Abstract. In this paper we examine iterative methods for solving the forward \((Ax = b)\) and adjoint \((A^T y = g)\) systems of linear equations used to approximate the scattering amplitude, defined by \(g^T x = y^T b\). Based on an idea first proposed by Gene Golub, we develop a conjugate gradient-like iteration for a nonsymmetric saddle point matrix that is constructed so as to have a real positive spectrum. Numerical experiments highlight contrasts, both positive and negative, in the performance of the proposed method compared to known methods for computing the scattering amplitude such as GLSQR or QMR. We then examine an approach to preconditioning and compare the performance to that of state-of-the-art iterative methods for nonsymmetric systems.

Key words. nonsymmetric saddle point matrix, conjugate gradient method, scattering amplitude

AMS subject classifications. 65F10, 65F08

1. Introduction.

1.1. The Scattering Amplitude Problem. The core objective of this paper is to design and implement an iterative method for the solution of a system where the coefficient matrix is large, sparse, and nonsymmetric. One application in which such a system arises is in the computation of the scattering amplitude. The scattering amplitude, in quantum physics, is the amplitude of the outgoing spherical wave relative to that of the incoming plane wave [8]. It is useful when it is of interest to know what is reflected when a radar wave is impinging on a certain object. The scattering amplitude can be computed by taking the inner product of the right hand side vector \(g\) of the adjoint system

\[ A^T y = g \]  

and the solution \(x\) of the forward system

\[ Ax = b. \]  

Applications of the scattering amplitude come up in nuclear physics [1], quantum mechanics [14], and computational fluid dynamics (CFD) [5]. One particular application is in the design of stealth planes [1].

The scattering amplitude \(g^T x = y^T b\) creates a relationship between the right hand side of the adjoint system and the solution to the forward system in signal processing. The field \(x\) is determined from the signal \(b\) in the system \(Ax = b\). Then the signal is received on an antenna characterized by the vector \(g\) which is the right hand side of the adjoint system \(A^T y = g\), and it is expressed as \(g^T x\) [8] or, equivalently, \(y^T b\). We are interested in efficiently approximating the scattering amplitude. It is informative to look at methods that other researchers have used to solve this problem, which will be discussed below.

In general, the solution of the linear system (1.2) is important for many applications beyond the scattering amplitude, such as in the numerical solution of PDE.
This solution can be obtained in many different ways, depending on the properties of the matrix $A$. The $LDL^T$ factorization can be used to solve some problems with a symmetric matrix or a Cholesky factorization can be used if the matrix is also known to be positive definite [10]. However, for large, sparse systems, an iterative method is preferred.

The conjugate gradient method is the preferred iterative method for a symmetric positive definite matrix $A$ [10]. However, it is much more difficult to find this solution for a matrix that is not symmetric positive definite. In the case that we have a matrix that is not even symmetric, as in the scattering amplitude problem, we can use methods like the biconjugate gradient (BiCG) [4] and generalized minimal residual (GMRES) methods [17]. Since the scattering amplitude depends on both the forward and adjoint problem, it makes sense to use methods that take both the forward and adjoint problems into account, like the quasi-minimal residual (QMR) [16] and generalized least squares residual (GLSQR) methods [25].

1.2. Approximation of the Scattering Amplitude. The method of this paper employs a conjugate gradient-like approach since, for large, sparse matrices, it is best to use an iterative approach, such as the conjugate gradient method [12] which is particularly effective for symmetric positive definite matrices. In particular, conjugate gradient has a very rapid convergence if $A$ is near the identity either in the sense of a low rank perturbation or in the sense of the norm [10].

Multiplying both sides of $Ax = b$ by $A^T$ yields the normal equations with a symmetric matrix $A^TA$ that is also positive definite when $A$ is invertible. However, this approach is not conducive to solving the forward and adjoint problems simultaneously. Furthermore, a significant problem with using $A^TA$ is that now the condition number in the two-norm is squared for $A^TA$. Since this increases the sensitivity of the system, possibly making it ill-conditioned, this paper explores an alternative approach. The idea is to transform the problems $Ax = b$ and $A^Ty = g$ into an equivalent system in which the matrix can be guaranteed to have real, positive eigenvalues, as well as eigenvectors that are in some sense orthogonal, which is then conducive to solution using a conjugate gradient-like iteration. It is not necessarily symmetry that we seek, but we will have symmetry with respect to some inner product. To this end, we use an idea first proposed by Gene Golub [6], and consider a nonsymmetric saddle point matrix that has the form

$$M = \begin{bmatrix} A^TW & A^T \\ -A & 0 \end{bmatrix}.$$  

As required by the definition of a nonsymmetric saddle point matrix, we assume that the matrix $W$ is symmetric positive definite. The goal is to choose $W$ so that we can guarantee $M$ has real, positive eigenvalues. In this paper we will introduce the nonsymmetric saddle point conjugate gradient (NspCG) method to solve a nonsymmetric, large, sparse linear system, which will then allow us to compute the scattering amplitude. We will also use a variation of ILU preconditioning with NspCG, which gives rapid convergence compared to existing methods for solving such systems.

This paper is organized as follows. In Section 2 we discuss the known methods for solving a large linear system with iterative approaches to compute the scattering amplitude such as Bidiagonalization or least squares QR (LSQR), quasi minimum residual (QMR), and block generalized LSQR (GLSQR). In Section 3 we will introduce the method of this paper, NspCG. Section 4 will include an analysis of the
numerical results. Preconditioning is discussed in Section 5, with accompanying numerical experiments. Conclusions and discussion of possible future work is given in Section 6.

