Electronic Transactions on Numerical Analysis.
Volume 18, pp. 49-64, 2004.
Copyright © 2004, Kent State University.
ISSN 1068-9613.

ETNA
Kent State University
etna@mcs.kent.edu

EFFICIENT PRECONDITIONING FOR SEQUENCES OF PARAMETRIC COMPLEX SYMMETRIC LINEAR SYSTEMS [*]

D. BERTACCINI [†]

Abstract. Solution of sequences of complex symmetric linear systems of the form $A_j x_j = b_j$, $j = 0, ..., s$, $A_j = A + \alpha_j E_j$, A Hermitian, $E_0, ..., E_s$ complex diagonal matrices and $\alpha_0, ..., \alpha_s$ scalar complex parameters arise in a variety of challenging problems. This is the case of time dependent PDEs; lattice gauge computations in quantum chromodynamics; the Helmholtz equation; shift-and-invert and Jacobi–Davidson algorithms for large-scale eigenvalue calculations; problems in control theory and many others. If A is symmetric and has real entries then A_j is *complex symmetric*.

The case A Hermitian positive semidefinite, $\mathrm{Re}(\alpha_j) \geq 0$ and such that the diagonal entries of E_j, $j = 0, ..., s$ have nonnegative real part is considered here.

Some strategies based on the update of incomplete factorizations of the matrix A and A^{-1} are introduced and analyzed. The numerical solution of sequences of algebraic linear systems from the discretization of the real and complex Helmholtz equation and of the diffusion equation in a rectangle illustrate the performance of the proposed approaches.

Key words. Complex symmetric linear systems; preconditioning; parametric algebraic linear systems; incomplete factorizations; sparse approximate inverses.

AMS subject classifications. 65F10, 65N22, 15A18.

1. Introduction.

The numerical solution of several problems in scientific computing requires the solution of sequences of parametrized large and sparse linear systems of the form

$$A_j x_j = b_j, \quad A_j = A + \alpha_j E_j, \quad j = 0, ..., s \tag{1.1}$$

where $\alpha_j \in \mathbb{C}$ are scalar quantities and E_0,..., E_s are complex symmetric matrices. Here we will consider the case where E_j, $j = 0, ..., s$ are diagonal matrices. Among these problems we recall partial differential equations (PDEs) ([1, 6, 7, 16] (time-dependent PDEs using implicit difference schemes; the Helmholtz equation; the Schrödinger equation, etc.); large and sparse eigenvalues computation (see, e.g., [12]) and model trust region, nonlinear least squares problems and globalized Newton algorithms in optimization; see, e.g., [11].

In the above mentioned frameworks, sequences of linear systems with matrices as in (1.1), each one with a possibly different right hand side and initial guess, have to be solved. Therefore, it would be desirable to have a strategy for modifying 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.

In most of the above mentioned applications, the matrix A is large and sparse, and preconditioned Krylov subspace methods are used to solve the linear systems. Hence, there are several situations where it is desirable to use cost-effective modifications of an initial preconditioner for solving such a sequence of linear systems.

[*] Received December 12, 2003. Accepted for publication March 3, 2004. Recommended by M. Benzi.

[†] Università di Roma "La Sapienza", Dipartimento di Matematica "Istituto Guido Castelnuovo", P.le A. Moro 2, 00185 Roma, Italy. E-mail: bertaccini@mat.uniroma1.it, web: http://www.mat.uniroma1.it/∼bertaccini. This work was supported in part by the COFIN2002 project "Analisi di strutture di matrici: metodi numerici e applicazioni" and by Center for Modeling Integrated Metabolic Systems (MIMS), Cleveland (http://www.csuohio.edu/mims).

In this paper, we propose to solve (1.1) by Krylov methods using preconditioning strategies based on operators P_j approximating A_j such that

$$P_j = \tilde{A} + \alpha_j K_j, \quad j = 0, ..., s, \tag{1.2}$$

where $\tilde{A}$ is an operator which approximates A in (1.1), $\alpha_0, ..., \alpha_s$ are complex scalar parameters and K_j, $j = 0, .., s$, are suitable corrections related to E_j, $j = 0, .., s$. In particular, we propose strategies to update incomplete factorizations with threshold for A and A^{-1}. The correction matrices K_j are computed by balancing two main requirements: the updated preconditioners (1.2) must be a cheap approximation for A_j and $||A_j - P_j||$ must be small.

In the paper [5], we proposed an updated preconditioner for sequences of *shifted* linear systems

$$(A + \alpha_j I)x_j = b_j, \quad \alpha_j \in \mathbb{C}, \; j = 0, ..., s, \tag{1.3}$$

based on approximate inverse factorization which is effective when A is symmetric positive definite and α is real and positive. Here, these results will be extended to more general perturbations of the originating (or *seed*) matrix A (see (1.1)), i.e., matrices A_j which are complex symmetric, and in the use of incomplete factorization as a basis of the updates. We stress that an update for the incomplete Cholesky factorization for shifted linear systems using a different technique was proposed in [17]. Note, however, that the standard incomplete Cholesky factorization is not entirely reliable even when applied to general positive definite matrices. On the other hand, our algorithms can be based on more robust preconditioners.

1.1. Krylov methods and shifted linear systems. Let us consider now the special case where $E_j \equiv I$, $j = 0, ..., s$, the initial guesses $x_j^{(0)}$ are all equal to the zero vector and $b_j = b$, $j = 0, ..., s$. Then, the family of linear systems (1.1) can be written as

$$(A + \alpha_j I)x_j = b, \quad \alpha_j \in \mathbb{C}, \; j = 0, ..., s, \tag{1.4}$$

and each system in (1.4) generates the same Krylov subspace $\mathcal{K}_m(A_j, r_j^{(0)})$ because

$$\mathcal{K}_m(A_j, r_j^{(0)}) = \mathcal{K}_m(A + \alpha_j I, r_j^{(0)}) = \mathcal{K}_m(A, r^{(0)}), \tag{1.5}$$

$$r^{(0)} = b - Ax^{(0)}, \; \alpha_j \in \mathbb{C}, \; j = 0, ..., s,$$

see, e.g., [15]. Therefore, the most expensive part of the Krylov method for solving simultaneously the $s+1$ linear systems (1.4), the computation of the Krylov basis, can be computed only once. Note that the corresponding iterates and the residuals for each of the underlying linear systems can be computed without further matrix-vector multiplications; this kind of approach is the most popular in the literature. Another possibility is restating the problem as the solution of a linear system $AX = B$ where $X = [x_0 \; \cdots \; x_s]$ and $B = [b_0 \; \cdots \; b_s]$. This approach can solve multiple linear systems whose right hand side differ; see [19].

