# Orthogonalization Techniques for the Lanczos Tridiagonalization of Complex Symmetric Matrices

Sanzheng Qiao

Department of Computing and Software, McMaster University, Hamilton, Ontario Canada L8S 4K1

## ABSTRACT

We present three orthogonalization schemes for stablizing Lanczos tridiagonalization of a complex symmetric matrix.

**Keywords:** Complex symmetric matrix, Lanczos tridiagonalization, singular value decomposition (SVD), Takagi factorization.

## 1. INTRODUCTION

For any complex symmetric matrix $A$ of order $n$, there exist a unitary $Q \in C^{n\times n}$ and an order $n$ nonnegative diagonal $\Sigma = \mathrm{diag}(\sigma_1, ..., \sigma_n)$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$, such that

$$A = Q\Sigma Q^{\mathrm{T}} \quad \text{or} \quad Q^{\mathrm{H}} A \bar{Q} = \Sigma,$$

where $\bar{Q}$ is the complex conjugate of $Q$. This special form of singular value decomposition (SVD) is called Takagi factorization.[1,2]

The computation of the Takagi factorization consists of two stages: tridiagonalization and diagonalization.[3] A complex symmetric matrix is first reduced to complex symmetric and tridiagonal form. There are various tridiagonalization schemes. Householder transformations can be used.[3] Unfortunately, when $A$ is sparse or structured, Householder transformations destroy sparsity or structure. Alternatively, Lanczos method can be applied. Since Lanczos algorithm involves only matrix-vector multiplication, sparsity and structures can be exploited to develop fast tridiagonalization algorithms.[4]

The second stage, diagonalization of the complex symmetric tridiagonal matrix computed in the first stage, can be implemented by the implicit QR method.[3,4]

This paper presents three orthogonalization techniques for the Lanczos tridiagonalization of complex symmetric matrices. In Section 2, we describe the generic Lanczos tridiagonalization algorithm for complex symmetric matrices. Unfortunately, this method is unstable in floating-point arithmetic. A simple selective orthogonalization scheme and a practical partial orthogonalization scheme are proposed in Sections 3 and 4. Then, in Section 5, we present a modified partial orthogonalization scheme. Finally, Section 6 demonstrates our numerical experiments.

## 2. LANCZOS TRIDIAGONALIZATION

For an $n$-by-$n$ complex symmetric $A$, we can find a unitary $Q \in C^{n\times n}$ such that

$$T = Q^{\mathrm{H}} A \bar{Q} \tag{1}$$

is complex symmetric and tridiagonal. For example, $Q$ may consist of a sequence of Householder transformations.[3] Let

$$T = \begin{bmatrix} \alpha_1 & \beta_1 & & 0 \\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{n-1} \\ 0 & & \beta_{n-1} & \alpha_n \end{bmatrix} \tag{2}$$

Send correspondence to S. Qiao: E-Mail: qiao@mcmaster.ca

and rewrite (1) as

$$A\bar{Q} = QT. \tag{3}$$

Comparing the $j$th columns on the both sides of (3), we have

$$A\bar{\mathbf{q}}_j = \beta_{j-1}\mathbf{q}_{j-1} + \alpha_j\mathbf{q}_j + \beta_j\mathbf{q}_{j+1}, \quad \beta_0\mathbf{q}_0 = 0,$$

which leads to a Lanczos three-term recursion:

$$\beta_j\mathbf{q}_{j+1} = A\bar{\mathbf{q}}_j - \alpha_j\mathbf{q}_j - \beta_{j-1}\mathbf{q}_{j-1}. \tag{4}$$

The orthogonormality of $\mathbf{q}_j$ implies

$$\alpha_j = \mathbf{q}_j^{\mathrm{H}} A\bar{\mathbf{q}}_j.$$

Let $\mathbf{r}_j = A\bar{\mathbf{q}}_j - \alpha_j\mathbf{q}_j - \beta_{j-1}\mathbf{q}_{j-1}$, then $\beta_j = \pm\|\mathbf{r}_j\|_2$ and $\mathbf{q}_{j+1} = \mathbf{r}_j/\beta_j$ if $\mathbf{r}_j \neq 0$. Thus we have a generic Lanczos tridiagonalization algorithm for complex symmetric matrices.