2. Methods for Solving the Linear Systems of the Forward and Adjoint Problems. In this section, we review some iterative methods that have previously been applied to the scattering amplitude problem [8].

2.1. QMR approach. The QMR approach [16, 8] is based on the spectral decomposition $A = XD X^{-1}$; also the basis of the QMR approach is the unsymmetric Lanczos [10, 18] process which generates two sequences

$$V_k = \begin{bmatrix} v_1 & v_2 & \ldots & v_k \end{bmatrix}$$

$$W_k = \begin{bmatrix} w_1 & w_2 & \ldots & w_k \end{bmatrix}$$

that are biorthogonal, meaning $V_k^T W_k = I$. We have the following relations:

$$AV_k = V_{k+1} T_{k+1,k}, \quad \text{(2.1)}$$

$$A^T W_k = W_{k+1} \hat{T}_{k+1,k}. \quad \text{(2.2)}$$

where the tridiagonal matrices

$$T_{k+1,k} = \begin{bmatrix} \alpha_1 & \gamma_1 & & & & \beta_1 & \alpha_2 & \gamma_2 & & & & \beta_2 & \ddots & \ddots & & & & \beta_{k-1} & \gamma_{k-1} & & & & \beta_k & \alpha_k & \beta_k \end{bmatrix} = \begin{bmatrix} T_{k,k} & \beta_k e_k^T \end{bmatrix}$$

and

$$\hat{T}_{k+1,k} = \begin{bmatrix} \hat{\alpha}_1 & \hat{\gamma}_1 & & & & \hat{\beta}_1 & \hat{\alpha}_2 & \hat{\gamma}_2 & & & & \hat{\beta}_2 & \ddots & \ddots & & & & \hat{\beta}_{k-1} & \hat{\gamma}_{k-1} & & & & \hat{\beta}_k & \hat{\alpha}_k & \hat{\beta}_k \end{bmatrix} = \begin{bmatrix} \hat{T}_{k,k} & \hat{\beta}_k e_k^T \end{bmatrix}$$

have block structures in which $T_{k,k}$ and $\hat{T}_{k,k}$ are not necessarily symmetric.

The residual, $r = b - Ax$, in each iteration can be expressed as

$$\|r_k\| = \|b - Ax_k\|$$

$$= \|b - Ax_0 - AV_k c_k\|$$

$$= \|r_0 - V_{k+1} T_{k+1,k} c_k\|$$

$$= \|V_{k+1}(|\|r_0\| e_1 - T_{k+1,k} c_k|)\| \quad \text{(2.3)}$$

with a choice of $v_1 = \frac{r_0}{\|r_0\|}$ where $r_0 = b - Ax_0$ and $x_k = x_0 + V_k c_k$. We now have the quasi-residual $\|r_k^Q\| = \|\|r_0\| e_1 - T_{k+1,k} c_k\|$. Then we choose $w_1 = \frac{\alpha_0}{\|\alpha_0\|}$,
where \( s_0 = g - A^T y_0 \) and \( y_k = y_0 + w_k d_k \). Then the adjoint residual is \( ||s_k^Q|| = ||s_0||e_1 - \hat{T}_{k+1,k}d_k || \). The vectors \( c_k \) and \( d_k \) are the solutions of the least squares problems for minimizing \( ||r_k^Q|| \) and \( ||s_k^Q|| \). So now the solutions can be defined as
\[
\begin{align*}
x_k &= x_0 + V_k c_k \\
y_k &= y_0 + U_k d_k.
\end{align*}
\]

2.2. LSQR approach. In LSQR [8, 19], a truncated bidiagonalization is used in order to solve the forward and adjoint problems approximately. The bidiagonal factorization of \( A \) is given by \( A = UBV^T \) where \( U \) and \( V \) are orthogonal and \( B \) is bidiagonal. Thus the forward and adjoint systems can be written as
\[
\begin{align*}
UBV^T x &= b \\
V^T U^T y &= g.
\end{align*}
\]

Now we can solve (2.6) by solving the following two systems
\[
\begin{align*}
Bz &= U^T b \\
x &= V^T z,
\end{align*}
\]

and we can solve (2.7) by solving
\[
\begin{align*}
B^T w &= V^T g \\
y &= U^T w.
\end{align*}
\]

We need to use the following recurrence relations in an iterative process to produce a bidiagonal matrix
\[
\begin{align*}
AV_k &= U_{k+1}B_k \\
A^T U_{k+1} &= V_k B_k^T + \alpha_{k+1} v_{k+1} e_{k+1}^T
\end{align*}
\]

where \( V_k \) and \( U_k \) are matrices with orthonormal columns, and
\[
B_k = \begin{bmatrix}
\alpha_1 \\
\beta_2 & \alpha_2 \\
& \beta_3 & \ddots \\
& & \ddots & \alpha_k \\
& & & \beta_{k+1}
\end{bmatrix}.
\]

Also we have that
\[
\begin{align*}
A^T A V_k &= A^T U_{k+1} B_k = (V_k B_k^T + \alpha_{k+1} v_{k+1} e_{k+1}^T) B_k \\
&= V_k B_k^T B_k + \hat{\alpha}_k v_{k+1} e_{k+1}^T
\end{align*}
\]

and
\[
\hat{\alpha}_{k+1} = \alpha_{k+1} \beta_{k+1}.
\]

