A ROBUST ITERATIVE SCHEME FOR SYMMETRIC INDEFINITE SYSTEMS*

MURAT MANGUOĞLU† AND VOLKER MEHRMANN‡

Abstract. We propose a two-level nested preconditioned iterative scheme for solving sparse linear systems of equations in which the coefficient matrix is symmetric and indefinite with relatively small number of negative eigenvalues. The proposed scheme consists of an outer Minimum Residual (MINRES) iteration, preconditioned by an inner Conjugate Gradient (CG) iteration in which CG can be further preconditioned. The robustness of the proposed scheme is illustrated by solving indefinite linear systems that arise in the solution of quadratic eigenvalue problems in the context of model reduction methods for finite element models of disk brakes as well as on other problems that arise in a variety of applications.

Key word. symmetric indefinite systems, Krylov subspace method, sparse linear systems, deflation, preconditioned minimum residual method, preconditioned conjugate gradient method
MSC 2000: 65F10, 65F15, 65F50

1. Introduction.

Symmetric indefinite linear systems

$$Ax = b, \tag{1}$$

arise in many applications ranging from optimization problems to problems in computational physics, see e.g. [2, 17]. In this paper we assume that $A \in \mathbb{R}^{n\times n}$ is a sparse, full-rank, symmetric and indefinite matrix with only few negative eigenvalues. Our motivation to develop a new preconditioned iterative method arises from an application in the automotive industry. In order to control brake squeal, large scale eigenvalue problems are solved via a shift-and-invert Arnoldi method to obtain a reduced model that can be used for parameter studies and optimization, see [10] and Section 3.1. We propose the use of a two-level preconditioned iterative method with a positive definite preconditioner for the solution of the arising linear systems. The basic idea of such a preconditioner iteration is well-known. In the context of optimization problems, see [9], a sparse Bunch-Parlett factorization

$$PAP^T = LDL^T \tag{2}$$

is suggested as a solver for the systems involving the indefinite blocks of various preconditioners. Where P is a permutation matrix (with $PP^T = I$), L is a sparse lower triangular matrix (typically with some fill-in compared to the sparsity pattern of A), and D is a block-diagonal matrix that contains either 1×1 or 2×2 blocks. Given such a factorization, one can modify the diagonal matrix D to obtain a positive definite $\tilde{D}$ such that the eigenvalues of $\tilde{D}$ are the absolute values of the eigenvalues of D, so that also $M := L\tilde{D}L^T$ is positive definite. If a diagonal block of D is 1×1 and

*Submitted DATE.

Funding: The first author would like to thank Alexander von Humboldt Foundation for support of a research stay at TU Berlin. The second author thanks Einstein Foundation Berlin for support within the project: *Algorithmic Linear Algebra: High Performance, Numerical Stability, and Error Resiliency.*

†Institut für Mathematik, Technische Universität, Berlin, 10623 Berlin, Germany, and Department of Computer Engineering, Middle East Technical University, 06800 Ankara, Turkey (manguoglu@ceng.metu.edu.tr)

‡Institut für Mathematik, Technische Universität Berlin, 10623 Berlin, Germany (mehrmann@math.tu-berlin.de)

negative, then one replaces it with its absolute value. Otherwise, it is a symmetric 2×2 block,

$$\begin{bmatrix} \alpha & \beta \\ \beta & \gamma \end{bmatrix}. \tag{3}$$

and one computes the spectral decomposition

$$\begin{bmatrix} \alpha & \beta \\ \beta & \gamma \end{bmatrix} = \begin{bmatrix} c & s \\ s & -c \end{bmatrix} \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix} \begin{bmatrix} c & s \\ s & -c \end{bmatrix} \tag{4}$$

where $c, s \in \mathbb{R}$ satisfy $c^2 + s^2 = 1$, and one replaces the 2×2 block with

$$\begin{bmatrix} \tilde{\alpha} & \tilde{\beta} \\ \tilde{\beta} & \tilde{\gamma} \end{bmatrix} = \begin{bmatrix} c & s \\ s & -c \end{bmatrix} \begin{bmatrix} |\lambda_1| & 0 \\ 0 & |\lambda_2| \end{bmatrix} \begin{bmatrix} c & s \\ s & -c \end{bmatrix}. \tag{5}$$

The matrix M, if easily available, is a good preconditioner for a preconditioned Krylov subspace method, such as the Minimum Residual method (MINRES) [14], since due to the fact that the spectrum of $M^{-1}A$ has only the values $+1, -1$, it would converge in at most 2 iterations in exact arithmetic if the factorization is exact. However, this preconditioner is, in general, not practical for large problems due to fill-in and large storage requirements. In [12], therefore, an incomplete LDL^T factorization ($ILDL^T$) based preconditioner for MINRES is proposed.

Another suggestion for a preconditioner of MINRES, proposed in [21], is the positive definite absolute value of A, defined as $|A| := V|\Lambda|V^T$ in which $A = V\Lambda V^T$ is the spectral decomposition of A, however, to avoid the high computational complexity of the spectral decomposition, in [21] it is suggested to use a geometric multigrid method instead of the absolute value preconditioner and it is illustrated via a model problem that this approach is very effective when the system matrix arises from elliptic partial differential equations.

In our motivating problem, the indefinite matrix arises from a perturbed wave equation where the resulting linear system depends on parameters and has the extra property that the number of negative eigenvalues is much smaller than the number of positive eigenvalues. For this class of problems we propose a new two-level iterative scheme that combines the absolute value preconditioner approach with a deflation procedure and we show that this method is also very effective for a large class of indefinite problems arising in other applications.

2. A two-level iterative scheme. In this section we describe a new two-level preconditioned iterative scheme for symmetric indefinite linear systems where the coefficient matrix has only very few negative eigenvalues. The method employs MINRES together with a modified absolute value preconditioner that is constructed via a deflation procedure which, however, is not carried out explicitly. The linear systems involving the preconditioner are solved again iteratively via the preconditioned Conjugate Gradient (CG) [7] which can be preconditioned via an incomplete LU (ILU) decomposition, see e.g. [17], of the original coefficient matrix A or any other preconditioner obtained from the original coefficient matrix . These include but are not limited to Sparse Approximate Inverse Algebraic Multigrid based preconditioner as well. We illustrate that this MINRES-CG iterative scheme is very effective and more robust than other preconditioned general Krylov subspace methods, such as the restarted Generalized Minimum Residual (GMRES) [18], the stabilized Bi-Conjugate

Gradient method (BiCGStab) [19], inner-outer FGMRES-GMRES [16] or just $ILDL^T$ preconditioned MINRES.