1.2. Why not preserve the Krylov subspace. There are many problems where the approaches in section 1.1 cannot be used. First, when $E_j \neq I$ for at least a $j \in \{0, ..., s\}$ in (1.1). Second, problems whose initial residuals $r_j^{(0)}$ are not collinear (example: right hand sides and/or the initial guesses are nonzero and/or differ) or $x_j^{(0)}$ and/or the right hand sides are not all available at the same time. Moreover, the linear systems in (1.1) (and, in particular, in (1.4)), can be ill-conditioned. Therefore, the underlying Krylov solver can converge very slowly. Unfortunately, the only preconditioners which preserve the same Krylov subspace (1.5) for all j, $j = 0, ..., s$ (i.e., after the shift) are polynomial preconditioners, but they are not competitive with the approaches based on incomplete factorization. In particular, performing our numerical experiments for (1.1), we experienced that often it is more convenient to use the same preconditioner computed for A, which does not take care of the perturbations E_j, instead of a more sophisticated polynomial preconditioner. Note that polynomial preconditioning for shifted linear systems as in (1.4) has been considered in [13]. On the other hand, if the matrix A_j in (1.1) is highly indefinite (i.e., there are more than few eigenvalues on both half complex plane), the approach proposed here using the standard $ILUT$ or incomplete LDL^H-threshold factorizations can fail and the use of Krylov methods for shifted linear systems can work better (in the framework (1.4), same initial guesses). However, an incomplete factorization for indefinite linear systems can be used to define an updated preconditioner for particular problems using our strategies. This will be considered in a future work.

Finally, we stress that the approaches considered in section 1.1 must be used in conjunction with (non-restarted) Krylov methods only. On the other hand, our algorithms do not have such restrictions. In particular, they can be used with restarted or oblique projection methods (e.g., restarted GMRES or any BICGSTAB) or with some other methods as well.

The paper is organized as follows. In Section 2 we give more comments on the solution of linear systems (1.1); Section 3 outlines our updated incomplete factorizations and their use as a preconditioner for the sequences of linear systems (1.1); in Section 4 we propose an analysis of the preconditioners and in Section 5 some numerical experiments. Finally, conclusions are given in Section 6.

2. Complex symmetric preconditioning.

Consider the sequence of linear systems (1.1) where A is a Hermitian large, sparse, possibly ill-conditioned matrix, b_j are given right-hand side vectors, α_j are (scalar) shifts, E_j are complex symmetric matrices and x_j are the corresponding solution vectors, $j = 0, ..., s$. The linear systems (1.1) 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 the case of time-dependent PDEs.

If $|\alpha_j|$ in (1.1) is large, then an effective preconditioning strategy is based on

$$P_j = \alpha_j E_j.$$

Indeed, we have that

$$(\alpha_j E_j)^{-1} A_j = I + \Delta, \quad \frac{||\Delta||}{||A_j||} \le \delta$$

and δ is small, $\delta \to 0$ for $|\alpha_j| \to \infty$. Moreover, preconditioning for the shifted problems in [5], where $\alpha \ge 0$, $E_j = I$ for all j and the matrices A were all symmetric positive definite, was found no longer beneficial as soon as α_j was of the order of 10^{-1}. On the other hand, we found cases where reusing a preconditioner computed for A for α_j of the order of 10^{-5} was ineffective. Therefore, by considering the above arguments and recalling that generating an approximation directly from the complex symmetric matrix $A_j = A + \alpha_j E_j$ requires complex arithmetic, suggests that algorithms computing approximations for A_j or A_j^{-1} working in real arithmetic can reduce significantly the overall computational cost.

3. The updated incomplete factorizations used as preconditioners.

For simplicity of notation, we will consider a generic shift α and parametric matrix E dropping the subscripts.

We propose updates for generating approximations for A_j as in (1.1) using approximations initially computed for the seed matrix A. The proposed preconditioners are based on incomplete factorizations (see, e.g., [18] and references therein) and approximate inverse factorizations of the type described in [2].

Let us suppose that A is Hermitian positive definite and that there exists an incomplete LDL^H factorization defined as follows

$$P = \tilde{L}\tilde{D}\tilde{L}^H \tag{3.1}$$

such that

$$A = LDL^H \simeq \tilde{L}\tilde{D}\tilde{L}^H, \tag{3.2}$$

where L, $\tilde{L}$ are unit lower triangular and D, $\tilde{D}$ are diagonal matrices, as usual. For later use, let us consider the (exact) inverse factorization of A such that

$$A^{-1} = ZD^{-1}Z^H \;\Rightarrow A = Z^{-H}DZ^{-1} \;\Rightarrow L = Z^{-H},\; L^H = Z^{-1}. \tag{3.3}$$

Moreover, let us suppose that αE has diagonal entries with positive real part. The proposed preconditioner based on (3.2) for $A_{\alpha,E} = A + \alpha E$ is given by

$$P_{\alpha,E} = \tilde{L}\left(\tilde{D} + \alpha B\right)\tilde{L}^H. \tag{3.4}$$

Note that it is customary to look for good preconditioners in the set of matrices approximating $A_{\alpha,E}$ which minimize some norm of $P_{\alpha,E} - A_{\alpha,E}$; see, e.g, [6, 7, 8].

Analogously, with a slight abuse of notation, we define the preconditioner for (1.1) based on the approximate inverse preconditioner for A

$$Q \equiv P^{-1} = \tilde{Z}\tilde{D}^{-1}\tilde{Z}^H \tag{3.5}$$

as

$$Q_{\alpha,E} \equiv P_{\alpha,B}^{-1} = \tilde{Z}\left(\tilde{D} + \alpha B\right)^{-1}\tilde{Z}^H, \tag{3.6}$$

where the component matrices in (3.5) can be computed by the algorithm in [2], which is robust if A is a Hermitian positive (or negative) definite matrix. Of course, $\tilde{D} + \alpha B$ must be invertible.

The following result is a generalization of (4.5) in [5].

Proposition 3.1. *Let us consider the exact LDL^H-factorization of A. By defining $P_{\alpha,E} = L\left(D + \alpha B\right)L^H$ and choosing $B = Z^H E Z$ we have $P_{\alpha,E} \equiv A_{\alpha,E}$, independently of α.*

Proof. Defining $P_{\alpha,E} := L\left(D + \alpha B\right)L^H$, i.e., using the exact LDL^H factorization of A, we have:

$$P_{\alpha,E} - A_{\alpha,E} = Z^{-H}\left(D + \alpha B\right)Z^{-1} - \left(Z^{-H}DZ^{-1} + \alpha E\right) = \alpha\left(LBL^H - E\right), \tag{3.7}$$

and therefore

$$B = L^{-1}EL^{-H} = Z^H EZ.$$

□

Unfortunately, we cannot use the form of B suggested in Proposition 3.1 in practice. Indeed, in general, only an incomplete LDL^H-factorization of A (or an incomplete $ZD^{-1}Z^H$-factorization of A^{-1}) is available, i.e., we don't have L to generate $L^{-H} = Z$ or directly Z but only (sparse) approximations. On the other hand, if the exact Z was available, we would have to solve, at each iteration step of the Krylov subspace solver, a linear system whose matrix $D + \alpha Z^H EZ$ is usually full.