Because \( B_k \) is bidiagonal, it follows that \( B_k^T B_k \) is symmetric and tridiagonal. It can be seen from (2.14) that (2.12) and (2.13) implicitly apply Lanczos iteration to
Now this iterative process can be used to obtain the approximate solution to the forward and adjoint systems. We define the residuals at step $k$ as

\[ r_k = b - Ax_k \]  
\[ s_k = g - A^T y_k \]  

where

\[ x_k = x_0 + V_k z_k \quad y_k = y_0 + U_{k+1} w_k. \]

The goal of the LSQR approach is to obtain an approximation that minimizes the norm of the residual. That is, the norm $\|r_k\| = \|b - Ax_k\|$ is minimized. When working with the forward and adjoint problems, this approach is limited due to the relationship between the starting vectors

\[ A^T u_1 = \alpha_1 v_1. \]

2.3. Generalized LSQR (GLSQR). The GSLQR method [8, 25] overcomes the disadvantage of the LSQR method by choosing starting vectors $u_1 = \frac{r_0}{\|r_0\|}$ and $v_1 = \frac{s_0}{\|s_0\|}$ independently where, for an initial guess of $x_0$ and $y_0$, $r_0 = b - Ax_0$ and $s_0 = g - A^T y_0$. It is based on the factorizations

\[ AV_k = U_{k+1} T_{k+1,k} = U_k T_{k,k} + \beta_{k+1} u_{k+1} e_1^T \]
\[ A^T U_k = V_{k+1} S_{k+1,k} = V_k S_{k,k} + \eta_{k+1} v_{k+1} e_1^T \]

From the above we get that

\[ \beta_{k+1} u_{k+1} = A v_k - \alpha_k u_k - \gamma_{k-1} u_{k-1} = c_k \]
\[ \eta_{k+1} v_{k+1} = A^T u_k - \delta_k v_k - \theta_{k-1} v_{k-1} = d_k, \]

where the recursion coefficients $\alpha_k, \gamma_k, \eta_k,$ and $\theta_k$ are chosen to make $U_k$ and $V_k$ have orthonormal columns, which yields

\[ \alpha_k = u_k^T A v_k, \]
\[ \gamma_k = u_{k-1}^T A v_{k+1}, \]
\[ \delta_k = v_k^T A^T u_k, \]
\[ \theta_k = v_{k+1}^T A^T u_{k+1}. \]

We can define $u_{k+1} = \frac{c_k}{\beta_k}$ and $v_k = \frac{d_k}{\eta_k}$, where $\beta_k = \|c_k\|$, and $\eta_k = \|d_k\|$. Now we have that

\[ T_{k+1,k} = \begin{bmatrix} \alpha_1 & \gamma_1 \\ \beta_2 & \alpha_2 \\ \vdots & \ddots \\ \beta_k & \gamma_{k-1} \\ \beta_{k+1} & \alpha_k \end{bmatrix}, \quad S_{k+1,k} = \begin{bmatrix} \delta_1 & \theta_1 \\ \eta_2 & \delta_2 \\ \vdots & \ddots \\ \eta_k & \delta_{k-1} \\ \eta_{k+1} & \theta_{k-1} \end{bmatrix}. \]

The residuals can be expressed as follows

\[ \|r_k\| = \|r_0 - U_{k+1} T_{k+1,k} x_k\| = \|r_0\| e_1 - T_{k+1,k} x_k\|, \]
and

\[ \|s_k\| = \|s_0 - V_kS_{k+1,k}y_k - \alpha_{k+1}v_{k+1}^Te_{k+1}y_k\|. \]  (2.26)

The solutions \( x_k \) and \( y_k \) are

\[ x_k = x_0 + \|r_0\|V_kT_{k,k}^{-1}e_1 \]  (2.27)

\[ y_k = y_0 + \|s_0\|U_kS_{k,k}^{-1}e_1. \]  (2.28)

3. Iterative Methods for Nonsymmetric Saddle Point Matrices. All of the methods in the Section 2 have a significant storage requirement, as the bases generated by the iteration are required, in their entirety, to produce each new set of iterates. For an alternative approach, we consider the matrix \( M \), defined as follows

\[ M \equiv \begin{bmatrix} A^TWA & A^T \\ -A & 0 \end{bmatrix}, \]  (3.1)

where \( A \in \mathbb{R}^{n \times n} \) is invertible and \( W \) is a symmetric positive definite matrix, is an example of a nonsymmetric saddle point matrix. It can be shown that \( x^TMx \geq 0 \) for all \( x \neq 0 \). To see this, we first let \( x = \begin{bmatrix} y \\ z \end{bmatrix} \). Then \( x^TMx \) can be written as

\[ x^TMx = \begin{bmatrix} y^T & z^T \end{bmatrix} \begin{bmatrix} A^TWA & A^T \\ -A & 0 \end{bmatrix} \begin{bmatrix} y \\ z \end{bmatrix} = y^T(A^TWA)y \geq 0 \]

because \( W \) is symmetric positive definite, with equality if and only if \( y = 0 \).