As an approximation to the absolute value preconditioner we use

$$M_{mr} := A + 2\hat{V}|\hat{\Lambda}|\hat{V}^T. \tag{6}$$

where $\hat{V}$ is an approximate invariant subspace of A associated with the (say k) negative eigenvalues and $|\hat{\Lambda}|$ is the corresponding absolute value of the diagonal matrix of negative eigenvalues. Since we have assumed that k is much smaller than n, the modification (or as it is sometimes called *deflation*) is of small rank. In each iteration of MINRES applied to (1) a system of the form

$$M_{mr}z = y \tag{7}$$

has to be solved, and again the preconditioned matrix $M_{mr}^{-1}A$ has only eigenvalues $+1$ or -1 so that MINRES with the exact preconditioner converges theoretically again in at most 2 iterations. However, since M_{mr} is symmetric and positive definite, we propose to use a preconditioned CG iteration for solving system (7) approximately with an indefinite preconditioner, M_{cg}, which is an approximation of the original coefficient matrix itself. Note that the eigenvalues of the preconditioned matrix for CG, $M_{cg}^{-1}M_{mr}$, would again be either $+1$ or -1 if the exact matrix A^{-1} was used.

Indefinite preconditioning for the CG method is rarely applied with the exception of [15], where CG for indefinite systems with indefinite preconditioner is used but it is assumed that the preconditioned matrix is positive definite. In our case, however, this will not be the case.

The first level preconditioner (M_{mr}) is symmetric and positive definite, but dense, so it should not be formed explicitly. On the other hand, the second level preconditioner (M_{cg}) is sparse and symmetric but not positive definite. However, the preconditioned CG is still guaranteed not to break down (see [17, p. 277]) using an indefinite preconditioner which can be seen as follows. It is well-known, see e.g. [17, p. 279], that preconditioned CG with a preconditioner M applied to a system $Wx = b$ with symmetric positive definite W can be expressed in an indefinite M-scalar product by replacing the Euclidean inner products in CG by the M-inner products. If W is symmetric positive definite, and M is symmetric indefinite (but invertible), then we can define the indefinite M-inner product as $(x, y)_M = (Mx, y) = y^T Mx = x^T My = (y, x)_M$, so $M^{-1}W$ is positive definite with respect to the M-inner product, since $(M^{-1}Wx, x)_M > 0$ for all $x \neq 0$.

Given the system $Wx = z$, an initial guess x_0, and a preconditioner M, as CG is a projection based Krylov subspace method, the vectors x_m must satisfy the orthogonality condition

$$(M^{-1}(z - Wx_m), v)_M = 0 \quad \text{for all } v \in \hat{\mathcal{K}}_m, \tag{8}$$

where $\hat{\mathcal{K}}_m = \text{span}\{\hat{r}_0, M^{-1}W\hat{r}_0, ..., (M^{-1}W)^{(m-1)}\hat{r}_0\}$ and $\hat{r}_0 = M^{-1}r_0$ with $r_0 = z - Wx_0$. Note that (8) is equivalent to the orthogonality condition of CG without preconditioning

$$(z - Wx_m, v) = 0 \quad \text{for all } v \in \hat{\mathcal{K}}_m \tag{9}$$

Therefore, indefinitely preconditioned CG minimizes the error

$$||x_m - x^*||_W = \inf_{x \in x_0 + \hat{\mathcal{K}}_m} ||x - x^*||_W, \tag{10}$$

in the energy norm defined by the positive definite matrix W.

In summary, our two-level scheme consists of two stages. First, we compute approximations to the negative eigenvalues and the corresponding invariant subspace (see Algorithm 1). This computation itself may be very expensive even if the invariant subspace has small dimension. However, in our motivating application many linear systems with the same coefficient matrix (or closely related coefficient matrices) need to be solved. Hence, this potentially expensive initial cost is quickly amortized. This is typical when solving eigenvalue problems with the shift-and-invert Arnoldi method as in [10]. The second stage in the iterative solution stage consists of nested MINRES and CG iterations (Algorithm 2). Note that while the outer MINRES iterations require matrix-vector multiplications with the original sparse coefficient matrix A, the inner CG iterations require matrix-vector multiplications of the form $v = M_{mr}u$ which are efficiently performed by using sparse matrix-vector multiplications and together with dense matrix-vector operations (BLAS Level 2) and vector-vector operations (BLAS Level 1) in the following procedure

$$M_{mr}u = Au + 2(\hat{V}(|\hat{\Lambda}|(\hat{V}^*u))) \tag{11}$$

The total cost of each such matrix multiplication operation is $O(nnz + 2kn)$ where nnz,k and n are the number of nonzeros, negative eigenvalues and rows of A, respectively. We note that this extra operation is much more cache friendly than constructing an orthonormal basis in GMRES which rely on dot products (BLAS Level 1). Alternatively, to further speed up the convergence, the proposed scheme can be implemented using RMINRES [8, 22], i.e. recycled and deflated MINRES, as the outer solver instead of plain MINRES. The trade-off would be increased storage and computation requirements due to the necessary orthogonalization against the recycled subspace and the updates of the recycled subspace [22]. Furthermore, finding subspaces that lead to improved convergence is considered to be a highly challenging task and application specific [8].

Algorithm 1 Preprocessing stage of MINRES-CG

Function `MINRES-CG-Preprocess(`A`):`

Compute (or approximate) all negative eigenvalues and the corresponding invariant subspace $\hat{V}$ of A ($A\hat{V} = \hat{\Lambda}\hat{V}$)

return $\hat{\Lambda}, \hat{V}$

2.1. Improvement via Sherman-Morrison-Woodbury Formula. The preconditioner M_{mr} in MINRES-CG is a k-rank update of A. Therefore, one can use the Sherman-Morrison-Woodbery formula to express M_{mr}^{-1}. Given,

$$M_{mr} := A + 2\hat{V}|\hat{\Lambda}|\hat{V}^T \tag{12}$$

after applying the Sherman-Morrison-Woodbury formula and some algebraic manipulations, we obtain,

$$M_{mr}^{-1} := A^{-1} - 2\hat{V}\hat{\Lambda}^{-1}\hat{V}^T. \tag{13}$$

Note that since we do not have the exact A^{-1}, but use an approximation of it, M_{mr}^{-1} is not positive definite. Still, we can use it as the preconditioner for the CG iterations. In other words, we apply the preconditioner for CG, $M_{cg}^{-1} = \tilde{A}^{-1} - 2\hat{V}\hat{\Lambda}^{-1}\hat{V}^T$ in which

Algorithm 2 Iterative solution stage of MINRES-CG

Function `MINRES-CG(`A,b,x_0,$\hat{\Lambda}$,$\hat{V}$`):`

Solve $Ax = b$ via MINRES using the preconditioner $M_{mr} = A + 2\hat{V}|\hat{\Lambda}|\hat{V}^T$ in each iteration of MINRES using the subroutines:

- Compute matrix-vector products with A.
- Solve $M_{mr}z = y$ via preconditioned CG using as preconditioner $M_{cg} = \tilde{A}$ an approximation of A.
  In each iteration of PCG:
  - Compute matrix-vector products: $v = M_{mr}u$
  - Solve the system $M_{cg}t = g$

return x

the action of $\tilde{A}^{-1}$ is approximated, such as by an incomplete factorization of A. In the improved scheme, application of the preconditioner involves an additional dense matrix-vector multiplication (BLAS Level 2) cost of $2kn$ but no additional storage requirement. Hereafter, we refer to this improved version as MINRES-CG*.