Similar to the approach in [5], for $k \geq 1$, we define the *order k modified preconditioner* (or the *order k updated preconditioner*) as

$$P^{(k)}_{\alpha,E} := \tilde{L}(\tilde{D} + \alpha B_k)\tilde{L}^H \tag{3.8}$$

with the same notation as in (3.4) and where B_k is the symmetric positive definite band matrix given by

$$B_k = \tilde{Z}_k^T\, E\, \tilde{Z}_k. \tag{3.9}$$

$\tilde{Z}_k$ is obtained by extracting the $k-1$ upper diagonals from $\tilde{Z}$ if $k > 1$ or $B_1 = \mathrm{diag}(\tilde{Z}^H E\tilde{Z})$ if $k = 1$.

Similarly, we define the *order k (inverse) modified preconditioner* (or the *order k (inverse) updated preconditioner*) as

$$Q^{(k)}_{\alpha,E} := \tilde{Z}(\tilde{D} + \alpha B_k)^{-1}\tilde{Z}^H. \tag{3.10}$$

Note that:

- B_k is always nonsingular since $\tilde{Z}_k$ is a unit upper triangular matrix and therefore nonsingular and αE is diagonal whose entries have positive real part.
- From the previous item, $\tilde{D} + \alpha B_k$ is nonsingular. Indeed, $\alpha E = \mathrm{Re}(\alpha E) + \mathbf{i}\mathrm{Im}(\alpha E) = E_R + \mathbf{i}E_I$, where E_R is a diagonal matrix whose real entries are non negative by hypothesis. Therefore, we can write

  $$\tilde{D} + \alpha B_k = \left(\tilde{D} + \tilde{Z}_k^H E_R \tilde{Z}_k\right) + \mathbf{i}\tilde{Z}_k^H E_I \tilde{Z}_k.$$

  Let us consider the matrix in brackets. Recalling that $\tilde{Z}_k$ is nonsingular, we have that $\tilde{Z}_k^H E_R \tilde{Z}_k$ is positive semidefinite and $\tilde{D}$ has real positive entries. By using Gershgorin circles, we have the thesis.
- The *order k* preconditioners $P^{(k)}_{\alpha,E}$ based on incomplete LDL^H-threshold factorization with $k \geq 1$ require the computation of the matrix $\tilde{Z}$ or of an approximation of Z by using $\tilde{L}$. Note that the computation of an approximation of Z for computing B_k can be done just once for all scalars α and matrices E in (1.1). Note that the algorithm described in [3] generates $\tilde{L}$ and $\tilde{Z}$ at the same time.

Whenever $E = I$, the choice $B = I$ in (3.4), (3.6) is certainly effective if $L\,L^H - I$ is a small norm perturbation of the null matrix. In practice, the above condition occurs if the entries along the rows of Z in (3.3) decay rapidly away from the main

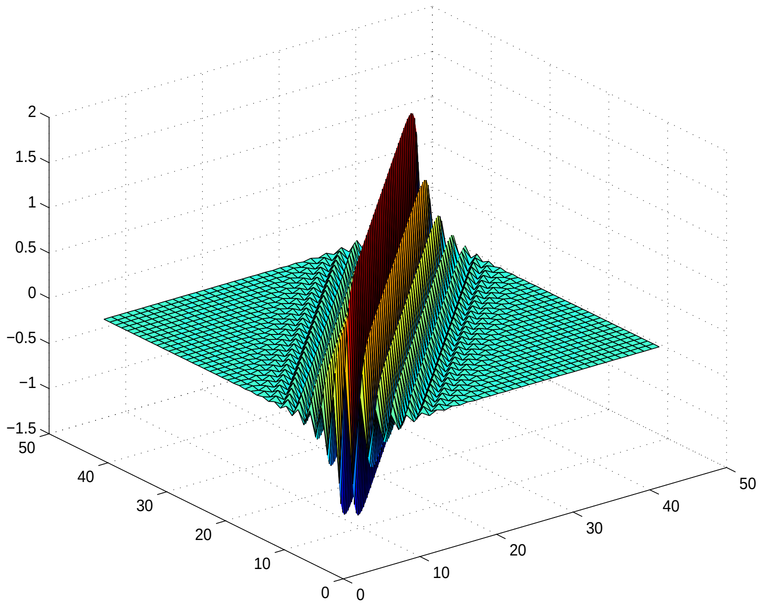

FIG. 3.1. *Values of the entries for* $(\text{toeplitz}(1, 2.1, 1)))^{-1}$.

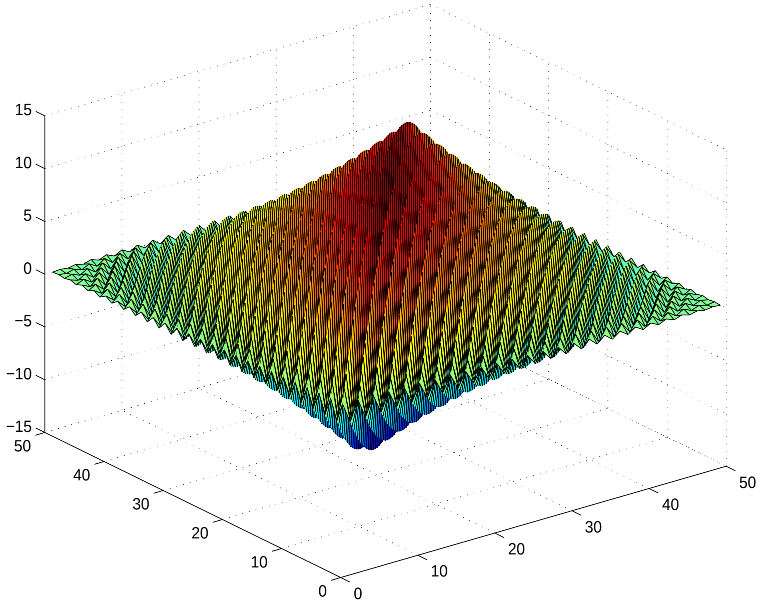

FIG. 3.2. *Values of the entries for* $(\text{toeplitz}(1, 2, 1)))^{-1}$.

diagonal. This happens if, e.g, A is strictly diagonally dominant; see Corollary 4.3 below and Figure 3.1. On the other hand, the matrix related to Figure 3.2 is only weakly diagonally dominant and the entries of the inverse of A decay much more slowly with respect to those of the matrix in Figure 3.1. However, we observed fast convergence of preconditioned iterations even for matrices that are not diagonally dominant. This is the case of most of the test problems considered in [5].