3.1. Ensuring a Real Positive Spectrum. We want to choose \( W \) so that the matrix \( M \) has a real positive spectrum, so it is suitable for a conjugate gradient-like iteration [15]. To make this choice we need to first define

\[ M(\gamma) \equiv Jp(M) = J(M - \gamma I) = \begin{bmatrix} A^TWA - \gamma I & A^T \\ A & \gamma I \end{bmatrix}, \]

where \( p \) is a polynomial of degree one in the form \( p(\zeta) = \zeta - \gamma \) for \( \gamma \in \mathbb{R} \) and

\[ J = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix}. \]

The goal here is to determine if there exists a symmetric positive definite matrix \( M(\gamma) \) with respect to which \( M \) is symmetric, meaning that \( M \) is \( M(\gamma) \)-symmetric if \( M(\gamma)M = M^TM(\gamma) = (M(\gamma)M)^T \).

Let us first define a generic nonsymmetric saddle point matrix

\[ \mathcal{A} = \begin{bmatrix} \hat{A} & \hat{B}^T \\ -\hat{B} & \hat{C} \end{bmatrix}. \]

and then define \( \mathcal{M}(\gamma) = Jp(\mathcal{A}) \). We can use the following results from [15] to determine how to obtain a real positive spectrum:

**Lemma 3.1.** Let the matrix

\[ J = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix} \]
be conformally partitioned with $A$. Then 

(1) $A$ is $J$-symmetric, i.e., $JA = A^T J = (J A)^T$, and for any polynomial $p$,
(2) $p(A)$ is $J$-symmetric, i.e., $Jp(A) = p(A^T) J = (J p(A))^T$, and
(3) $A$ is $Jp(A)$-symmetric, i.e., $(Jp(A)) A = A^T (p(A)^T) J = (Jp(A) A)^T$.

**Theorem 3.2.** The symmetric matrix $M(\gamma)$ is positive definite if and only if

$$\lambda_{\min}(\hat{A}) > \gamma > \lambda_{\max}(\hat{C})$$  \hspace{1cm} (3.2)

where $\lambda_{\min}$ and $\lambda_{\max}$ denote the smallest and largest eigenvalues, respectively, and

$$\|((\gamma I - \hat{C})^{-1/2} \hat{B} (\hat{A} - \gamma I)^{-1/2})\|_2 < 1. \hspace{1cm} (3.3)$$

A sufficient condition that makes $M(\gamma)$ positive definite can be derived from the above theorem.

**Corollary 3.3.** The matrix $M(\gamma)$ is symmetric positive definite when (3.2) holds, and, in addition,

$$\|\hat{B}\|_2^2 < (\lambda_{\min}(\hat{A}) - \gamma)(\gamma - \lambda_{\max}(\hat{C})). \hspace{1cm} (3.4)$$

For $\gamma = \hat{\gamma} \equiv \frac{1}{2}(\lambda_{\min}(\hat{A}) + \lambda_{\max}(\hat{C}))$, the right hand side of (3.4) is maximal and (3.4) reduces to

$$2\|\hat{B}\|_2 < (\lambda_{\min}(\hat{A}) - \lambda_{\max}(\hat{C})). \hspace{1cm} (3.5)$$

The preceding results lead to a simple approach to determining whether $A$ is suitable for a conjugate gradient-like iteration [15].

**Corollary 3.4.** If there exists a $\gamma \in \mathbb{R}$ so that $M(\gamma)$ is positive definite, then $A$ has a nonnegative real spectrum and a complete set of eigenvectors that are orthonormal with respect to the inner product defined by $M(\gamma)$. In case $\hat{B}$ has full rank, the spectrum of $A$ is real and positive.

In this paper, we are limiting ourselves to the case $\hat{C} = 0$; the case of nonzero $\hat{C}$ will be discussed in the conclusion. We can then make use of the following simpler criterion from [2, Proposition 3.1] for ensuring positive definiteness of $M(\gamma)$:

**Proposition 3.5.** Let $\hat{A}$ be symmetric positive definite. Let $\gamma = \frac{1}{2} \lambda_{\min}(\hat{A})$. If

$$\lambda_{\min}(\hat{A}) > 4\lambda_{\max}(\hat{B} \hat{A}^{-1} \hat{B}^T), \hspace{1cm} (3.6)$$

then $M(\gamma)$ is positive definite and $M$ is diagonalizable.

Using the previous results from [2, 15], we obtain a simple criterion for determining whether the matrix $M$ from (3.1) can be constructed in such a way as to satisfy the criterion in Corollary 3.4.

**Theorem 3.6.** Let $A$ be an invertible $n \times n$ real matrix, and let $W$ be a symmetric positive definite $n \times n$ matrix that satisfies

$$\sigma_{\min}(W) > \frac{2}{\sigma_{\min}(A)}.$$ \hspace{1cm} (3.7)

Then the matrix $M$ defined by

$$M = \begin{bmatrix} A^T W A & A^T \\ -A & 0 \end{bmatrix}$$
A. C. SUMNER AND J. V. LAMBERS

has real positive eigenvalues and eigenvectors that are orthogonal with respect to the inner product defined by $M(\gamma) = Jp(M)$. That is, the above selection of $W$ makes the matrix $M$ suitable for a conjugate gradient-like iteration.