3. Application of the two-level method. In this section we describe the applications to which we apply the proposed two-level procedure.

3.1. Finite Element Models of Disk Brakes. In the context of noise reduction in disk brakes, reduced order models are determined from the finite element model [10] by computing the eigenvalues in the right half plane and close to the imaginary axis of a parametric Quadratic Eigenvalue Problem (QEP)

$$(\lambda^2\mathcal{M} + \lambda D_\Omega + K_\Omega)x = 0 \tag{14}$$

in which

$$D_\Omega = D_M + \left(\frac{\Omega_{ref}}{\Omega} - 1\right) D_R + \left(\frac{\Omega}{\Omega_{ref}}\right) D_G \tag{15}$$

and

$$K_\Omega = K_E + K_R + \left(\left(\frac{\Omega}{\Omega_{ref}}\right)^2 - 1\right) K_g, \tag{16}$$

where $\mathcal{M}$ and K_E are symmetric positive definite, D_G is skew-symmetric, D_M, D_R, K_g are symmetric indefinite, and K_R is general [10]. Here Ω denotes the angular velocity in the disk ($2\pi < \Omega < 4 \times 2\pi$) and Ω_{ref} is the reference angular velocity.

The QEP is solved by first rewriting it as a linear eigenvalue problem, using a companion linearization of (14) given by

$$\left(\begin{bmatrix} 0 & I \\ -K_\Omega & -D_\Omega \end{bmatrix} - \lambda \begin{bmatrix} I & 0 \\ 0 & \mathcal{M} \end{bmatrix}\right) \begin{bmatrix} x \\ \lambda x \end{bmatrix} = 0. \tag{17}$$

Audible brake squeal is associated with eigenvalues in the right half plane. For this reason we are interested in those eigenvalues that lie in a rectangular domain in the complex plane given by $-50 < \mathrm{Re}(\lambda) < 1,000$ and $-1 < \mathrm{Im}(\lambda) < 20,000$ corresponding to the audible range.

Solving the eigenvalue problem (17) via an eigensolver such as the shift-and-invert Arnoldi method [13], requires the solution of a shifted linear system of equations in each iteration, see [10] for details of the eigensolver. To apply our two-level linear system solver, we consider the solution of the following shifted linear system with complex shifts (γ inside the rectangular domain of interest),

$$C(x+iy) = f + ig \tag{18}$$

where $i = \sqrt{-1}$, $C = \gamma B - A$, and

$$B = \begin{bmatrix} I & 0 \\ 0 & \mathcal{M} \end{bmatrix}, A = \begin{bmatrix} 0 & I \\ -K_\Omega & -D_\Omega \end{bmatrix}. \tag{19}$$

In [10] this complex linear system is solved with a sparse complex direct solver. To solve the problem iteratively, we follow [1] and map the complex system (18) to an equivalent double-size real system.

Splitting into real and imaginary parts $C = \hat{A} + i\hat{B}$ and $\gamma = \gamma_r + i\gamma_i$ with $\gamma_r = \mathrm{Re}(\gamma)$ and $\gamma_i = \mathrm{Im}(\gamma)$, we obtain

$$\hat{A} = \begin{bmatrix} \gamma_r I & -I \\ K_\Omega & \gamma_r \mathcal{M} + D_\Omega \end{bmatrix}, \hat{B} = \begin{bmatrix} \gamma_i I & 0 \\ 0 & \gamma_i \mathcal{M} \end{bmatrix}, \tag{20}$$

for the real and complex parts of C, respectively. This leads to the real system

$$\begin{bmatrix} \hat{B} & -\hat{A} \\ \hat{A} & \hat{B} \end{bmatrix} \begin{bmatrix} x \\ -y \end{bmatrix} = \begin{bmatrix} g \\ f \end{bmatrix}. \tag{21}$$

which we then solve via a preconditioned Krylov subspace method with preconditioner

$$M = \begin{bmatrix} \tilde{B} & -\tilde{A} \\ \tilde{A} & \tilde{B} \end{bmatrix} \tag{22}$$

where

$$\tilde{A} = \begin{bmatrix} \gamma_r I & -I \\ K_E & \gamma_r \mathcal{M} \end{bmatrix} \tag{23}$$

and $\tilde{B} = \hat{B}$. Note that both $\mathcal{M}$ and K_E are symmetric and positive definite. The preconditioner can be block LU factorized as

$$\begin{bmatrix} \tilde{B} & -\tilde{A} \\ \tilde{A} & \tilde{B} \end{bmatrix} = \begin{bmatrix} \tilde{B} & 0 \\ \tilde{A} & \tilde{B} + \tilde{A}\tilde{B}^{-1}\tilde{A} \end{bmatrix} \begin{bmatrix} I & -\tilde{B}^{-1}\tilde{A} \\ 0 & I \end{bmatrix}. \tag{24}$$

Hence, the major cost in solving systems involving the preconditioner M is the solution of two linear systems where the coefficient matrix is (i) $\tilde{B}$ and (ii) $S = (\tilde{B} + \tilde{A}\tilde{B}^{-1}\tilde{A})$, namely the Schur complement. Since the solution of (i) is quite trivial, we only discuss how to solve systems involving the Schur complement matrix, which typically is dense see [2], but in our case it has the factorization

$$\underbrace{\tilde{B} + \tilde{A}\tilde{B}^{-1}\tilde{A}}_{S} = \underbrace{\begin{bmatrix} \mathcal{M}^{-1} & 0 \\ 0 & I \end{bmatrix}}_{S_1} \underbrace{\begin{bmatrix} \left(\gamma_i + \frac{\gamma_r^2}{\gamma_i}\right)\mathcal{M} - \frac{1}{\gamma_i}K_E & -2\frac{\gamma_r}{\gamma_i}\mathcal{M} \\ 2\frac{\gamma_r}{\gamma_i}K_E & \left(\gamma_i + \frac{\gamma_r^2}{\gamma_i}\right)\mathcal{M} - \frac{1}{\gamma_i}K_E \end{bmatrix}}_{S_2}. \tag{25}$$