Algorithm 1 (Lanczos Tridiagonalization). *Given a starting vector* $\mathbf{b}$ *and a subroutine for matrix-vector multiplication* $\mathbf{y} = A\mathbf{x}$ *for any* $\mathbf{x}$*, where* $A$ *is an* $n$*-by-*$n$ *complex symmetric matrix. This algorithm computes the diagonals of the complex symmetric and tridiagonal matrix* $T$ *in (2) and a unitary* $Q$ *such that* $T = Q^{\mathrm{H}}A\bar{Q}$*.*

```
q_0 = 0; β_0 = 0;
q_1 = b/||b||_2;
for j = 1 to n
    y = A q̄_j;
    α_j = q_j^H y;
    r_j = y − α_j q_j − β_{j−1} q_{j−1};
    β_j = ||r_j||_2;
    if β_j = 0, quit; end
    q_{j+1} = r_j/β_j;
end.
```

Since the major computation in Lanczos method is matrix-vector multiplication, fast tridiagonalization can be developed by exploiting the structure of $A$.[4] Unfortunately, in floating-point arithmetic, the above algorithm suffers from the loss of the orthogonality of the computed $Q$. To circumvent the problem, we may orthogonalize each $\mathbf{q}_j$ against all previous $\mathbf{q}_{j-1}, ..., \mathbf{q}_1$. This is called complete orthogonalization. For example, Householder matrices[5] or Gram-Schmidt scheme[6] can be used. Complete orthogonalization, however, is prohibitively expensive. In the following section, we propose a selective orthogonalization scheme.

## 3. SELECTIVE ORTHOGONALIZATION

Analogous to the Lanczos algorithms for symmetric eigenvalue problem,[6] in this section, we present a selective orthogonalization scheme for the Lanczos tridiagonalization of a complex symmetric matrix.

Before discussing the selective orthogonalization, we introduce some notations and definitions. During the $k$th iteration, $\alpha_k$, $\beta_k$, and $\mathbf{q}_{k+1}$ are computed. Denote

$$Q_k = [\mathbf{q}_1, ..., \mathbf{q}_k]$$

and

$$T_k = \begin{bmatrix} \alpha_1 & \beta_1 & & 0 \\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{k-1} \\ 0 & & \beta_{k-1} & \alpha_k \end{bmatrix}.$$

Suppose that

$$T_k = U\Sigma_k U^{\mathrm{T}} \tag{5}$$

is the Takagi factorization of $T_k$. We call the singular values or Takagi values on the diagonal of $\Sigma_k$ the Takagi-Ritz values and the columns of $Q_k U$ or their complex conjugates the Takagi-Ritz vectors. These values and vectors are approximations of the Takagi values (singular values) and Takagi vectors (left and right singular vectors) of $A$.

The basic idea behind the selective orthogonalization is to orthogonalize $\mathbf{q}_{k+1}$ against only a few selected Takagi-Ritz vectors, rather than all previously computed $\mathbf{q}_i$. What are the criteria for selecting the Takagi-Ritz vectors?

Similar to the case of real symmetric tridiagonalization problem considered by Paige.[7] we will show, for the case of complex symmetric tridiagonalization, that

$$\text{if} \quad |\beta_k u_k|/\|A\|_2 \leq \sqrt{\epsilon} \quad \text{then} \quad |\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}| \geq O(\sqrt{\epsilon}), \tag{6}$$

where $\mathbf{u}$ is a column of $U$ in (5) and $u_k$ is the $k$th or the last entry of $\mathbf{u}$ and $\epsilon$ is the unit of roundoff. A large $|\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}|$, which measures the orthogonality between $\mathbf{q}_{k+1}$ and a Takagi-Ritz vector $Q_k \mathbf{u}$ indicates that $\mathbf{q}_{k+1}$ has a large component in the direction of $Q_k \mathbf{u}$. We then orthogonalize $\mathbf{q}_{k+1}$ against $Q_k \mathbf{u}$.

Now, we prove the statement (6). Incorporating roundoff errors into the three-term recursion (4), we write

$$\beta_j \mathbf{q}_{j+1} + \mathbf{f}_j = A\bar{\mathbf{q}}_j - \alpha_j \mathbf{q}_j - \beta_{j-1}\mathbf{q}_{j-1}, \quad j = 1, .., k, \tag{7}$$

where $\mathbf{f}_j$ represents roundoff errors. In matrix form,

$$[0, ..., 0, \beta_k \mathbf{q}_{k+1}] + F_k = A\bar{Q}_k - Q_k T_k,$$

where $F_k = [\mathbf{f}_1, ..., \mathbf{f}_k]$, that is

$$A\bar{Q}_k = Q_k T_k + \beta_k \mathbf{q}_{k+1} \mathbf{e}_k^{\mathrm{T}} + F_k$$

where $\mathbf{e}_k = [0, ..., 0, 1]^{\mathrm{T}}$. Premultiplying the above equation with $Q_k^{\mathrm{H}}$, we get

$$Q_k^{\mathrm{H}} A\bar{Q}_k = Q_k^{\mathrm{H}} Q_k T_k + \beta_k Q_k^{\mathrm{H}} \mathbf{q}_{k+1} \mathbf{e}_k^{\mathrm{T}} + Q_k^{\mathrm{H}} F_k.$$