Let $\mathrm{Re}(M) = (M + \overline{M})/2$ and $\mathrm{Im}(M) = (M - \overline{M})/(2\mathbf{i})$.

PROPOSITION 3.2. *Let the incomplete factorization $\tilde{L}\tilde{D}\tilde{L}^H$ in (3.1) (the approximate inverse factorization $\tilde{Z}\tilde{D}^{-1}\tilde{Z}^H$ in (3.5)) be positive definite. If E (E_j in (1.1)) is such that $\mathrm{Re}(\alpha E)$ is positive definite, then $P^{(k)}_{\alpha,E}$ in (3.8) ($Q^{(k)}_{\alpha,E}$ in (3.10)) is nonsingular. Moreover, $\mathrm{Re}(P^{(k)}_{\alpha,E})$ ($\mathrm{Re}(Q^{(k)}_{\alpha,E})$) is positive definite.*

Proof. The result follows from (3.8), (3.10) by observing that $\mathrm{Re}(\tilde{D} + \alpha B_k)$ is positive definite for $k \geq 0$, the incomplete factorization (3.2) for the seed matrix A (the preconditioner based on the approximate inverse preconditioner (3.5)) is positive definite and therefore $P^{(k)}_{\alpha,E}$ ($Q^{(k)}_{\alpha,E}$) is nonsingular. □

4. Convergence of iterations. In the following analysis we consider the (exact) LDL^H factorization of A and the (exact) $ZD^{-1}Z^H$ factorization of the inverse for the matrix A in (1.1), where $Z = L^{-H}$. The results below extend those formerly given in [5] for symmetric positive definite matrices, α_0,...α_s real and positive and $E_j = I$ for all j.

THEOREM 4.1. *Let us consider the sequence of algebraic linear systems in (1.1). Let A be Hermitian positive definite, $\alpha \in \mathbb{C}$. Moreover, let $\delta > 0$ be a constant such that the singular values of the matrix $E - LB_kL^H$, B_k as in (3.9), are as follows*

$$\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_t \geq \delta \geq \sigma_{t+1} \geq ... \geq \sigma_n \geq 0,$$

and $t \ll n$. Then, if

$$\max_{\alpha \in \{\alpha_0,\ldots,\alpha_s\}} |\alpha| \cdot ||D^{-1}Z_k^H E Z_k||_2 \leq 1/2, \tag{4.1}$$

there exist matrices F, Δ and a constant c_α such that

$$\left(P^{(k)}_{\alpha,E}\right)^{-1}(A + \alpha\, E) = I + F + \Delta, \tag{4.2}$$

$$||F||_2 \leq \frac{2\max_{\alpha\in\{\alpha_0,\ldots,\alpha_s\}} |\alpha| c_\alpha \delta}{\lambda_{\min}(A)} \left(\frac{||Z||_2}{\min_i ||z_i||_2}\right)^2, \; Z = [z_1 \; \cdots \; z_n], \; z_i \in \mathbb{C}^n,$$

$\mathrm{rank}(\Delta) \leq t \ll n$, *the rank of Δ does not depend on α, c_α is a constant such that* $\lim_{|\alpha|\to 0} c_\alpha = 1$, *$c_\alpha$ of the order of unity. The same properties hold true for*

$$Q^{(k)}_{\alpha,E} \cdot (A + \alpha E),$$

with $Q^{(k)}_{\alpha,E}$ defined as in (3.10).

Proof. The matrix

$$E - LB_kL^H = \frac{1}{\alpha}\left(P^{(k)}_{\alpha,E} - A_{\alpha,E}\right)$$

(see (3.7)) can be decomposed as $F_1 + \Delta_1$, where, if $U\Sigma V^H$, $\Sigma = \mathrm{diag}(\sigma_1, ..., \sigma_n)$, is its singular value decomposition, we have:

$$\Delta_1 = U\mathrm{diag}\;(\sigma_1, \ldots, \sigma_t, 0, \ldots, 0)V^H, \;\; F_1 = U\mathrm{diag}\;(0, \ldots, 0, \sigma_{t+1}, \ldots, \sigma_n)V^H.$$

To simplify the notation, from here on we remove the superscript denoting the order of the preconditioner $P_{\alpha,E}$. We have that

$$\mathrm{rank}(\Delta_1) \leq t, \quad ||F_1||_2 \leq \delta.$$

Therefore, by observing that

$$P_{\alpha,E}^{-1}(A+\alpha E) = I + \alpha P_{\alpha,E}^{-1} F_1 + \alpha P_{\alpha,E}^{-1}\Delta_1,$$

and defining

$$F = \alpha P_{\alpha,E}^{-1} F_1, \quad \Delta = \alpha P_{\alpha,E}^{-1}\Delta_1,$$

we have (4.2), where $\operatorname{rank}(\Delta) = \operatorname{rank}(\Delta_1) \leq t \ll n$.

The matrix $D + \alpha B$ is nonsingular because

$$|\alpha|\, ||D^{-1}B||_2 \leq 1/2.$$

Indeed,

$$D + \alpha B = D(I + \alpha D^{-1}B) = D(I + \alpha D^{-1} Z_k^H E Z_k) = D(I + Y)$$

and $||Y||_2 \leq 1/2$.

By observing that

$$||P_{\alpha,E}^{-1}||_2 = ||Z(D+\alpha B)^{-1} Z^H||_2 \leq ||Z||_2^2 ||(D+\alpha B)^{-1}||_2, \tag{4.3}$$

$||P_{\alpha,E}^{-1}||$ can be bounded by

$$\begin{aligned} ||P_{\alpha,E}^{-1}|| &\leq ||Z||^2 \cdot c_\alpha \cdot \max_r \left\{|d_r|^{-1}\right\} \left(1 - \max_{\alpha\in\{\alpha_0,\dots,\alpha_s\}} \{|\alpha|\} ||D^{-1}B||\right)^{-1} \\ &\leq 2||Z||^2 \cdot c_\alpha \cdot \max_r \left\{|d_r|^{-1}\right\}, \end{aligned} \tag{4.4}$$

where c_α is a parameter of the order of unity such that $\lim_{|\alpha|\to 0} c_\alpha = 1$. Therefore, we can write

$$||F||_2 = |\alpha| ||P_{\alpha,E}^{-1} F_1||_2 \leq 2|\alpha|\delta ||Z||_2^2 c_\alpha \max_r \left\{|d_r|^{-1}\right\}.$$

Hence, denoting by $\lambda_{\min}(A)$ the smallest eigenvalue of A, by

$$|d_r|^{-1} \leq (\lambda_{\min}(A)||z_r||_2^2)^{-1},$$

(see [2]), we have

$$||F||_2 \leq \frac{2|\alpha| c_\alpha \delta}{\lambda_{\min}(A)} \left(\frac{||Z||_2}{\min_r ||z_r||_2}\right)^2,$$

where $Z = [z_1 \ \cdots \ z_n]$, $z_r \in \mathbb{C}^n$. □

Note that the norm of F can be bounded by a constant which does not depend on α. Moreover, if the condition (4.1) is not satisfied because $|\alpha| \cdot ||E||$ is large, thus we can use $P_{\alpha,E} = \alpha E$, as suggested in Section 2.