Solving systems involving the Schur complement matrix, therefore, requires two steps: (i) scaling the right hand side vector with S_1^{-1} and (ii) solving systems where the coefficient matrix is S_2. Step (i) is again trivial, hence we now look into (ii) which we solve iteratively using a Krylov subspace method where the preconditioner is

$$\tilde{S}_2 = \begin{bmatrix} \left(\gamma_i + \frac{\gamma_r^2}{\gamma_i}\right)\mathcal{M} - \frac{1}{\gamma_i}K_E & 0 \\ 2\frac{\gamma_r}{\gamma_i}K_E & \left(\gamma_i + \frac{\gamma_r^2}{\gamma_i}\right)\mathcal{M} - \frac{1}{\gamma_i}K_E \end{bmatrix}, \tag{26}$$

since in our case $||\mathcal{M}||_F \ll ||K_E||_F$. Hence, the main cost in solving the block triangular systems lies in the solution of

$$\left[\left(\gamma_i + \frac{\gamma_r^2}{\gamma_i}\right)\mathcal{M} - \frac{1}{\gamma_i}K_E\right] u = v, \tag{27}$$

or after multiplying both sides of the system by $-\gamma_i$ we obtain

$$[K_E - |\gamma|^2\mathcal{M}]u = -\gamma_i v, \tag{28}$$

where $|\gamma|^2 = \gamma_i^2 + \gamma_r^2$. Even though $\mathcal{M}$ and K_E are symmetric and positive definite, there is no guarantee that the symmetric coefficient matrix $K_E - |\gamma|^2 M$ is positive definite. However, system (28) is a perfectly suitable for the proposed MINRES-CG scheme, since in our application it only has few negative eigenvalues and they need to be computed only once. Furthermore, the preconditioner (22) is completely independent of the parameters Ω and Ω_{ref}, and the coefficient matrix of inner systems that have to be solved (28) are the same for a given $|\gamma|$. This means that a factorization (incomplete or exact) or an approximation for the coefficient matrix $K_E - |\gamma|^2\mathcal{M}$ can be computed once and re-used for all values of γ of the same absolute value and for all corresponding Ω values.

Numerical experiments for this class of problems are presented in section 4.

3.2. Other applications. As further applications we consider all symmetric indefinite problems in SuiteSparse Matrix Collection [5] of sizes between $n = 1000$ and $n = 50,000$ and with at most 100 negative eigenvalues. Since this includes 7 matrices from the PARSEC group [3], we exclude the 3 smallest matrices from this group. Furthermore, since shifts around the so-called Fermi level are also of interest in the PARSEC group of matrices, we shift the largest matrix (SiO) by $A - \sigma I$. For σ, we chose three values (0.25, 0.5 and 0.75) which approximately correspond to the gaps in the spectrum of A. The properties of these 11 matrices are given in Table 1. Note that we include two examples (*) that arise in finite element discretization of structural problems. These are not full eigenvalue problems but just mass matrices; solving these linear systems is useful if eigenvalues in inverse mass matrix inner product space are computed.

4. Numerical results. In this section, we study the robustness of the proposed two-level scheme for indefinite linear systems described in the previous section. All experiments are performed using MATLAB R2018a.

In MINRES-CG, we use an indefinite preconditioner (M_{cg}) obtained either by an incomplete LDL^T or LU factorization of the coefficient matrix for inner CG iterations. Former is the only suitable preconditioner available in MATLAB which we refer to as ILU. Hence, even though it does not exploit symmetry, we use it to show the robustness of the proposed scheme in Section 4.2.1. For the latter, on the other hand,

Table 1: Matrices from the SuiteSparse Matrix Collection with application domains and properties (n is matrix dimension, nnz is number of nonzeros and k is number of negative eigenvalues).

Matrix	n	nnz	k	Application
Bcsstm10*	1,086	22,092	54	Structural Engineering
Bcsstm27*	1,224	56,126	31	Structural Engineering
Nasa1824	1,824	39,208	20	Structural Engineering
Meg4	5,860	25,258	54	RAM Simulation
Benzene	8,219	242,669	2	Real-space pseudopotential method
Si10H16	17,077	875,923	41	Real-space pseudopotential method
Si5H12	19,898	738,598	6	Real-space pseudopotential method
SiO	33,401	1,317,655	8	Real-space pseudopotential method
SiO($\sigma = 0.25$)	33,401	1,317,655	16	Real-space pseudopotential method
SiO($\sigma = 0.5$)	33,401	1,317,655	26	Real-space pseudopotential method
SiO($\sigma = 0.75$)	33,401	1,317,655	41	Real-space pseudopotential method

we use symm-ildl which is an external package [11] that has an interface for MATLAB and is robust. Hereafter, we refer to this preconditioner as $ILDL^T$. Therefore, we use it to show that the proposed scheme is competitive against other solvers in terms of number of iterations even when a much more robust preconditioner is used in Sections 4.1 and 4.2.2. For a fair comparison, exactly the same preconditioner is used for BiCGStab, GMRES(m) ($m = 20, 40, 60$ and 120) as well as another outer-inner scheme with Flexible GMRES (FGMRES) as the outer solver and GMRES as the inner solver [16]. In Section 4.2.2, we also use a MINRES preconditioner with the modified $ILDL^T$ factorization. For FGMRES-GMRES we use a restart value of 120 for both inner and outer iterations. Iterative solvers, except FGMRES, are the implementations that are available in MATLAB. We note that MATLAB's BiCGStab implementation terminates early before completing a full iteration if the relative residual is already small enough. This counts as a half iteration. We modified GMRES to stop the iteration based on the true relative residual rather than the preconditioned relative residual. Storage requirements for MINRES, CG, BiCGStab, FGMRES and GMRES are given in [4, 17, 20, 16, 18], respectively. In Table 2, we illustrate the storage requirement of each of the iterative solvers via the number of vectors in addition to the coefficient matrix, the preconditioner (i.e. incomplete factors) and the right hand side vector which are common for all solvers.

Table 2: Total additional memory requirements (number of vectors) of various iterative solver (not counting A , M and b) where m is the restart and k is the number of negative eigenvectors.

MINRES	MINRES-CG	GMRES	FGMRES-GMRES	BiCGStab
7	$11 + k$	$m + 2$	$3m + 4$	6

4.1. Disk brake example. In the following we solve (28) for the small and large test problems of [10] of sizes $n = 4,669$ and $n = 842,638$, respectively, with $\Omega_{ref} = 5$. Note again that (28) is independent of Ω. For the first set of experiments

we fix the shift γ to be the largest value in the range of values of interest, namely $1,000 + 20,000j$. This also happens to be the most challenging case since the number of negative eigenvalues is also the largest, with $k = 18$ and $k = 60$, respectively.

For the proposed scheme, an $ILDL^T$ factorization of the coefficient matrix is used as the preconditioner (M_{cg}) of the inner CG iteration. We use the same preconditioner for BiCGStab, GMRES(m) and FGMRES-GMRES. For the smaller problem, we also use the ILU factorization with no fill-in (i.e. $ILU(0)$) preconditioner of MATLAB.

For all experiments a moderate outer stopping tolerance of relative residual norm less than or equal to 10^{-3} is used. For MINRES-CG and FGMRES-GMRES schemes the inner stopping tolerance is 10^{-2}. For all methods, the maximum (total) number of iterations are $2,000$ and $15,000$ for small and large problems, respectively. In all experiments, the right hand side vector is a random vector of size n.

The required number of iterations for the proposed scheme as well as for baseline algorithms are given in Table 3 for solving the small problem using $ILU(0)$, $ILDL^T(1, 10^{-2})$ and $ILDL^T(1, 10^{-3})$ preconditioners. GMRES(20) reaches the maximum number of iterations without converging (†) irrespective of the preconditioner. When the preconditioner is $ILU(0)$, BiCGStab converges but it requires twice as many iterations as MINRES-CG, while all other solvers reach the maximum number of iterations without converging. Using $ILDL^T(1, 10^{-2})$ and $ILDL^T(1, 10^{-3})$ as the preconditioners, GMRES(m) converges for $m = \{120\}$ and $m = \{40, 60, 120\}$, respectively.