Since $Q_k^{\mathrm{H}} A\bar{Q}_k$ is symmetric,

$$(Q_k^{\mathrm{H}} Q_k T_k - T_k Q_k^{\mathrm{T}} \bar{Q}_k) + \beta_k (Q_k^{\mathrm{H}} \mathbf{q}_{k+1} \mathbf{e}_k^{\mathrm{T}} - \mathbf{e}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k) + (Q_k^{\mathrm{H}} F_k - F_k^{\mathrm{T}} \bar{Q}_k) = 0.$$

Let $Q_k^{\mathrm{H}} Q_k = I + C + C^{\mathrm{H}}$, where $C$ is the strictly lower triangular part of $Q_k^{\mathrm{H}} Q_k$. We assume that $\mathbf{q}_{j+1}$ is almost orthogonal to $\mathbf{q}_j$ for $j = 1, ..., k$, i.e., $\mathbf{q}_{j+1}^{\mathrm{H}} \mathbf{q}_j = O(\epsilon)$, then both the diagonal and subdiagonal of $C$ are zero. Also, $\mathbf{q}_{k+1}^{\mathrm{H}} \mathbf{q}_k = O(\epsilon)$ implies that the last entry of $\mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k$ is almost zero, which means that $\mathbf{e}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k$ is also strictly lower triangular. Thus $CT_k - T_k \bar{C}$ is the strictly lower triangular part of $Q_k^{\mathrm{H}} Q_k T_k - T_k Q_k^{\mathrm{T}} \bar{Q}_k$ and $\beta_k \mathbf{e}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k$ is the strictly lower triangular part of $\beta_k (Q_k^{\mathrm{H}} \mathbf{q}_{k+1} \mathbf{e}_k^{\mathrm{T}} - \mathbf{e}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k)$. Denoting $L$ as the strictly lower triangular part of $Q_k^{\mathrm{H}} F_k - F_k^{\mathrm{T}} \bar{Q}_k$, we get

$$(CT_k - T_k \bar{C}) - \beta_k \mathbf{e}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k + L = 0. \tag{8}$$

Let $(\sigma, \mathbf{u})$ be a Takagi pair of $T_k$, then premultiplying and postmultiplying (8) with $\mathbf{u}^{\mathrm{H}}$ and $\bar{\mathbf{u}}$ respectively, we have

$$\sigma(\mathbf{u}^{\mathrm{H}} C\mathbf{u} - \mathbf{u}^{\mathrm{T}} \bar{C}\bar{\mathbf{u}}) - \beta_k \bar{u}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k \bar{\mathbf{u}} + \mathbf{u}^{\mathrm{H}} L\bar{\mathbf{u}} = 0.$$

Consider the real part. Since $\mathrm{Real}(\mathbf{u}^{\mathrm{H}} C\mathbf{u} - \mathbf{u}^{\mathrm{T}} \bar{C}\bar{\mathbf{u}}) = 0$,

$$|\mathrm{Real}(\beta_k \bar{u}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k \bar{\mathbf{u}})| = |\mathrm{Real}(\mathbf{u}^{\mathrm{H}} L\bar{\mathbf{u}})|.$$

The right side

$$|\mathrm{Real}(\mathbf{u}^{\mathrm{H}} L\bar{\mathbf{u}})| \leq |\mathbf{u}^{\mathrm{H}} L\bar{\mathbf{u}}| \leq \|L\|_2 = O(\|F\|_2) = O(\epsilon \, \|A\|_2).$$