Similarly to what observed in [5], the results of Theorem 4.1, without further assumptions, have a limited role in practice in order to explain the convergence of preconditioned iterations. For example, the norm of $\tilde{Z}$ and of $\tilde{L}^{-1}$ ($Z = L^{-H}$) can be large and therefore the spectrum of the preconditioned matrix cannot be considered clustered at all. Note that $||\tilde{Z}||$, $||Z||$ (and therefore $||L^{-1}||$, $||\tilde{L}^{-1}||$) can be large if, e.g., the entries of A^{-1} do not decay or decay very slowly away from the main diagonal. To this end, we give the (straightforward) extensions of [5, Theorem 4.2] and of [5, Corollary 4.3] for (1.1).

THEOREM 4.2. *Let A be Hermitian, positive (or negative) definite and normalized, $A^{-1} = ZD^{-1}Z^H$, $D = \mathrm{diag}(d_1, ..., d_n)$, $Z = (z_{i,j})$. Then, for $j > i$, we have:*

$$|z_{i,j}| \le \sqrt{|d_j|}\,\frac{1-t^n}{1-t}\max\left\{|\lambda_{\min}^{-1}(A)|, \frac{(1+\sqrt{|\lambda_{\max}(A)|/|\lambda_{\min}(A)|})^2}{2|\lambda_{\max}(A)|}\right\} t^{j-i}, \tag{4.5}$$

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

When the bandwidth and the condition number of A are not too large, the entries of Z (i.e., of L^{-H}) are bounded in a rapidly decaying fashion away from the main diagonal along rows. In this case, we can find a constant such that the inverse of L (and thus $P^{(k)}_{\alpha,E}$)) has uniformly bounded norm. Thus, by Theorems 4.1 and 4.2, we have the following consequence.

COROLLARY 4.3. *Let A be a normalized symmetric positive definite diagonally dominant matrix, and let αE, $\alpha \in \mathbb{C}^+ = \{z \in \mathbb{C} : \mathrm{Re}(z) \ge 0\}$, be a diagonal matrix whose entries have positive real part. Then, $P^{-1}_{\alpha,E}A_{\alpha,E}$ ($Q_{\alpha,E}A_{\alpha,E}$) has a clustered spectrum.*

We recall that if $\kappa_2(V_{\alpha,E})$ is moderate, the eigenvalue distribution is relevant for the convergence of GMRES, see [18] and, e.g., [9], and this is the case for all test problems considered here. In practice, in our tests, $\kappa_2(V_{\alpha,E})$ is always less than 100 and diminishes for increasing σ_1 for Problems 1 and 2 in section 5.

The hypotheses in Theorem 4.1 and Corollary 4.3 are restrictive for several classes of problems. However, we experienced that preconditioned iterations can converge fast even when A is not diagonally dominant and there is no decay of the entries of $Z = L^{-H}$ far away from the main diagonal at all.

5. Numerical experiments.

We implemented a Matlab version of our algorithms to solve the linear systems arising in two classes of problems: the diffusion equation in a rectangle and the Helmholtz equation. Here the updated preconditioners (3.8) and (3.10) have order $k = -1, 0, 1, 2$. The iterations stop when the relative residual is less than 10^{-6}. The estimated computational costs are reported in the columns "Mf" including the number of floating point operations for both the computation of the preconditioner and the iteration phase (for the proposed updated incomplete factorizations the former is negligible).

5.1. Using updated incomplete factorizations.

Let us consider the application of the proposed algorithms based on (3.8), (3.10) at each iteration step of a Krylov subspace method. In particular, we show results using (full) GMRES.

Updated factorization (3.8) used as a preconditioner for (1.1) requires the solution of the sparse linear systems whose matrices are as follows:

- $\tilde{L}$ (sparse lower triangular);
- $\tilde{D} + \alpha B$ (with band k);
- $\tilde{L}^H$ (sparse upper triangular).

Therefore, the computational cost per iteration is of the order of $O(n)$ provided that solving

$$(D + \alpha B)\, x = y \tag{5.1}$$

requires $O(n)$ flops as well. In particular, the order k updated factorization requires $O(k^2\, n)$ for the solution of the banded linear system (5.1) and $O(dn)$ flops for forward and backward substitution per iteration, where d is the number of nonzero diagonals of $\tilde{L}$. On the other hand, the updated incomplete inverse factorization (3.10) used as a preconditioner for (1.1) does not require the solution of the two (sparse) triangular linear systems because (approximations for) their inverses are already available. This can be much more effective than the iterations using (3.8) in a parallel implementation. However, we observed that for some problems, more flops are required for the additional fill-in of $\tilde{Z}$ with respect to $\tilde{L}$ for the latter algorithm.

Note that if A in (1.1) is positive (or negative) definite with real entries, the updated preconditioners (3.8), (3.10) can be particularly convenient with respect to the "full" ones (i.e., those which compute an approximation for A_j in (1.1) by applying the $ILDL^H$ or AINV algorithms for each j to A_j from scratch). Indeed, the application of the latter algorithms requires

- generating s incomplete factorizations which have to be done in complex arithmetic even if A, the *seed* matrix, has real coefficients;
- performing matrix-vector multiplications (with $\tilde{Z}$ and $\tilde{Z}^H$) or solving triangular systems (i.e., those with matrices $\tilde{L}$ and $\tilde{L}^H$) in complex arithmetic.

On the other hand, the strategies we propose here require the generation of just one incomplete factorization performed in real arithmetic. Moreover, the most expensive part of the matrix-vector multiplications for the underlying Krylov accelerator can be performed in real arithmetic as well. Complex arithmetic is used just for solving diagonal linear systems and performing operations with vectors.