In Table 4, results are presented for solving the large problem using the seven iterative methods with the preconditioners $LDL^T(4, 10^{-4})$, $LDL^T(5, 10^{-5})$ as well as $LDL^T(5, 10^{-6})$. Note that a much smaller dropping tolerance is required for the large problem. Incomplete factors contain 117.1 , 144.5, and 146.8 nonzeros per row which are relatively small considering complete LDL^T factorization would produce 558.4 nonzeros per row. In fact, incomplete factorization may not be an efficient preconditioner for this problem. However, we still include these results here only to show the robustness of the proposed scheme in terms of number of iterations. While for all preconditioners GMRES(m), FGMRES-GMRES and BiCGStab reach the maximum number of iterations without converging, MINRES-CG still converges in 4 outer iterations albeit with a large number of inner iterations.

In Figure 1, the relative residual history is given when the $ILU(0)$ preconditioner is used for three algorithms for the small test problem. Note that for MINRES-CG the relative residual is only available at each outer iteration. Hence, only those are presented in the figure.

As second application we fix the preconditioner to be $ILU(0)$ and vary the shift γ in the complex domain of interest for the small test problem. Here γ is a parameter that we change in the context of the eigenvalue problem. It is of interest to see how the method behaves as γ is varied. In Figure 2, the total number of iterations is presented. Preconditioned BiCGStab fails to converge for some values of γ (shown as the white area in the figure) while MINRES-CG converges for all γ values. Figure 3 depicts the number of outer iterations and the average number of inner iterations for MINRES-CG.

4.2. Test cases from the SuiteSparse matrix collection. In this subsection, the results are presented for systems that are obtained from the SuiteSparse Matrix Collection. In the first set we compare the proposed method against the classical general iterative schemes GMRES(m) and BiCGStab using an incomplete LU factorization based preconditioner. In the second set, we compare against the modified

Table 3: Required number of iterations using various preconditioners and iterative methods for the small system (†: maximum number of iterations is reached without convergence)

Preconditioner	Solver	Outer its.	Inner its. (Avg.)	Total its.
$ILU(0)$	BiCGStab	$1,421.5$	-	$1,421.5$
	GMRES(20)	†	-	†
	GMRES(40)	†	-	†
	GMRES(60)	†	-	†
	GMRES(120)	†	-	†
	FGMRES-GMRES	†	†	†
	MINRES-CG	4	177.75	711
$ILDL^T(1, 10^{-2})$	BiCGStab	$1,337.5$	-	$1,337.5$
	GMRES(20)	†	-	†
	GMRES(40)	†	-	†
	GMRES(60)	†	-	†
	GMRES(120)	7(103)	-	823
	FGMRES-GMRES	1(3)	259.3	778
	MINRES-CG	4	199.5	798
$ILDL^T(1, 10^{-3})$	BiCGStab	143	-	143
	GMRES(20)	†	-	†
	GMRES(40)	8(40)	-	200
	GMRES(60)	2(7)	-	67
	GMRES(120)	1(62)	-	62
	FGMRES-GMRES	1(4)	46.3	185
	MINRES-CG	4	32.8	131

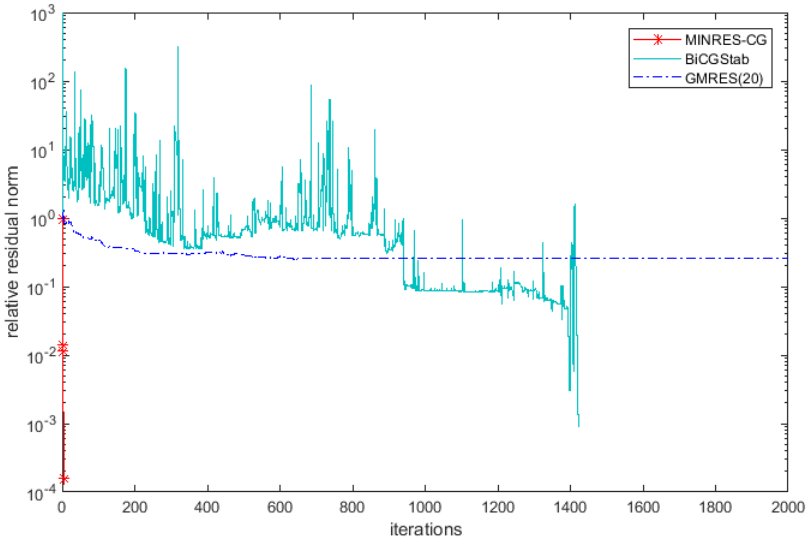
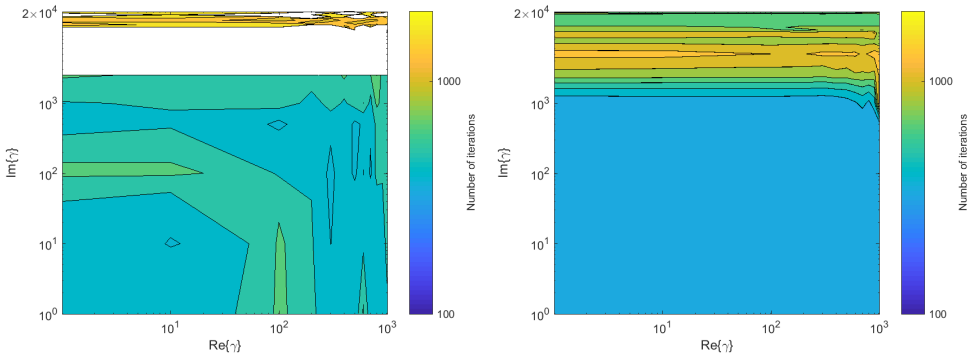


Fig. 1: The relative residual history for MINRES-CG, BiCGStab and GMRES(20).

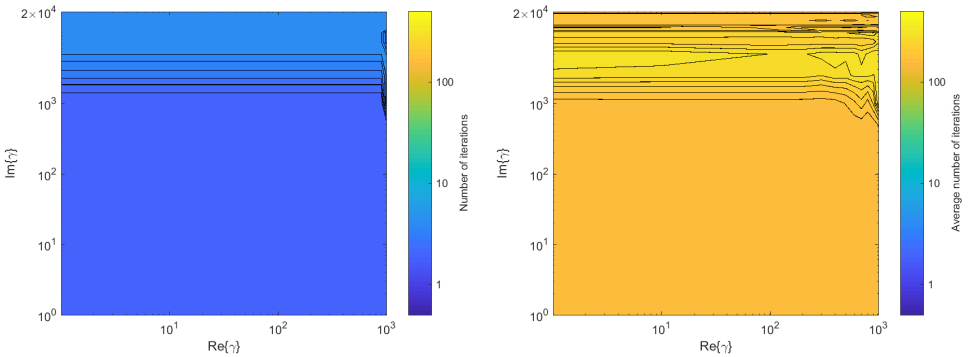
Table 4: Required number of iterations using various preconditioners and iterative methods for the large system (†: maximum number of iterations is reached without convergence)