Proof: We need to satisfy (3.6) from Proposition 3.5. With that in mind, let

$$\gamma = \frac{1}{2} \lambda_{\min}(A^TWA).$$

From the fact that $A^TWA$ is symmetric positive definite, we obtain

$$\frac{1}{\lambda_{\min}(A^TWA)} = \rho((A^TWA)^{-1})$$

$$= \| (A^TWA)^{-1} \|_2$$

$$\leq \| A^{-1} \|_2 \| W^{-1} \|_2$$

$$\leq \frac{1}{\sigma_{\min}(A)^2 \sigma_{\min}(W)}.$$  

Therefore, (3.6) is satisfied if

$$\sigma_{\min}(A)^2 \sigma_{\min}(W) > \frac{4}{\sigma_{\min}(W)} \quad (3.8)$$

or, equivalently, if (3.7) is satisfied. □

It follows that the matrix $W$ satisfies the requirements to make $M(\gamma)$ be symmetric positive definite and that $A = M$ has a real, positive spectrum from Corollary 3.4. This result makes the matrix suitable for a conjugate gradient-like iteration, as will be described below.

3.2. The Case $W = wI$. Let $A = U \Sigma V^T$ be the SVD of $A$, where

$$U = [ u_1 \quad \cdots \quad u_n ] , \quad V = [ v_1 \quad \cdots \quad v_n ]$$

and $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n)$. In the case $W = wI$ for some scalar $w$, the condition from Theorem 3.6 reduces to

$$w > \frac{2}{\sigma_n}. \quad (3.9)$$

We now study the eigensystem of $M$. Let $Mx_j = \lambda_j x_j$ for $j = 1, 2, \ldots, 2n$, where $x_j = [ y_j^T \quad z_j^T ]^T$. The form of $M$ from (3.1), with $W = wI$, yields

$$w A^T Ay_j + A^T z_j = \lambda_j y_j , \quad (3.10)$$

$$-A y_j = \lambda_j z_j \quad (3.11)$$

for $j = 1, 2, \ldots, 2n$. Substituting (3.11) into (3.10) yields

$$(1 - w \lambda_j) A^T z_j = \lambda_j y_j. \quad (3.12)$$

Multiplying through by $A$ and applying (3.11), we obtain

$$(w \lambda_j - 1) A A^T z_j = \lambda_j^2 z_j.$$  

It follows that each $z_j$ is a multiple of a left singular vector of $M$, and $\lambda_j^2/(w \lambda_j - 1)$ is the square of the corresponding singular value. Furthermore, from (3.11), we find that $y_j$ is a multiple of a right singular vector of $M$.  


We conclude that the eigenvectors $x_1, x_2, \ldots, x_{2n}$ of $M$ are given by

$$x_{2j-1} = \begin{bmatrix} -\lambda_j^+ v_j \\ \sigma_j u_j \end{bmatrix}, \quad x_{2j} = \begin{bmatrix} -\lambda_j^- v_j \\ \sigma_j u_j \end{bmatrix}, \quad j = 1, 2, \ldots, n,$$

(3.13)

with corresponding eigenvalues $\lambda = \lambda_j^+, \lambda_j^-$ that satisfy the quadratic equation

$$\lambda^2 - \sigma_j^2 w \lambda + \sigma_j^2 = 0.$$

(3.14)

It can be shown directly from (3.13) and (3.14) that these eigenvalues are real and positive, and the corresponding eigenvectors linearly independent, if and only if $w$ satisfies the condition (3.9), which is consistent with the necessary and sufficient condition for $M(\gamma)$ to be positive definite given in Theorem 3.2.

The spectrum of $M$ exhibits interesting behavior, as shown in Figure 3.1. The $n$ smallest eigenvalues are nearly constant, with values very close to $1/w$, as can be confirmed by inspecting the roots of (3.14). Then, the eigenvalues increase sharply in magnitude, particularly if $A$ is ill-conditioned.

![Fig. 3.1. Eigenvalues of matrix $M$ from Example 1](image)

### 3.3. Nonsymmetric Saddle Point Conjugate Gradient Method

Let $A \in \mathbb{R}^{n \times n}$ be nonsymmetric. We will now introduce a Conjugate Gradient (CG) approach that solves the linear system $Ax = c$ by solving an equivalent system of the form $Mz = b$, where

$$M \equiv \begin{bmatrix} A^TWA & A^T \\ -A & 0 \end{bmatrix}.$$

(3.15)
The matrix $M$ is also not symmetric; however, the spectrum is entirely contained in the right half of the complex plane, due to the fact that $x^T M x \geq 0$ for all $x$. In the preceding discussion, we established that if $W$ was chosen so as to satisfy the assumptions of Theorem 3.6, then $M$ is diagonalizable with real, positive eigenvalues. Furthermore, the bilinear form $(u,v)_G = u^T G v$, where $G = M(\gamma) = J p(M)$, is a proper inner product, as $G$ is symmetric positive definite. It follows that $M$ is $G$-symmetric and $G$-definite, meaning that $(Mu, v)_G = (u, Mv)_G$ for all $u, v \in \mathbb{R}^{2n}$, and $(u, Mu)_G > 0$ for all $u \neq 0$.

Let the vectors $p$ and $b$ be defined by

$$b = \begin{bmatrix} A^T W c + d \\ -c \end{bmatrix}, \quad p = \begin{bmatrix} d \\ 0 \end{bmatrix},$$

where $Ax = c$, $Mz = b$, and $p^T z = d^T x$ is the scattering amplitude for given vectors $c$ and $d$ that represent the field and antenna, respectively. The following conjugate gradient method is based on a given inner product $(u, v)_G = v^T G u$ for solving the linear system $M x = b$.