5.2. Time-dependent PDEs. Consider a linear (or linearized) boundary value problem for a time-dependent PDE discretized in space written as

$$\left\{\begin{array}{ll} y'(t) & = f(t, y(t)) := J\, y(t) + g(t), \quad t \in (t_0, T] \\ y(t_0) & = \eta_1, \quad y(T) = \eta_2 \end{array}\right. \tag{5.2}$$

where $y(t), g(t) : \mathbb{R} \to \mathbb{R}^m$, $J \in \mathbb{R}^{m\times m}$, $\eta_j \in \mathbb{R}^m$, $j = 1, 2$. By applying linear multistep formulas in boundary value form as in [6, 8], we obtain the linear system

$$\begin{gathered} MY = b,\ Y = \left(y_0^T, y_1^T, ..., y_s^T\right)^T,\ M = A \otimes I_m - hB \otimes J, \\ b = e_1 \otimes \eta_1 + e_{s+1} \otimes \eta_2 + h(B \otimes I_m)g,\ g = (g(t_0) \cdots g(t_s))^T \end{gathered} \tag{5.3}$$

where $e_i \in \mathbb{R}^{s+1}$, $i = 1, ..., s+1$, is the i–th column of the identity matrix and A, $B \in \mathbb{R}^{(s+1)\times(s+1)}$ are small rank perturbations of Toeplitz matrices. In particular, the preconditioners introduced in [6, 7] for the linear systems are given by

$$P = c(A) \otimes I - hc(B) \otimes J, \tag{5.4}$$

where $c(A)$, $c(B)$ are $\{\omega\}$-circulant approximations of the matrices A, B, which are related to the discretization in time and $\tilde{J}$ is an approximation of the Jacobian matrix. Other possible choices for $c(A)$, $c(B)$ are proposed in [6, 7]. As observed in [6, 8], P can be written as

$$P = (U^* \otimes I)G(U \otimes I), \tag{5.5}$$

U is an unitary matrix given by $U = F\Omega$, F is the Fourier matrix

$$F = (F_{j,r}),\ F_{j,r} = \exp(2\pi\mathbf{i}jr/(s+1)),\ \Omega = \mathrm{diag}(1, \omega^{-1/(s+1)}, ..., \omega^{-s/(s+1)}),$$

$$\omega = \exp(\mathbf{i}\theta), \; -\pi < \theta \leq \pi,$$

ω is the parameter which determines the $\{\omega\}$-circulant approximations (for the best choices of ω, see [8]), and G is a block diagonal matrix given by

$$G = \mathrm{diag}(G_0, ..., G_s), \quad G_j = \phi_j I - h\psi_j \tilde{J}, \; j = 0, ..., s. \tag{5.6}$$

The parameter h is related to the step size of the discretization in time, $\phi_0, ..., \phi_s$ and $\psi_0, ..., \psi_s$ are the (complex) eigenvalues of $c(A)$, $c(B)$ in (5.4), respectively, see [8] for more details.

We stress that in [6, 7, 8] we considered $\tilde{J} = J$ in practice. Now, we can use the tools developed in the previous sections to solve the $s+1$ linear systems with matrices G_j as in (5.6) by a Krylov accelerator. In particular, we apply our updated preconditioners (3.8), (3.10). Therefore, we need to compute just an incomplete factorization which approximates J or an approximate inverse factorization which approximates J^{-1}. We update the underlying factorization(s) to solve the $s+1$ $m \times m$ complex symmetric linear systems with matrices G_0,...,G_s. Indeed, the matrices G_j can be written as the matrix J shifted by a complex parameter times the identity matrix:

$$G_j = h\psi_j \left(\left(-\tilde{J} \right) + \frac{\phi_j}{h\psi_j} I \right), \quad j = 0, ..., s,$$

where the ratios $\phi_j/(h\psi_j)$, $j = 0, ..., s$ have nonnegative real part; see [6].

Therefore, we can solve the block linear system $Gy = c$ by solving the component linear systems

$$G_j y_j = c_j, \quad j = 0, ..., s,$$

by a Krylov subspace solver suitable for complex symmetric linear systems. The matrix-vector products with the unitary matrices U and U^* are performed by using FFTs.

To test the performance of our updated incomplete factorizations in the above mentioned strategy, we consider the diffusion equation in a rectangular domain with variable diffusion coefficient

$$\begin{cases} \dfrac{\partial u}{\partial t} = \nabla \cdot (c \nabla u), & (x, y) \in \mathbf{R} = [0, 3] \times [0, 3], \\ u((x, y), t) = 0, & (x, y) \in \partial\mathbf{R}, \quad t \in [0, 6], \\ u((x, y), 0) = x\, y, & \;\; (x, y) \in \mathbf{R}, \end{cases}$$

where $c(x, y) = \exp(-x^\beta - y^\beta)$, $\beta \geq 0$. Using centered differences to approximate the spatial derivatives with stepsize $\delta x = 3/(m + 1)$, we obtain a system of m^2 ordinary differential equations whose $m^2 \times m^2$ Jacobian matrix J_m is block tridiagonal. The underlying initial value problem is thus solved by the linear multistep formulas in boundary value form used in [8]. In Table 5.1 we report the outer preconditioned GMRES iterations and the average inner iterations (i.e., for the solution of the $m \times m$ linear systems as in (5.6)) with the related global cost using (full) GMRES; GMRES preconditioned by the standard "complete"approximate inverse factorization for each G_j; GMRES preconditioned by (3.10) (order 0) and GMRES preconditioned by the same approximate inverse factorization computed for J reused for all $j = 0, ..., s$ (i.e., order "-1"). Both the outer and the inner iterations are considered converged if the initial residuals are reduced by 10^{-6}.

			Not prec		full $AINV$		$AINV_0$		$AINV(J)$	
m	s	out	avg	Mf	avg	Mf	avg	Mf	avg	Mf
8	8	9	21.6	44	10.5	58	14.8	**18**	30.8	53
	16	8	17.3	58	10.3	103	15.7	**35**	35.9	122
	24	8	15.2	74	10.2	155	16.0	**55**	38.6	206
	32	8	13.9	84	10.2	204	16.2	**72**	40.2	291
16	8	9	51.8	789	15.7	2845	19.3	**105**	36.9	288
	16	9	40.2	1054	14.4	5663	19.2	**210**	45.0	815
	24	9	35.3	1139	14.1	7633	19.6	**292**	50.8	1334
	32	9	31.6	1276	13.7	10153	19.6	**385**	54.5	1997
24	8	9	84.3	4328	21.7	31282	24.1	**330**	40.8	754
	16	9	65.9	5783	19.4	62453	23.3	**635**	47.8	2001
	24	9	56.3	6698	19.1	93845	22.9	**942**	53.1	3629
	32	8	51.2	6753	18.7	124899	23.1	**1114**	57.5	4896
32	8	10	117	15874	*	*	28.8	**868**	43.9	1685
	16	9	92.7	19445	*	*	28.2	**1512**	51.2	3933
	24	9	79.3	22500	*	*	27.2	**2187**	55.7	6909
	32	9	70.6	24709	*	*	26.6	**2785**	59.0	10257

TABLE 5.1

GMRES (average inner and **out***er) iterations for the diffusion equation in a rectangle* $[0,3]\times[0,3]$*,* $t\in[0,6]$*,* $c=e^{-x-y}$ *(LMF in bv form). The "*"means that more than 0.5 (Matlab counter) Teraflops are required. Note that the preconditioner based on (3.6) (order 0, i.e.,* $B=I$*) gives the best performance.*

Results in Table 5.1 confirm that GMRES preconditioned with (3.10) is a little bit slower in term of average inner iterations with respect to the standard preconditioning strategy based on "full" approximate inverse factorization. However, the global computational cost of our strategy is greatly reduced with respect to the others. For this example, we found that the updated inverse factorizations (3.10) of order greater than 0 do not improve the overall performance.