Preconditioner	Solver	Outer its.	Inner its. (Avg.)	Total its.
$ILDL^T(4, 10^{-4})$	BiCGStab	†	-	†
	GMRES(20)	†	-	†
	GMRES(40)	†	-	†
	GMRES(60)	†	-	†
	GMRES(120)	†	-	†
	FGMRES-GMRES	†	†	†
	MINRES-CG	4	3,032	12,128
$ILDL^T(5, 10^{-5})$	BiCGStab	†	-	†
	GMRES(20)	†	-	†
	GMRES(40)	†	-	†
	GMRES(60)	†	-	†
	GMRES(120)	†	-	†
	FGMRES-GMRES	†	†	†
	MINRES-CG	4	2,221	8,884
$ILDL^T(5, 10^{-6})$	BiCGStab	†	-	†
	GMRES(20)	†	-	†
	GMRES(40)	†	-	†
	GMRES(60)	†	-	†
	GMRES(120)	†	-	†
	FGMRES-GMRES	†	†	†
	MINRES-CG	4	2,242.8	8,971



(a) BiCGStab (b) MINRES-CG

Fig. 2: Total number of iterations for BiCGStab and MINRES-CG using the preconditioner $ILU(0)$. White color indicates that the method failed to converge. GMRES(20) fails for all shifts.



(a) Outer (b) Inner

Fig. 3: Number of outer (MINRES) iterations and average number of inner (CG) iterations for MINRES-CG using the preconditioner $ILU(0)$.

incomplete Bunch-Parlett based preconditioned MINRES.

4.2.1. Comparison against ILU preconditioner. We use $ILU(0)$ for all cases except Meg4 where incomplete LU factorization fails due to a zero pivot. Therefore, we use the modified incomplete LU factorization in MATLAB with 10^{-2} dropping tolerance (i.e. $MILU(10^{-2})$) for this case only. Since in practice GMRES is always used with a value for the restart (m) we choose a restart value of $m = 20, 40, 60$ and 120. In FGMRES-GMRES, we use a restart of 120 for both inner and outer iterations. We stop the iterations when the relative residual norm is less than 10^{-5} for all cases. The inner iteration stops when the relative residual norm is less than 10^{-3} for CG and GMRES, in MINRES-CG and FGMRES-GMRES, respectively. Both in MINRES-CG and BiCGStab iterations stop when the true relative residual is less than the tolerance. For preconditioned GMRES the available residual is only the preconditioned residual. In order to have a fair comparison, we explicitly compute the true residual at each GMRES iteration and stop the iteration based on the true relative residual norm. For all methods, the maximum (total) number of iterations are $20,000$.

In Table 6, the detailed number of iterations for ILU preconditioned MINRES-CG, GMRES(m), FGMRES-GMRES and BiCGStab are given. GMRES(20) fails in 6 cases out of 11. For $bcsstm10$, GMRES(20) stagnates (‡), while for 5 other cases (namely $bcsstm27$, $nasa1824$, $Si10H16$, $SiO(\sigma = 0.25)$ and $Sio(\sigma = 0.75)$), the maximum number of iterations is reached without convergence (†). If the restart is increased to 40, 60 and 120, GMRES(m), fails in 4, 3 and 2 cases, respectively. BiCGStab fails for $bcsstm27$ and $Meg4$ due to the maximum of iterations being reached without convergence (†) and a scalar quantity became too large or too small during the iteration ($*$), respectively. FGMRES-GMRES fails in 3 cases due to the maximum of iterations being reached without convergence (†). The proposed MINRES-CG method does not fail in any of the test problems. Although the cost per iteration is different for each method, the total number of iterations are presented in Table 7. For the cases they do not fail, GMRES(120) and FGMRES-GMRES requires fewer number of iterations than MINRES-CG but they also require more storage. In

4 cases MINRES-CG requires fewer iterations than BiCGStab. It is possible to improve the total number of iterations of MINRES-CG via using the algorithm described in Section 2.1, Table 5 shows the improved number of iterations which is significant especially for the cases where the inner or outer number of iterations are high.

In order to study the effect of the inner stopping tolerance on the eigenvalues of the preconditioned matrix, we explicitly compute $M_{mr}^{-1}A$ using preconditioend CG iterations using stopping tolerances of 10^{-2}, 10^{-3} and 10^{-4}. In Figure 5, a clear clustering of eigenvalues of the preconditioned matrix $M_{mr}^{-1}A$ is visible around $+1$ and -1 for *bcsstm*10 while the unpreconditioned coefficient matrix had no clustering of eigenvalues (see Figure 4). As expected, the clustering around -1 and $+1$ improves as the stopping tolerance for the inner CG iterations is decreased.

Table 5: Comparison of MINRES-CG and MINRES-CG*, the improvement via the Sherman-Morrison-Woodbury formula is given in the second column, both are using the same ILU preconditioner.

	MINRES-CG		MINRES-CG*	
Name	MINRES	CG (Avg.)	MINRES	CG (Avg.)
Bcsstm10	4	650.5	4	470
Bcsstm27	5	3,186.4	4	1,339.5
Nasa1824	4	455.3	4	309.3
Meg4	16	18.6	4	1.5
Benzene	3	24.3	3	22.7
Si10H16	4	831	4	893.5
Si5H12	4	53.8	4	47.3
SiO	4	50.5	4	48.8
SiO($\sigma = 0.25$)	4	259	4	141.8
SiO($\sigma = 0.5$)	4	94.3	4	80
SiO($\sigma = 0.75$)	4	179.5	4	184

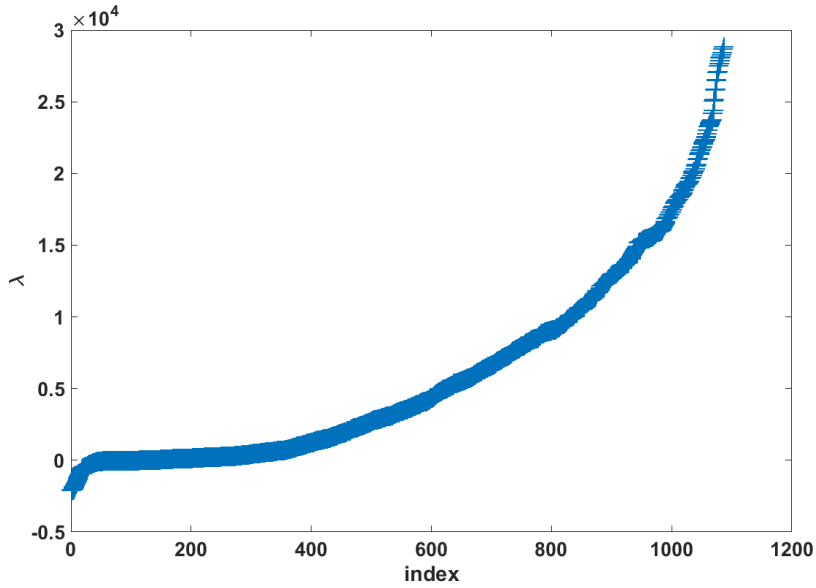


Fig. 4: Eigenvalues of A (*bccstm*10)

4.2.2. Comparisons using incomplete LDL^T. In this subsection, we compare the proposed scheme against a robust incomplete LDL^T (Bunch-Parlett) factorization [6].