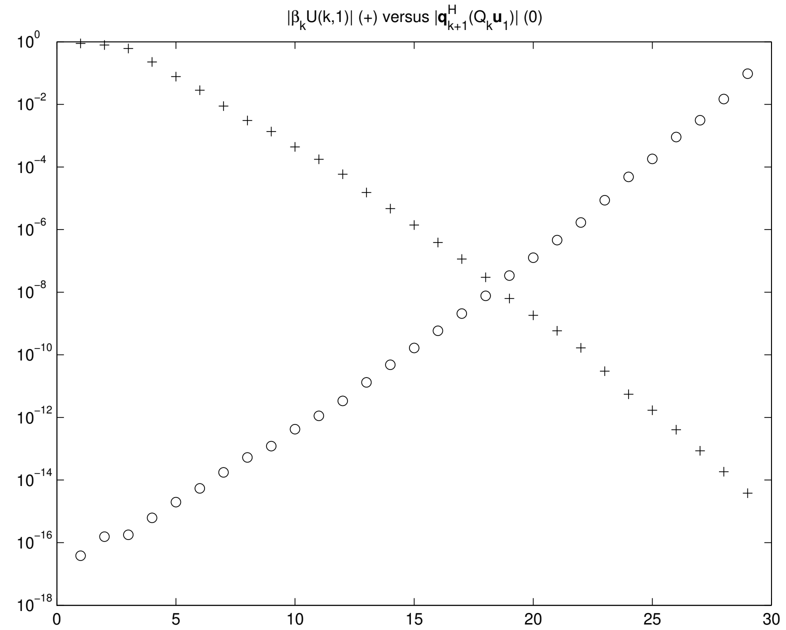


**Figure 1.** Algorithm 1 is applied to a 40-by-40 complex symmetric matrix. Column $k$ plots $|\beta_k u_k|$ (a "+") and $|\mathbf{q}_{k+1}^{\mathrm{H}}(Q_k \mathbf{u}_1)|$ (an "o") computed in the $k$th iteration, $k = 1, ..., 30$.

The left side

$$\begin{aligned} & |\mathrm{Real}(\beta_k \bar{u}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k \bar{\mathbf{u}})| \\ = \quad & |\beta_k (\mathrm{Real}(u_k)\,\mathrm{Real}(\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}) - \mathrm{Im}(u_k)\,\mathrm{Im}(\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}))|. \end{aligned}$$

Thus

$$|\beta_k (\mathrm{Real}(u_k)\,\mathrm{Real}(\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}) - \mathrm{Im}(u_k)\,\mathrm{Im}(\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}))| = O(\epsilon\,\|A\|_2).$$

If $|\beta_k u_k|/\|A\|_2 \leq \sqrt{\epsilon}$, then $|\beta_k \mathrm{Real}(u_k)|/\|A\|_2 \leq \sqrt{\epsilon}$ and $|\beta_k \mathrm{Im}(u_k)|/\|A\|_2 \leq \sqrt{\epsilon}$. Consequently,

$$\begin{aligned} O(\epsilon\,\|A\|_2) &= |\beta_k \mathrm{Real}(\bar{u}_k \mathbf{q}_{k+1}^{\mathrm{T}} \bar{Q}_k \bar{\mathbf{u}})| \\ &\leq \sqrt{\epsilon}\,\|A\|_2 (|\mathrm{Real}(\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u})| + |\mathrm{Im}(\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u})|) \\ &\approx \sqrt{\epsilon}\,\|A\|_2 |\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}|, \end{aligned}$$

which implies that $|\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}| \geq O(\sqrt{\epsilon})$.

The relation between $|\beta_k u_k|$ and $|\mathbf{q}_{k+1}^{\mathrm{H}} Q_k \mathbf{u}|$ is depicted in Figure 1. We ran 30 iterations in Algorithm 1 (without orthogonalization) for a 40-by-40 complex symmetric matrix. For each iteration $k$, $k = 1, ..., 30$, we computed $|\mathbf{q}_{k+1}^{\mathrm{H}}(Q_k \mathbf{u}_1)|$, where $\mathbf{u}_1$ is the first column of $U$, and $|\beta_k u_k|$, where $u_k$ is the last entry of $\mathbf{u}_1$. In Figure 1, a "+" in the $k$th column is $|\beta_k u_k|$ and an "o" is $|\mathbf{q}_{k+1}^{\mathrm{H}}(Q_k \mathbf{u}_1)|$. The figure shows that these two values are related approximately by

$$|\beta_k u_k| = \frac{O(\epsilon)}{|\mathbf{q}_{k+1}^{\mathrm{H}}(Q_k \mathbf{u}_1)|},$$

where $\epsilon$ is the double precision, approximately $10^{-16}$.

Finally, we present the following Lanczos algorithm with a simple selective orthogonalization scheme. We use the largest singular value $\sigma_1$ of $T_k$ as an approximation of $\|A\|_2$ and Gram-Schmidt method for orthogonalization.

Algorithm 2 (Selective Orthogonalization). *Given a starting vector* $\mathbf{b}$ *and a subroutine for matrix-vector multiplication* $\mathbf{y} = A\mathbf{x}$ *for any* $\mathbf{x}$, *where* $A$ *is an* $n$*-by-*$n$ *complex symmetric matrix. This algorithm* *computes the diagonals of the complex symmetric and tridiagonal matrix $T$ in (2) and a unitary $Q$ such that $T = Q^{\mathrm{H}} A \bar{Q}$.*

$\mathbf{q}_0 = 0$; $\beta_0 = 0$;
$\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|_2$;
for $j = 1$ to $n$
  $\mathbf{y} = A\bar{\mathbf{q}}_j$;
  $\alpha_j = \mathbf{q}_j^{\mathrm{H}}\mathbf{y}$;
  $\mathbf{r}_j = \mathbf{y} - \alpha_j \mathbf{q}_j - \beta_{j-1}\mathbf{q}_{j-1}$;
  $\beta_j = \|\mathbf{r}_j\|_2$;
  Compute Takagi factorization $T_j = U\Sigma U^{\mathrm{T}}$;
  for $k = 1$ to $j$
    if $|\beta_j U(j,k)| \leq \sigma_1\sqrt{\epsilon}$
      $\mathbf{v} = Q_j \mathbf{u}_k$;
      $\mathbf{r}_j = \mathbf{r}_j - (\mathbf{v}^{\mathrm{H}}\mathbf{r}_j)\mathbf{v}$;
    end
  end
  $\beta_j = \|\mathbf{r}_j\|_2$;
  if $\beta_j = 0$, quit; end
  $\mathbf{q}_{j+1} = \mathbf{r}_j/\beta_j$;
end.