**Algorithm 3.1**

**Input:** System matrix $M$, right hand side vector $b$, inner product matrix $W$, initial guess $x_0$

**Require:** $r_0 = b - Mx_0$

**for** $i = 0, 1, \ldots$ **until convergence** **do**

$$\alpha_i = \frac{(x - x_i, p_i)_G}{(p_i, p_i)_G},$$

$$x_{i+1} = x_i + \alpha_i p_i,$$

$$r_{i+1} = r_i - \alpha_i M p_i,$$

$$\beta_{i+1} = \frac{(r_{i+1}, r_{i+1})_G}{(p_i, p_i)_G},$$

$$p_{i+1} = r_{i+1} + \beta_{i+1} p_i$$

**end for**

We have the inner product matrix $G = M(\gamma) M$ suggested by [15]. From [15], we see that this choice of $G$ gives a working CG from the following lemma.

**Lemma 3.7.** Suppose that the symmetric matrix $M(\gamma)$ is positive definite. Then Algorithm 3.1 is well defined for $M$ and $G = M(\gamma) M$, and (until convergence) the scalars $\alpha_i$ and $\beta_{i+1}$ can be computed as

$$\alpha_i = \frac{(r_i, r_i)_{M(\gamma)}}{(M p_i, p_i)_{M(\gamma)}},$$

$$\beta_{i+1} = \frac{(r_{i+1}, r_{i+1})_{M(\gamma)}}{(M r_i, r_i)_{M(\gamma)}}.$$  

(3.17)  

(3.18)

With this choice of inner product matrix, it can be shown that the residuals computed using the preceding algorithm are, in some sense, orthogonal.

**Theorem 3.8.** Each residual $r_k$ as defined in Algorithm 3.1 is orthogonal to all previous residuals with respect to $M(\gamma)$, i.e. $(r_i^T, r_j)_{M(\gamma)} = 0$, where $i \neq j$.

**Proof:** We know that $r_{i+1} = r_i - \alpha_i M p_i$. Let $\alpha_i$ be defined as in (3.17). Also, we know that all of the search directions are orthogonal, i.e. $p_i^T M(\gamma) M p_j = 0$ for $i \neq j$. We want to show that $r_i M(\gamma) r_j = 0$. This will be shown by induction, where the base case that we need to establish is

$$r_{i+1}^T M(\gamma) r_i = 0, \quad i = 0, 1, \ldots$$

(3.19)
To show this we use the definition of \( \alpha_i \) and the expression for the search directions in the above algorithm, \( r_{i+1} = r_i - \alpha_i M p_i \). Now we have that

\[
r_{i+1}^T M(\gamma) r_i = r_i^T M(\gamma) r_i - \frac{r_i^T M(\gamma) r_i}{p_i^T M^T M(\gamma) p_i} p_i^T M^T M(\gamma) r_i.
\] (3.20)

Reindexing the definition of the residual from the algorithm yields the following expression for \( r_i \)

\[
r_i = p_i - \beta_i p_{i-1}.
\]

Substituting this into (3.20) gives

\[
r_{i+1}^T M(\gamma) r_i = r_i^T M(\gamma) r_i - \frac{r_i^T M(\gamma) r_i}{p_i^T M^T M(\gamma) p_i} (p_i^T M^T M(\gamma) p_i - \beta_i p_{i-1}^T M^T M(\gamma) p_i)
\] (3.21)

Rearranging the last term in (3.21) yields

\[
p_{i-1}^T M^T M(\gamma) \beta_i p_i = \beta_i p_i^T M(\gamma) M p_{i-1} = 0
\]

because \( M(\gamma) \) is symmetric, and we already know that the search directions \( p_i \) are orthogonal with respect to \( M(\gamma) \). Now it is easy to see that the denominator in (3.21) and the last factor in the numerator cancel leaving

\[
r_{i+1}^T M(\gamma) r_i = r_i^T M(\gamma) r_i - r_i^T M(\gamma) r_i = 0.
\]

Now we need to show that each residual is orthogonal to all previous residuals. We will do this by showing \( r_i^T M(\gamma) r_{1:d} = 0 \), where \( d > 1 \). Our induction hypothesis is \( r_{i-1}^T M(\gamma) r_{1:d} = 0 \). To show this, first shift the indices to get the expression

\[
r_i = r_{i-1} - \alpha_{i-1} M p_{i-1}.
\]

Rearranging the recurrence relation for the search directions yields

\[
r_{i-1} = p_{i-1} - p_{i-1-d} \beta_{i-1-d}.
\]

Using this expression for \( r_i \) and \( r_{i-1} \) we obtain

\[
r_i^T M(\gamma) r_{1:d} = r_{i-1}^T M(\gamma) r_{1:d} - \alpha_{i-1} p_{i-1}^T M(\gamma) (p_{1:d} - p_{1:d-1} \beta_{1:d-1})
\]

\[
= r_{i-1}^T M(\gamma) r_{1:d} - \alpha_{i-1} p_{i-1}^T M^T M(\gamma) p_{1:d-1} + \alpha_{i-1} p_{i-1}^T M^T M(\gamma) p_{1:d-1} \beta_{1:d-1},
\] (3.22)

where

\[
r_{i-1}^T M(\gamma) r_{1:d} = 0
\]

by the induction hypothesis. Now we are left with

\[
r_i^T M(\gamma) r_{1:d} = -\alpha_{i-1} p_{i-1}^T M^T M(\gamma) p_{1:d} + \alpha_{i-1} p_{i-1}^T M^T M(\gamma) p_{1:d-1} \beta_{1:d-1} = 0,
\]

where both terms are 0 due to the orthogonality of the search directions. □
4. Numerical Results. In this section, we will analyze the results from the methods described in this paper. These methods include QMR from Section 2.2, GLSQR from Section 2.3, and NspCG from Section 3.1. The examples included are from [8].