We use the $ILDL^T$ implementation of [11] in MATLAB which computes the incomplete Bunch-Parlett factorization of the coefficient matrix. The default parameters are 3 and 10^{-3} for the level of fill-in and the dropping tolerance, respectively. Furthermore, it uses the Approximate Minimum Degree reordering, Rook pivoting and Scaling are used to improve the numerical stability of the incomplete factors by default. Note that all of those enhancements that are implemented in $ILDL^T$ makes the preconditioner much more robust than the $ILU(0)$ preconditioner. In the following experiments all methods are applied to the permuted and scaled linear systems.

After computing the $ILDL^T$ factorization the D matrix is modified as described in [9] in order to obtain a positive definite preconditioner to be used with MINRES. To have a fair comparison, the same $ILDL^T$ factorization (without the modification) is used as the preconditioner for GMRES(m), FGMRES-GMRES, BiCGStab and as the inner preconditioner for MINRES-CG. Stopping tolerances and the maximum number of iterations allowed are set exactly the same as in Section 4.2.

In Table 8 the total number of iterations for all methods are given. Even though it is a much more robust preconditioner, MINRES preconditioned with the modified $ILDL^T$ preconditioner stagnates (‡) for *bcsstm*27. For the same problem BiCGStab, FGMRES-GMRES and GMRES(m) (for all restart values $m = 20, 40, 60, 120$) reach the maximum number of iterations without converging (†). On the other hand, MINRES-CG converges in all problems which confirms the robustness of the proposed scheme. In Table 9, the total number of iterations for all methods are given. GMRES(m) with larger restart values and BiCGStab require the fewest number of iterations. FGMRES-GMRES requires fewer number of iterations than MINRES-CG.

Table 6: Number of iterations using ILU

Name	MINRES-CG		GMRES(m)				FGMRES(m_1)-GMRES(m_2)		BiCGStab
	MINRES	CG	$m = 20$	$m = 40$	$m = 60$	$m = 120$	$m_1 = 120$	$m_2 = 120$	
Bcsstm10	4	650.5	‡	‡	12(60)	1(80)	1(57)	14.5	443.5
Bcsstm27	5	3,186.4	†	†	†	†	†	†	†
Nasa1824	4	455.3	†	101(14)	15(30)	4(78)	1(6)	92	483
Meg4	16	18.6	5(20)	1(38)	1(38)	1(38)	1(3)	28.3	*
Benzene	3	24.3	6(4)	2(1)	1(41)	1(41)	1(2)	30.5	37.5
Si10H16	4	831	†	†	†	†	†	†	6,956.5
Si5H12	4	53.8	347(19)	9(18)	3(14)	1(80)	1(3)	66.7	108.5
SiO	4	50.5	106(5)	6(15)	2(18)	1(70)	1(2)	55	114
SiO ($\sigma = 0.25$)	4	259	†	†	†	25(120)	†	†	2,109.5
SiO ($\sigma = 0.5$)	4	94.3	492(20)	48(31)	12(59)	2(111)	1(3)	116.3	621.5
SiO ($\sigma = 0.75$)	4	179.5	†	51(33)	20(24)	5(56)	1(3)	232.7	1,007

Table 7: Total number of iterations using ILU

Name	MINRES-CG	GMRES(m)				FGMRES(120) -GMRES(120)	BiCGStab
		$m = 20$	$m = 40$	$m = 60$	$m = 120$		
Bcsstm10	2,642	‡	‡	720	80	826	443.5
Bcsstm27	15,932	†	†	†	†	†	†
Nasa1824	1,821	†	4,014	870	438	552	483.0
Meg4	297	100	38	38	38	85	*
Benzene	73	104	41	41	41	61	37.5
Si10H16	3,324	†	†	†	†	†	6,956.5
Si5H12	215	6,939	338	134	80	200	108.5
SiO	202	2,105	215	78	80	110	114
SiO ($\sigma = 0.25$)	1,036	†	†	†	†	†	2,109.5
SiO ($\sigma = 0.5$)	377	9,840	1,911	719	231	349	621.5
SiO ($\sigma = 0.75$)	718	†	2,033	1,164	536	689	1,007

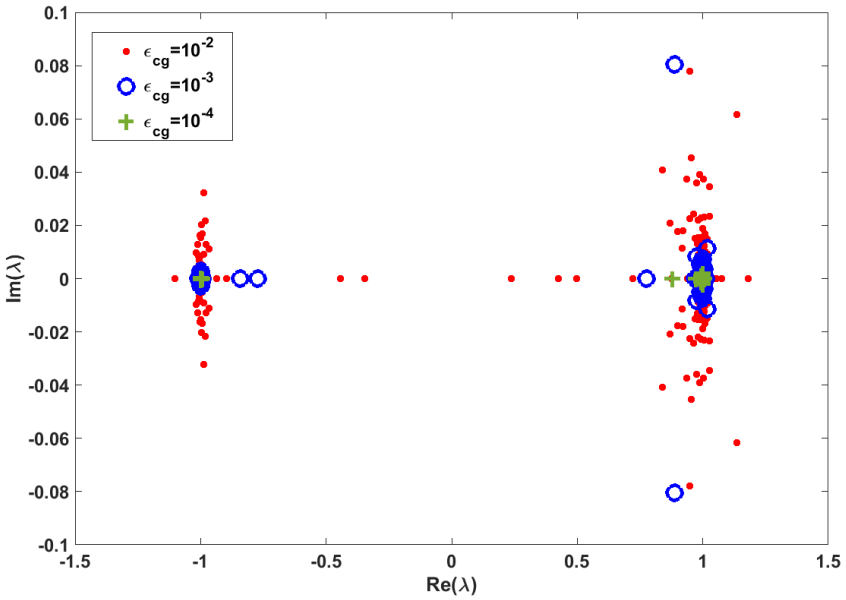


Fig. 5: Eigenvalues of $M_{mr}^{-1}A$ ($\epsilon_{cg} = 10^{-2}$, 10^{-3} and 10^{-4}) ($bccstm10$)

On the other hand, MINRES requires more iterations than MINRES-CG in 4 cases, and the required number of iterations are marginally better than that of MINRES-CG for 3 other cases.

5. Conclusions. A two-level nested iterative scheme is proposed for solving sparse linear systems of equations where the coefficient matrix is symmetric indefinite with few negative eigenvalues. The first level is MINRES preconditioned via CG. The inner level CG is preconditioned via the original indefinite coefficient matrix. The robustness of the proposed scheme is presented for linear systems that arise in disk brake squeal as well as systems that arise in a variety of test cases from the SuiteSparse Matrix Collection.