This algorithm can be useful in computing the approximations of a few largest Takagi values and their corresponding Takagi vectors of a large complex symmetric matrix. Figure 2 shows the approximations of the Takagi values. We generated a 40-by-40 complex symmetric matrix, ran 19 iterations of Algorithm 2. The $j$th, $j = 1, ..., 19$, column in Figure 2 plots the $j$ Takagi-Ritz values computed in the $j$th iteration. The 21st column shows the Takagi values or the singular values of $A$. This example shows that the Takagi-Ritz values quickly converge to the Takagi values or the singular values of $A$, especially the large ones.

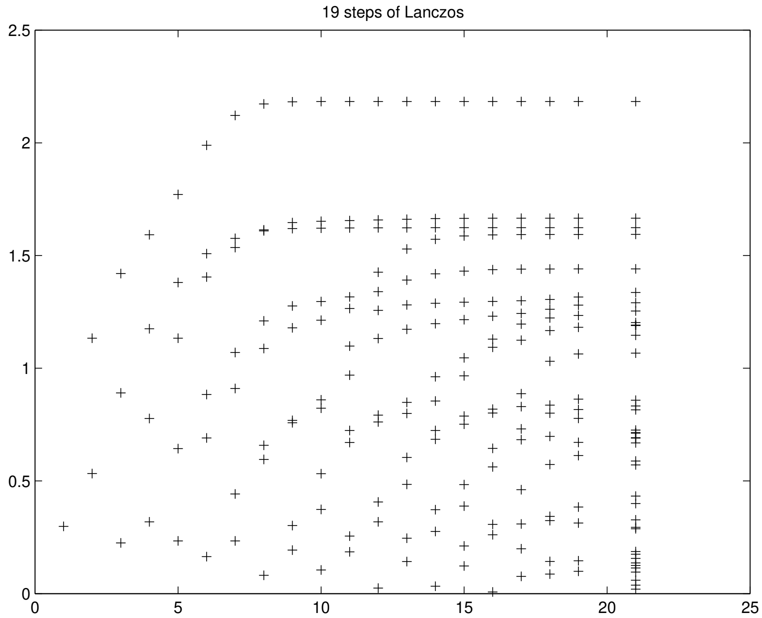


**Figure 2.** Takagi-Ritz values of a 40-by-40 complex symmetric matrix $A$. Column $j$ shows the $j$ Takagi-Ritz values computed in the $j$th iteration of Algorithm 2. The 21st column shows the 40 Takagi values of $A$.

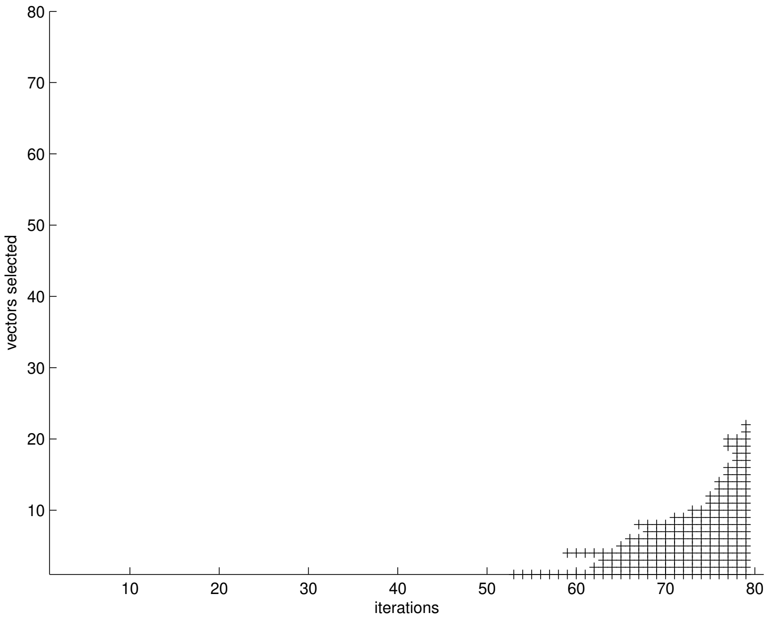


**Figure 3.** Vectors selected for orthogonalization in the selective orthogonalization scheme. Column $j$ plots the vectors selected in iteration $j$.

