelphia, PA, 1995.
2. M. Benzi, J. K. Cullum, and M. Tůma, *Robust approximate inverse preconditioning for the conjugate gradient method*, SIAM J. Sci. Comput. 22 (2000), pp. 1318–1332.
3. M. Benzi, R. Kouhia, and M. Tůma, *Stabilized and block approximate inverse preconditioners for problems in solid and structural mechanics*, Comput. Methods Appl. Mech. Engrg., 190 (2001), pp. 6533–6554.
4. M. Benzi, C. D. Meyer, and M. Tůma, *A sparse approximate inverse preconditioner for the conjugate gradient method*, SIAM J. Sci. Comput., 17 (1996), pp. 1135–1149.
5. M. Benzi and M. Tůma, *Orderings for factorized sparse approximate inverse preconditioners*, SIAM J. Sci. Comput., 21 (2000), pp. 1851–1868.
6. D. Bertaccini, *A circulant preconditioner for the systems of LMF-based ODE codes*, SIAM J. Sci. Comput., 22 (2000), pp. 767–786.
7. D. Bertaccini, *Reliable preconditioned iterative linear solvers for some numerical integrators*, Numer. Linear Algebra Appl., 8 (2001), pp. 111–125.
8. M. Bollhöfer and Y. Saad, *On the relations between ILUs and factored approximate inverses*, SIAM J. Matrix Anal. Appl., 24 (2002), pp. 219–237.
9. S. Demko, W. F. Moss, and P. W. Smith, *Decay rates for inverses of band matrices*, Math. Comp., 43 (1984), pp. 491–499.
10. J. J. Dongarra, I. S. Duff, D. C. Sorensen, and H. A. van der Vorst, *Numerical Linear Algebra for High-Performance Computers*, SIAM, Philadelphia, PA, 1998.
11. E. Haber, U. M. Ascher, and D. Oldenburg, *On optimization techniques for solving nonlinear inverse problems*, Inverse Problems, 16 (2000), pp. 1263–1280.
12. E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, Springer-Verlag, Berlin–Heidelberg, 1991.
13. S. A. Kharchenko, L. Yu. Kolotilina, A. A. Nikishin, and A. Yu. Yeremin, *A robust AINV-type method for constructing sparse approximate inverse preconditioners in factored form*, Numer. Linear Algebra Appl., 8 (2001), pp. 165–179.
14. G. Meurant, *A review on the inverse of symmetric tridiagonal and block tridiagonal matrices*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 707–728.
15. G. Meurant, *On the incomplete Cholesky decomposition of a class of perturbed matrices*, SIAM J. Sci. Comput., 23 (2001), pp. 419–429.
16. National Institute of Standards, *Matrix market*, available online at http://math.nist.gov/MatrixMarket.