4.1. Example 1. This example uses the matrix created by 
\[ A = \text{sprand}(n,n,0.2) + \text{speye}(n) \]
in MATLAB where \( n = 100 \). This creates a random sparse \( n \times n \) matrix, where 0.2 is the density of uniformly distributed nonzero entries, and adds this to the identity.

In Figure 4.1 we see that NspCG exhibits a monotonically decreasing residual norm. The decrease is most rapid at the beginning of the iteration, and in spite of slowing convergence later, NspCG still significantly outperforms both QMR and GLSQR, which stagnates for the first 60 iterations before showing signs of convergence.

4.2. Example 2. Example 2 uses the ORSIR1 matrix from the Matrix Market collection, which represents a linear system used in oil reservoir modeling. This matrix can be obtained from [http://math.nist.gov/MatrixMarket/](http://math.nist.gov/MatrixMarket/).

We see that NspCG starts out with the lowest error in the 2-norm of the residual. Also we see that in both Figure 4.2 and Figure 4.1 that NspCG performs more consistently than either GLSQR or QMR. In the present example, although QMR eventually outperforms both GLSQR and NspCG, it takes over 800 iterations to do so. Unfortunately, unlike in the previous example, NspCG stagnates after just a few iterations. This is due to the fact that although the residuals are orthogonal with respect to the nonstandard inner product defined by \( \mathcal{M}(\gamma) \), they can become nearly parallel in the standard inner product.

Fig. 4.1. Example 1: NspCG, QMR and GLSQR for a sparse matrix of dimension 100
4.3. Example 3. First define the circulant matrix

\[ J = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & 0 & 1 \\ 1 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{p,p} \]

Now the matrix used in this example \( A = 10^{-3} \cdot \text{sprand}(n,n,0.2) + J \), where \( n = 100 \), can be constructed in MATLAB. In this example, both NspCG and GLSQR converge rapidly, while QMR fails to show any sign of convergence. While GLSQR has a growing storage requirement, which causes the iteration itself to take more time than that of NspCG, NspCG incurs significant overhead from the need to compute \( w \) and \( \gamma \), which requires estimating smallest and largest eigenvalues and singular values. This performance bottleneck will need to be addressed in future work.

4.4. Example 4. We need to first define

\[ D_1 = \begin{bmatrix} 1000 \\ \vdots \\ 1000 \end{bmatrix} \in \mathbb{R}^{p,p} \quad D_2 = \begin{bmatrix} 1 & 2 & \cdots \\ \vdots & \ddots & \ddots \\ q & \cdots \end{bmatrix} \in \mathbb{R}^{q,q} \]

where \( n = p + q \) and \( \Sigma = \text{diag}(D_1,D_2) \). Now we can define \( A = UV^T \), where \( U \) and \( V \) are orthogonal matrices. For this example we use \( n = 100 \) and \( D_1 \in \mathbb{R}^{90,90} \). From 4.4 we see that NspCG starts off with the best approximation, but only for about 15 iterations. Then it is overtaken by GLSQR. Also, we can see that QMR fails to converge at all.
Fig. 4.3. Example 3: NspCG, QMR, and GLSQR for a perturbed circulant shift matrix.

Fig. 4.4. Example 4: NspCG, QMR, and GLSQR with $D_1$ of dimension 90.

4.5. Example 5. This example uses the same definition of $D_1$, $D_2$, and $A$ from Example 4. In this example we will let $n = 100$ again, and $D_1 \in \mathbb{R}^{50,50}$. Figure 4.5 shows the same trend we have been seeing, that NspCG is more consistent at the beginning than any other method. Unfortunately, as in Example 2, NspCG stagnates, and at about 65 iterations GLSQR outperforms NspCG, while QMR fails to converge again.
4.6. Example 6. This example uses the same definition of $D_1$, $D_2$, and $A$ from Example 4. In this example we will let $n = 1000$ again, and $D_1 \in \mathbb{R}^{600,600}$. From

Figure 4.6 we see that NspCG shows the best results for the first 600 iterations, in spite of quickly stagnating. GLSQR takes many iterations to converge in this case, and QMR does not converge at all.
5. Preconditioning. According to [10] conjugate gradient has very rapid convergence for a symmetric positive definite matrix $A$ that is nearly identity. We need to apply preconditioning techniques to make our matrix $M$ satisfy this criterion. The result will be that the original system is transformed into an equivalent system where the coefficient matrix is near identity. As we have seen previously with conjugate gradient, preconditioning techniques can be generalized to the nonsymmetric case. The goal is to apply ILU preconditioning [23], while taking into account the structure of the nonsymmetric saddle point matrix $M$ defined in (3.1). The matrix $W$ in the (1,1) block is assumed to be a symmetric positive definite matrix; therefore it has a Cholesky factorization $W = GG^T$. We can use the QR factorization $G^TA = QR$ to obtain the factorization $M = LU$, where

$$L = \begin{bmatrix} R^T & 0 \\ -G^{-T}Q & G^{-T}Q \end{bmatrix}, \quad U = \begin{bmatrix} R & Q^TG^{-1} \\ 0 & Q^TG^{-1} \end{bmatrix}. \quad (5.1)$$