This selective orthogonalization scheme has two drawbacks. First, it requires the Takagi factorization of $T_j$ for each iteration. Second, it orthogonalizes $\mathbf{q}_{j+1}$ against only selected Takagi-Ritz vectors. What is wrong with the selective orthogonalization? Suppose that $\mathbf{q}_k^{\mathrm{H}}\mathbf{q}_{j+1}$ has exceeded the threshold, usually some neighboring $\mathbf{q}_i^{\mathrm{H}}\mathbf{q}_{j+1}$ have grown to about the threshold.[8] If we reorthogonalize $\mathbf{q}_{j+1}$ only against $\mathbf{q}_k$, then its effect will be wiped out immediately by the neighboring terms. Figure 3 illustrates the second drawback. Each "+" in the figure represents a vector selected for reorthogonalization. For example, in iteration 62, vectors $Q_{62}\mathbf{u}_1$, $Q_{62}\mathbf{u}_2$, and $Q_{62}\mathbf{u}_4$ are selected for reorthogonalization. While $Q_{62}\mathbf{u}_2$ is selected, its neighbor $Q_{62}\mathbf{u}_3$ is not selected in iteration 62. However, $Q_{63}\mathbf{u}_3$ is selected in the subsequent iteration 63. Note that both $Q_{62}\mathbf{u_3}$ and $Q_{63}\mathbf{u}_3$ are approximations of a same Takagi vector. This means that although $\mathbf{q}_3^{\mathrm{H}}Q_{62}\mathbf{u}_3$ does not exceed the threshold, it has grown to about the threshold in iteration 62. The fact that $Q_{62}\mathbf{u}_2$ and $Q_{63}\mathbf{u}_2$ are selected in two consecutive iterations indicates that the effect of the orthogonalization against $Q_{62}\mathbf{u}_2$, which is an approximation of $\mathbf{q}_2$, performed in iteration 62 is immediately wiped out by its neighboring $Q_{63}\mathbf{u}_3$. This effect becomes more dramatic as iteration continues. In the next section, we apply the partial reorthogonalization[8] to complex symmetric case to overcome these two drawbacks.

## 4. PARTIAL ORTHOGONALIZATION

To avoid the calculation of the Takagi-Ritz vectors and values, we check the orthogonalities $\mathbf{q}_k^{\mathrm{H}}\mathbf{q}_{j+1}$ of Takagi vectors, instead of Takagi-Ritz vectors. In this section, we first establish a recursion on the estimates for the orthogonalities of Takagi vectors. This recursion provides an efficient way of monitoring the orthogonality. Based on the recursion, we propose a reorthogonalization algorithm.

From (7), we have

$$\begin{aligned} \beta_j\mathbf{q}_{j+1} &= A\bar{\mathbf{q}}_j - \alpha_j\mathbf{q}_j - \beta_{j-1}\mathbf{q}_{j-1} - \mathbf{f}_j \\ \beta_k\mathbf{q}_{k+1} &= A\bar{\mathbf{q}}_k - \alpha_k\mathbf{q}_k - \beta_{k-1}\mathbf{q}_{k-1} - \mathbf{f}_k. \end{aligned}$$

Premultiplying the above two equations with $\mathbf{q}_k^{\mathrm{H}}$ and $\mathbf{q}_j^{\mathrm{H}}$ respectively and denoting $\omega_{k,j} = \mathbf{q}_k^{\mathrm{H}}\mathbf{q}_j$, we get

$$\beta_j\omega_{k,j+1} = \mathbf{q}_k^{\mathrm{H}}A\bar{\mathbf{q}}_j - \alpha_j\omega_{k,j} - \beta_{j-1}\omega_{k,j-1} - \mathbf{q}_k^{\mathrm{H}}\mathbf{f}_j$$

$$\beta_k \omega_{j,k+1} = \mathbf{q}_j^{\mathrm{H}} A \bar{\mathbf{q}}_k - \alpha_k \omega_{j,k} - \beta_{k-1}\omega_{j,k-1} - \mathbf{q}_j^{\mathrm{H}} \mathbf{f}_k.$$

Since $A$ is symmetric, $\mathbf{q}_k^{\mathrm{H}} A \bar{\mathbf{q}}_j = \mathbf{q}_j^{\mathrm{H}} A \bar{\mathbf{q}}_k$. Thus, subtracting the above two equations and noting that $\omega_{k,j} = \bar{\omega}_{j,k}$, we have the following recursion on the orthogonalities of the Takagi vectors:

$$\beta_j \omega_{k,j+1} = \beta_k \bar{\omega}_{k+1,j} + \alpha_k \bar{\omega}_{k,j} - \alpha_j \omega_{k,j} + \beta_{k-1}\bar{\omega}_{k-1,j} - \beta_{j-1}\omega_{k,j-1} + \mathbf{q}_j^{\mathrm{H}}\mathbf{f}_k - \mathbf{q}_k^{\mathrm{H}}\mathbf{f}_j. \tag{9}$$