5.3. Helmholtz equation. An example of a problem whose discretization produces complex symmetric linear systems is the Helmholtz equation with complex wave numbers

$$-\nabla\cdot(c\nabla u)+\sigma_1(j)u+\mathbf{i}\sigma_2(j)u=f_j,\quad j=0,\dots,s, \tag{5.7}$$

where $\sigma_1(j)$, $\sigma_2(j)$ are real coefficient functions and c is the diffusion coefficient. The above equation describes the propagation of damped time-harmonic waves. We consider (5.7) on the domain $\mathcal{D}=[0,1]\times[0,1]$ with σ_1 constant, σ_2 a real coefficient function and $c(x,y)=1$ or $c(x,y)=e^{-x-y}$. As in [14], we consider two cases.

- [Problem 1] Complex Helmholtz equation, u satisfies Dirichlet boundary conditions in $\mathcal{D}$. We discretize the problem with finite differences on a $n\times n$ grid, $N=n^2$, and mesh size $h=1/(n+1)$. We obtain the $s+1$ linear systems $(j=0,\dots,s)$:

  $$A_jx_j=b_j,\ A_j=H+h^2\sigma_1(j)I+\mathbf{i}h^2D_j,\ D_j=\mathrm{diag}(d_1,\dots,d_N), \tag{5.8}$$

  where H is the discretization of $-\nabla\cdot(c\nabla u)$ by means of centered differences. The $d_r=d_r(j)$, $r=1,\dots,N$, $j=0,\dots,s$, are the values of $\sigma_2(j)$ at the grid points.

- [Problem 2] Real Helmholtz equation with complex boundary condition

$$\frac{\partial u}{\partial n} = \mathbf{i}\sigma_2(j)u, \ \{(1,y) \mid 0 < y < 1\}$$

  discretized with forward differences and Dirichlet boundary conditions on the remaining three sides gives again (5.8). The diagonal entries of D_j are given by $d_r = d_r(j) = 1000/h$ if r/m is an integer, 0 otherwise.

All test problems are based on a 31×31 mesh, the right hand sides are vectors with random components in $[-1,1] + \mathbf{i}[-1,1]$ and the initial guess is a random vector. Notice that the methods based on (1.5) cannot be used here because the right hand side and the initial guess x_j change for each $j = 0, ..., s$. Moreover, the seed matrix is not just shifted because the E_j are diagonal and not just the identity matrices (they have complex, non-constant diagonal entries) and thus the Krylov subspaces (1.5) do not coincide.

We consider the solution of the related 2D model problem in the unit square by using GMRES without preconditioning, with the approximate inverse preconditioner described in [2], the order 0, 1 and 2 updated approximate inverse preconditioner (3.10) and the incomplete LDL^H-threshold preconditioner (3.8). The drop tolerance for the incomplete factorizations is set to preserve the sparsity of the originating matrices. The results for the Problems 1 and 2 are shown in tables 5.2, 5.3 and 5.4, 5.5, respectively, confirming that our approach is effective. In particular, even if the global number of iterations using the update preconditioners can increase with respect to “full preconditioned” iterations, the flop count is lower with respect to the other methods. However, if σ_1 is greater than a suitable value, our update preconditioners should be used with updated triangular factors $\tilde{L}$ and $\tilde{Z}$ as well otherwise our preconditioners can be not efficient; see table 5.2 for $\sigma_1 = 800$. However, we recall that usually in these cases preconditioning is not necessary at all; see section 2. Strategies for updating $\tilde{Z}$ were described for positive definite matrices in [5, section 6].

For the particular seed matrix A considered in these examples, we can observe that incomplete LDL^H-based preconditioners perform slightly better. This is caused by the particular properties of the Laplacian discretized by the usual five-point formula. Indeed, we observed that for other problems updated approximate inverse factorization preconditioners (3.10) perform better. In particular, this happens for positive definite seed matrices A whose incomplete factorization generated by incomplete Cholesky factorization is ill conditioned. On the other hand, we stress that the (stabilized) approximate inverse preconditioner described in [2] is well defined for all positive definite matrices. The same is true for incomplete LDL^H factorization generated by the algorithm proposed in [3]. Therefore, the updated approximate inverse (3.10) and incomplete factorization (3.8) based on those are well defined as well.

We stress that updates of order k greater than one are sometimes not efficient for the considered problems. On the other hand, recall that the matrices related to problems 1 and 2 can be written as A_j in (5.8). Therefore, if σ_1 is positive and not negligible with respect to the entries of H, say, the inverse of (5.8) has entries which exhibit a fast decay away from the main diagonal; see, e.g., figure 3.1. Therefore, it seems reasonable that diagonal corrections (i.e., updates of order 0 and 1) give good approximations with the minimum computational cost. Moreover, order $k > 1$ approximations require the solution of a k–banded linear system in complex arithmetic per iteration, which can represent a not negligible computational cost if we

	Not prec		$ILDL_0^H$		$ILDL_1^H$		$ILDL_2^H$		full $ILDL^H$	
σ_1	it	Mf	it	Mf	it	Mf	it	Mf	it	Mf
50	38	13.9	22	7.1	22	7.1	18	**7.0**	19	9.0
100	36	12.7	20	**6.2**	20	**6.2**	17	6.5	17	7.7
200	32	10.2	18	**5.3**	18	**5.3**	15	**5.3**	15	6.6
400	26	7.2	16	**4.5**	16	**4.5**	13	**4.5**	12	5.1
800	20	4.6	15	4.1	15	4.1	12	4.1	9	**3.7**

TABLE 5.2

Order k, $k = 0, 1, 2$ modified and incomplete LDL^H (i.e., recomputed at each step) preconditioners. Results for the complex Helmholtz equation and Dirichlet boundary conditions as in Problem 1. The diagonal entries of D are chosen randomly in $[0, 1000]$.

	$AINV_0$		$AINV_1$		AINV2		full $AINV$	
σ_1	it	Mf	it	Mf	it	Mf	it	Mf
50	26	8.8	26	8.8	20	**7.8**	15	1793
100	25	8.2	25	8.3	19	**7.3**	14	1793
200	22	6.7	22	6.8	17	**6.3**	13	1793
400	19	**5.4**	19	5.5	15	**5.4**	11	1792
800	17	4.6	17	4.7	13	**4.5**	8	1791

TABLE 5.3

Order k, $k = 0, 1, 2$ inverse modified and AINV (i.e., recomputed at each step) preconditioners. Results for the complex Helmholtz equation and Dirichlet boundary conditions as in Problem 1. The diagonal entries of D are chosen randomly in $[0, 1000]$.

have convergence in almost the same number of iterations of the underlying diagonal corrections. This is the case of problem 2; see tables 5.4 and 5.5.