Let us define

$$C = G^TA \tilde{R}^{-1} = \tilde{Q},$$

where an incomplete QR factorization [20] is computed from the sparse matrix $G^TA$ which gives $G^TA \approx \tilde{Q}\tilde{R}$. By finding

$$\tilde{L}^{-1} = \begin{bmatrix} \tilde{R}^{-T} & 0 \\ \tilde{R}^{-T} & \tilde{Q}G^T \end{bmatrix}, \quad \tilde{U}^{-1} = \begin{bmatrix} \tilde{R}^{-1} & -\tilde{R}^{-1} \\ 0 & G\tilde{Q}^{-T} \end{bmatrix}, \quad (5.2)$$

it can be seen that the resulting preconditioned system matrix is given by

$$\tilde{L}^{-1}M\tilde{U}^{-1} = \begin{bmatrix} C^TC & -C^T\tilde{Q} \\ C^TC - \tilde{Q}G^T & -C^T\tilde{C} + C^T\tilde{Q} + \tilde{Q}G^TC \end{bmatrix}. \quad (5.3)$$

The above matrix has the structure similar to that of $M$ from (3.1), except it cannot readily be confirmed that the (2,2) block is positive semidefinite; therefore, although it is near $I$ to the extent that $\tilde{Q}\tilde{R}$ approximates $G^TA$, it is not necessarily a nonsymmetric saddle point matrix. Algorithm 3.1 can be modified to account for this preconditioner; the resulting algorithm is equivalent to one described in [26, Algorithm 3.2].

5.1. Example 1. We re-run Example 1 from the previous section with preconditioning. For NspCG, we use the approach described above; for all other methods featured in Figure 5.1, we use $ILU(0)$ preconditioning, and these methods are applied to the forward problem $Ax = b$. We see that NspCG converges in very few iterations, and unlike in Section 4, where preconditioning was not used, QMR performs well, followed by GMRES. As before, though, it should be noted that NspCG, while not needing many iterations, still incurs considerable overhead from the setup of the transformed system with $M$ and the inner product matrix $M(\gamma)$.

5.2. Example 2. We also re-run Example 2 from the previous section with preconditioning. In Figure 5.2 we see that NspCG again converges very rapidly, while GMRES, BiCGSTAB and QMR also converging within a reasonable number of iterations.
Fig. 5.1. Example 1 with preconditioning

Fig. 5.2. Example 2 with preconditioning

In this example, we find that the matrix of the preconditioned system, $\tilde{L}^{-1}M\tilde{U}^{-1}$, has eigenvalues that are clustered around 1, but are not all real. That is, this kind of preconditioning has not preserved the properties of $M$ that make a conjugate gradient-like iteration an attractive option; it seems that there is no reason not to use an alternative iterative method. We find that BiCGSTAB, applied to $M$ with the same preconditioner, does converge but not nearly as rapidly; GMRES, on the other
A. C. SUMNER AND J. V. LAMBERS

hand, converges almost as rapidly as NspCG, in terms of the number of iterations required.

6. Conclusions and Future Work. The results from this paper show that the NspCG method exhibits both promising and troubling behavior, compared to other iterative methods for nonsymmetric systems. NspCG only takes a few iterations to make fairly significant progress in reducing the residual norm, but then often stagnates due to near-parallelism of its residuals in the standard inner product. If preconditioning is used with NspCG, as is usually done with a conjugate gradient method, we have provided evidence that it will dramatically accelerate convergence, compared to state-of-the-art iterative methods such as GMRES or BiCG that are typically used to solve such systems. However, adjustments to our preconditioning approach are needed to ensure that the preconditioned system, like the original system with $M$, still shares essential properties with symmetric positive definite matrices and is therefore suitable for conjugate gradient-like iteration.

Future work will include relating the NspCG method to a quadrature rule, as in [7, 11], that can be used to compute the scattering amplitude without explicitly solving the forward or adjoint problem. This has been done in [8] with the symmetric matrix

$$C = \begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix}$$

in conjunction with block Lanczos iteration [9], but our goal is to achieve more rapid convergence.

Because the forward system $Ax = b$ is replaced with a system with twice as many unknowns and equations, it is essential to implement the iteration carefully so that the gain in convergence speed is not offset by the additional expense of each iteration. To that end, it is worthwhile to consider other choices for the matrix $W$ instead of just a multiple of identity.

For example, if we perform $ILU$ preconditioning on $A$ to obtain $A \approx LU$, then we could set $W = w(LL^T)^{-1}$, where the scalar $w$ is chosen to ensure a real positive spectrum. Then, a block diagonal preconditioner $P = HH^T$, where $H = \text{diag}(U^T, L)$, would yield a system in which each nonzero block would be an approximate multiple of identity.

Another option to explore is allowing a nonzero $(2, 2)$ block. That is, we use

$$M = \begin{bmatrix} A^TW_1A & A^T \\ -A & AW_2A^T \end{bmatrix}.$$ 

The theory presented in [15], and repeated in Section 3, can be applied to ensure a real positive spectrum. In particular, if $W_1$ and $W_2$ are chosen based on incomplete Cholesky factorizations of $AA^T$ and $A^TA$, respectively, then the diagonal blocks would be approximate multiples of identity. Based on an examination of the condition number of matrices of the form

$$\begin{bmatrix} w_1 & \sigma \\ -\sigma & w_2 \end{bmatrix},$$

for the cases $w_2 = 0, w_1 > 2\sigma$ and $w_1 - w_2 > 2\sigma$, it is our hope that such a choice of $M$ can be made better conditioned than one of the form (3.1).
REFERENCES