The above equation shows that we need $\omega_{k-1,j}$, $\omega_{k,j}$ and $\omega_{k+1,j}$ computed in iteration $j$ and $\omega_{k,j-1}$ in iteration $j-1$ to calculate $\omega_{k,j+1}$. Obviously, we define $\omega_{0,j} = 0$ and $\omega_{j,j} = 1$ for all $j$. Thus, we also define

$$\psi_j := \omega_{j,j+1} \tag{10}$$

as a random variable whose value will be discussed later. The problem is that the round-off error term $\mathbf{q}_j^{\mathrm{H}}\mathbf{f}_k - \mathbf{q}_k^{\mathrm{H}}\mathbf{f}_j$ in (9) is unknown. Again, we define

$$\theta_{k,j} := \mathbf{q}_j^{\mathrm{H}}\mathbf{f}_k - \mathbf{q}_k^{\mathrm{H}}\mathbf{f}_j, \tag{11}$$

as a random variable whose value will be discussed soon. Using these notations, we get

$$\omega_{k,j+1} = \beta_j^{-1}(\beta_k \bar{\omega}_{k+1,j} + \alpha_k \bar{\omega}_{k,j} - \alpha_j \omega_{k,j} + \beta_{k-1}\bar{\omega}_{k-1,j} - \beta_{j-1}\omega_{k,j-1}) + \theta_{k,j}, \tag{12}$$

for $k = 1, ..., j-1$, with

$$\beta_0 = \omega_{0,j} = 0, \quad \omega_{j,j+1} = \psi_j \quad \text{and} \quad \omega_{j+1,j+1} = 1.0$$

How do we choose the values for $\psi_j$ in (10) and $\theta_{k,j}$ in (11)? Based on the statistical study by Simon,[8] let $\epsilon$ be the roundoff unit, we propose that

$$\psi_j = n\epsilon \frac{\beta_1}{\beta_j}(\Psi_{\mathrm{r}} + i\Psi_{\mathrm{i}}), \quad \Psi_{\mathrm{r}},\ \Psi_{\mathrm{i}} \in N(0, 0.6), \tag{13}$$

where $N(0, v)$ means normal distribution with zero mean and variance $v$, and

$$\theta_{k,j} = \epsilon(\beta_k + \beta_j)(\Theta_{\mathrm{r}} + i\Theta_{\mathrm{i}}), \quad \Theta_{\mathrm{r}},\ \Theta_{\mathrm{i}} \in N(0, 0.6). \tag{14}$$

To aleviate the problem caused by isolated reorthogonalization, when $\omega_{k,j+1}$ exceeds the threshold $\sqrt{\epsilon}$ for some $k$, we reorthogonalize $\mathbf{q}_{j+1}$ against all the previous Takagi vectors $\mathbf{q}_k$, $k = 1, ..., j$. Moreover, we always perform a reorthogonalization in the subsequent iteration. Theoretically, after the reorthogonalization, $\omega_{k,j+1} = 0$, $k = 1, ..., j$. To incorporate the rounding errors, we set

$$\omega_{k,j+1} = \epsilon(\Omega_{\mathrm{r}} + i\Omega_{\mathrm{i}}), \quad \Omega_{\mathrm{r}},\ \Omega_{\mathrm{i}} \in N(0, 1.5). \tag{15}$$

This completes the algorithm for computing the estimates $\omega_{k,j+1}$ for $\mathbf{q}_k^{\mathrm{H}}\mathbf{q}_{j+1}$, for $k = 1, ..., j$. Finally, we have the following algorithm for the partial reorthogonalization.

ALGORITHM 3 (PARTIAL ORTHOGONALIZATION). *Given a starting vector* $\mathbf{b}$ *and a subroutine for matrix-vector multiplication* $\mathbf{y} = A\mathbf{x}$ *for any* $\mathbf{x}$, *where* $A$ *is an* $n$*-by-*$n$ *complex symmetric matrix. Using partial orthogonalization, this algorithm computes the diagonals of the complex symmetric and tridiagonal matrix* $T$ *in (2) and a unitary* $Q$ *such that* $T = Q^{\mathrm{H}} A \bar{Q}$.