REFERENCES

[1] O. Axelsson, M. Neytcheva, and B. Ahmad, *A comparison of iterative methods to solve complex valued linear algebraic systems*, Numerical Algorithms, 66 (2014), pp. 811–841, https://doi.org/10.1007/s11075-013-9764-1.

[2] M. Benzi, G. Golub, and J. Liesen, *Numerical solution of saddle point problems*, Acta Numerica, 14 (2005), pp. 1–137.

[3] J. R. Chelikowsky, N. Troullier, and Y. Saad, *Finite-difference-pseudopotential method: Electronic structure calculations without a basis*, Physical review letters, 72 (1994), p. 1240.

[4] S.-C. T. Choi, C. C. Paige, and M. A. Saunders, *Minres-qlp: A krylov subspace method for indefinite or singular symmetric systems*, SIAM Journal on Scientific Computing, 33 (2011), pp. 1810–1836.

[5] T. Davis and Y. Hu, *The university of florida sparse matrix collection*, ACM Trans. Math. Softw., 38 (2011), pp. 1:1–1:25, https://doi.org/10.1145/2049662.2049663.

[6] B. Fischer, *Polynomial Based Iteration Methods for Symmetric Linear Systems*, Society for Industrial and Applied Mathematics, 2011, https://doi.org/10.1137/1.9781611971927.

Table 8: Number of iterations using $ILDL^T$ preconditioner (MINRES uses the modified spd preconditioner)

Name	MINRES-CG		GMRES(m)				FGMRES(m_1)-GMRES(m_2)		BiCGStab	MINRES
	MINRES	CG	$m = 20$	$m = 40$	$m = 60$	$m = 120$	$m_1 = 120$	$m_2 = 120$		
Bcsstm10	4	10.5	1(10)	1(10)	1(10)	1(10)	1(4)	5.5	8.5	37
Bcsstm27	4	2,127	†	†	†	†	†	†	†	‡
Nasa1824	4	19.5	2(11)	1(23)	1(23)	1(23)	1(21)	3.5	32	48
Meg4	2	1.5	1(1)	1(1)	1(1)	1(1)	1(1)	1	0.5	2
Benzene	3	7.7	1(9)	1(9)	1(9)	1(9)	1(3)	4.7	4.5	12
Si10H16	4	72	7(12)	2(13)	1(42)	1(42)	1(3)	27	49.5	1,192
Si5H12	4	17.3	1(17)	1(17)	1(17)	1(17)	1(3)	10.7	11.5	34
SiO	4	18	1(17)	1(17)	1(17)	1(17)	1(3)	10.7	11.5	51
SiO($\sigma = 0.25$)	4	24.8	4(3)	1(31)	1(31)	1(31)	1(3)	18	40.5	80
SiO($\sigma = 0.5$)	4	43.3	12(20)	2(2)	1(41)	1(41)	1(3)	28.3	72	355
SiO($\sigma = 0.75$)	4	116.3	12(19)	2(33)	1(52)	1(52)	1(3)	26.3	85	1,670

Table 9: Total number of iterations using $ILDL^T$ (MINRES uses the modified spd preconditioner)

Name	MINRES-CG	GMRES(m) $m = 20$	$m = 40$	$m = 60$	$m = 120$	FGMRES(120) -GMRES(120)	BiCGStab	MINRES
Bcsstm10	42	10	10	10	10	22	8.5	37
Bcsstm27	8,500	†	†	†	†	†	†	‡
Nasa1824	78	31	23	23	23	74	32	48
Meg4	3	1	1	1	1	1	0.5	2
Benzene	23	9	9	9	9	14	4.5	12
Si10H16	288	132	53	42	42	81	49.5	1,192
Si5H12	69	17	17	17	17	32	11.5	34
SiO	72	17	17	17	17	32	11.5	51
SiO ($\sigma = 0.25$)	99	63	31	31	31	54	40.5	80
SiO ($\sigma = 0.5$)	173	240	42	41	41	85	72	355
SiO ($\sigma = 0.75$)	466	239	73	52	52	79	85	1,670

[7] L. Fox, H. D. Huskey, and J. H. Wilkinson, *Notes on the solution of algebraic linear simultaneous equations*, The Quarterly Journal of Mechanics and Applied Mathematics, 1 (1948), pp. 149–173, https://doi.org/10.1093/qjmam/1.1.149.

[8] A. Gaul, M. H. Gutknecht, J. Liesen, and R. Nabben, *A framework for deflated and augmented krylov subspace methods*, SIAM Journal on Matrix Analysis and Applications, 34 (2013), pp. 495–518.

[9] P. Gill, W. Murray, D. B. Ponceleon, and M. A. Saunders, *Preconditioners for indefinite systems arising in optimization*, SIAM Journal on Matrix Analysis and Applications, 13 (1992), pp. 292–311, https://doi.org/10.1137/0613022.

[10] N. Gräbner, V. Mehrmann, S. Quraishi, C. Schröder, and U. von Wagner, *Numerical methods for parametric model reduction in the simulation of disk brake squeal*, ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik, 96 (2016), pp. 1388–1405.

[11] C. Greif, S. He, and P. Liu, *SYM-ILDL: incomplete ldl^{T} factorization of symmetric indefinite and skew-symmetric matrices*, CoRR, abs/1505.07589 (2015), http://arxiv.org/abs/1505.07589.

[12] C. Greif, S. He, and P. Liu, *Sym-ildl: Incomplete ldlt factorization of symmetric indefinite and skew-symmetric matrices*, ACM Trans. Math. Softw., 44 (2017), pp. 1:1–1:21, https://doi.org/10.1145/3054948.

[13] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK Users Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM, 1998.

[14] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM Journal on Numerical Analysis, 12 (1975), pp. 617–629, https://doi.org/10.1137/0712047.

[15] M. Rozloznk and V. Simoncini, *Krylov subspace methods for saddle point problems with indefinite preconditioning*, SIAM Journal on Matrix Analysis and Applications, 24 (2002), pp. 368–391, https://doi.org/10.1137/S0895479800375540.

[16] Y. Saad, *A flexible inner-outer preconditioned gmres algorithm*, SIAM Journal on Scientific Computing, 14 (1993), pp. 461–469.

[17] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Society for Industrial and Applied Mathematics, second ed., 2003, https://doi.org/10.1137/1.9780898718003.

[18] Y. Saad and M. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on scientific and statistical computing, 7 (1986), pp. 856–869.

[19] H. A. van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 13 (1992), pp. 631–644, https://doi.org/10.1137/0913035.

[20] H. A. Van der Vorst, *Bi-cgstab: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems*, SIAM Journal on scientific and Statistical Computing, 13 (1992), pp. 631–644.

[21] E. Vecharynski and A. Knyazev, *Absolute value preconditioning for symmetric indefinite linear systems*, SIAM Journal on Scientific Computing, 35 (2013), pp. A696–A718, https://doi.org/10.1137/120886686.

[22] S. Wang, E. d. Sturler, and G. H. Paulino, *Large-scale topology optimization using preconditioned krylov subspace methods with recycling*, International journal for numerical methods in engineering, 69 (2007), pp. 2441–2468.