In tables 5.2–5.5 different strategies seems to give the same number of Mflops and/or iterations. This effect is caused by different reasons.

- Tables 5.2–5.5. Order 0 and order 1 updates have a computational cost which is almost the same, but the latter approach requires obviously slightly more operations than the former. Therefore, rounding the (estimated) flops gives sometimes the same numbers.
- Table 5.2, $\sigma_1 = 100, 200, 400$ and table 5.3, $\sigma_1 = 400$. In this case, for order 0 and 1 we have the same number of Mflops and iterations for some σ_1 and, in three runs, the same (rounded) number of Mflops but a different number of iterations for order 2 (necessarily lower because more expensive than order 0 and 1).

Finally, from the above tables, we could argue that the computational cost for the application of the standard approximate inverse preconditioner (the columns "full AINV" in the tables) is higher with respect to the standard incomplete factorization (3.2) based on the Incomplete Cholesky algorithm provided by Matlab (the columns "full $ILDL^H$" in the tables). However, this is partly a consequence of the use of our rough Matlab implementation for the approximate inverse factorizations, while the incomplete LDL^H preconditioner uses a built-in function.

6. Conclusions.

We proposed a general framework for the solution of sequences of complex symmetric linear systems of the form (1.1) which is based on our algorithms for updating preconditioners generated from a seed matrix. The seed preconditioner can be based on incomplete LDL^H factorizations for A or for A^{-1} and the sequence

	Not prec		$ILDL_0^H$		$ILDL_1^H$		$ILDL_2^H$		full $ILDL^H$	
σ_1	it	Mf	it	Mf	it	Mf	it	Mf	it	Mf
.5	146	175	34	**14.0**	34	**14.0**	34	17.2	29	17.0
1	145	173	33	**13.0**	33	**13.0**	33	16.5	28	15.7
2	143	168	33	**13.0**	33	**13.0**	33	16.5	28	15.7
4	137	155	31	**12.1**	31	**12.1**	31	15.0	27	14.8
8	127	134	28	**10.3**	28	**10.3**	29	13.6	24	12.5

TABLE 5.4

Order k, $k = 0, 1, 2$ modified and incomplete LDL^H (i.e., recomputed at each step) preconditioners. Results for the real Helmholtz equation and complex boundary conditions as in Problem 2.

	$AINV_0$		$AINV_1$		$AINV_2$		full $AINV$	
σ_1	it	Mf	it	Mf	it	Mf	it	Mf
.5	47	**23.4**	47	23.5	47	27.8	46	1812
1	46	**22.6**	46	**22.6**	46	26.9	45	1811
2	46	22.6	45	**21.8**	45	26.0	45	1811
4	44	**20.0**	44	20.1	44	25.1	42	1809
8	41	18.5	40	**17.9**	40	21.6	40	1807

TABLE 5.5

Order k, $k = 0, 1, 2$ inverse modified and AINV (i.e., recomputed at each step) preconditioners. Results for the real Helmholtz equation and complex boundary conditions as in Problem 2.

of the iterations is well defined provided that the seed matrix A is definite and its preconditioner is well defined. Sufficient conditions for the fast convergence of the underlying iterations are proposed.

The updated preconditioners (3.8) and (3.10) can be improved by applying several techniques. In particular, the strategies proposed in [5, Section 6] can be generalized to the underlying algorithms.

Numerical experiments with Helmholtz equations and boundary value methods for a diffusion problem show that the proposed framework can give reasonably fast convergence. The seed preconditioner is computed just once and therefore the solution of several linear systems of the type (1.1) can be globally much cheaper than recomputing a new preconditioner from scratch and with respect to nonpreconditioned iterations, especially if the matrix A_j is ill-conditioned (if A_j is Hermitian) or has non clustered and/or close to the origin eigenvalues (if A_j is non-Hermitian and its basis of eigenvectors is not ill-conditioned). On the other hand, if the problems are not ill-conditioned (have clustered spectra), using the seed preconditioner and the updated preconditioners can give similar performance.

Acknowledgments. The author would like to thank the editor and two anonymous referees for helpful comments which have improved this presentation.

This work is dedicated to Federico who was born during the preparation of the bulk of this paper.

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 and M. Tůma, *A robust incomplete factorization preconditioner for positive definite matrices*, Numer. Linear Algebra Appl., 10 (2003), pp. 385–400.

[4] ——— *Orderings for factorized sparse approximate inverse preconditioners*, SIAM J. Sci. Comput., 21 (2000), pp. 1851–1868.

[5] M. Benzi and D. Bertaccini, *Approximate inverse preconditioning for shifted linear systems*, BIT, 43 (2003), pp. 231–244.

[6] D. Bertaccini, *A circulant preconditioner for the systems of LMF-based ODE codes*, SIAM J. Sci. Comput., 22 (2000), pp. 767–786.

[7] ——— *Reliable preconditioned iterative linear solvers for some numerical integrators*, Numer. Linear Algebra Appl., 8 (2001), pp. 111–125.

[8] ——— *Block $\{\omega\}$-circulant preconditioners for the systems of differential equations*, Calcolo, 40-2 (2003), pp. 71-90.

[9] D. Bertaccini and M. K. Ng, *Band-Toeplitz Preconditioned GMRES Iterations for time-dependent PDEs*, BIT, 43 (2003), pp. 901-914.

[10] S. Demko, W. F. Moss, and P. W. Smith, *Decay rates for inverses of band matrices*, Math. Comp., 43 (1984), pp. 491–499.

[11] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, Philadelphia, PA, 1996.

[12] 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.

[13] R. Freund, *On Conjugate Gradient Type methods and Polynomial Preconditioners for a Class of Complex Non-Hermitian Matrices*, Numer. Math., 57 (1990), pp. 285–312.

[14] ———*Conjugate gradient-type methods for, linear systems with complex symmetric matrices*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 425–448.

[15] ———*Solution of shifted linear systems by quasi-minimal residual iterations*, in Numerical Linear Algebra, L. Reichel, A. Ruttan and R. S. Varga, eds., de Gruyter, Berlin, 1993.

[16] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems*, Springer-Verlag, Berlin–Heidelberg, 1991.

[17] G. Meurant, *On the incomplete Cholesky decomposition of a class of perturbed matrices*, SIAM J. Sci. Comput., 23 (2001), pp. 419–429.

[18] Y. Saad, *Iterative methods for sparse linear systems*, PWS Publishing Company, 1996.

[19] V. Simoncini and E. Gallopoulos, *An iterative method for nonsymmetric systems with multiple right hand sides*, SIAM J. Sci. Comput., 16 (1995), pp. 917-933.