$\mathbf{q}_0 = 0$; $\beta_0 = 0$; $\omega_{1,1} = 1$;
$\mathbf{q}_1 = \mathbf{b}/\|\mathbf{b}\|_2$;
for $j = 1$ to $n$
  $\mathbf{y} = A\bar{\mathbf{q}}_j$;
  $\alpha_j = \mathbf{q}_j^{\mathrm{H}}\mathbf{y}$;
  $\mathbf{r}_j = \mathbf{y} - \alpha_j \mathbf{q}_j - \beta_{j-1}\mathbf{q}_{j-1}$;
  $\beta_j = \|\mathbf{r}_j\|_2$;
  Compute $\omega_{k,j+1}$ for $k = 1, ..., j-1$ using (12);
  Set $\omega_{j,j+1}$ to $\psi_j$ using (13);

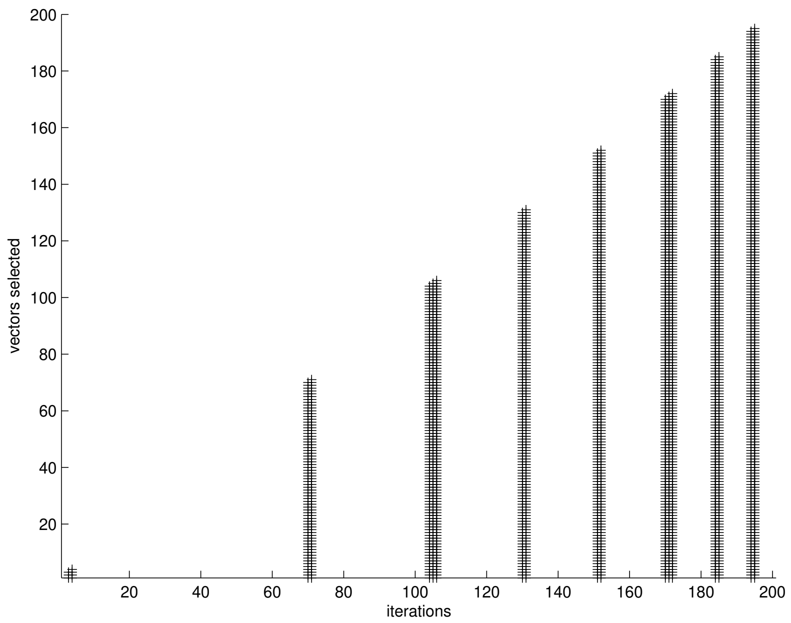


**Figure 4.** Vectors selected for orthogonalization in partial orthogonalization scheme. Column $j$ plots the vectors selected in iteration $j$.

Set $\omega_{j+1,j+1} = 1$;
if $\max_{1\le k\le j}(|\omega_{k,j+1}|) > \sqrt{\epsilon}$
Orthogonalize $\mathbf{r}_j$ against $\mathbf{q}_1, ..., \mathbf{q}_j$;
Perform orthogonalization in the next iteration;
Reset $\omega_{k,j+1}$ using (15);
Recalculate $\beta_j = \|\mathbf{r}_j\|_2$;
end
if $\beta_j = 0$, quit; end
$\mathbf{q}_{j+1} = \mathbf{r}_j/\beta_j$;
end.

Figure 4 depicts the vectors selected for reorthogonalization in partial orthogonalization scheme. As shown in the figure, after orthogonalization is performed in two consecutive iterations, good orthogonality stays for a few iterations.

## 5. MODIFIED PARTIAL ORTHOGONALIZATION

In the partial orthogonalization scheme Algorithm 3, $\mathbf{r}_j$ is orthogonalized against all $\mathbf{q}_1, ..., \mathbf{q}_j$ when $|\omega_{k,j+1}| > \sqrt{\epsilon}$ for some $k$. The idea behind the modified partial orthogonalization is to orthogonalize $\mathbf{r}_j$ against only $\mathbf{q}_k$ and its neighboring vectors when $|\omega_{k,j+1}| > \sqrt{\epsilon}$. How do we define the neighborhood of $k$? The neighborhood should include all the vectors $\mathbf{q}_i$ neighboring $\mathbf{q}_k$ for which $|\omega_{i,j+1}|$ have grown to about $\sqrt{\epsilon}$. Thus, for each $k$, we define the neighborhood with the lower bound $l_k \le k$, the smallest integer such that $|\omega_{i,j+1}| \ge tol$ for all $i$ between $l_k$ and $k$, and the upper bound $u_k \ge k$, the largest integer such that $|\omega_{i,j+1}| \ge tol$ for all $i$ between $k$ and $u_k$. In other words, $[l_k, u_k]$ is the largest interval such that $k \in [l_k, u_k]$ and $|\omega_{i,j+1}| \ge tol$ for all $i$ between $l_k$ and $u_k$. The tolerance $tol$ should be some value between $\epsilon$ and $\sqrt{\epsilon}$. We choose $tol = \epsilon^{3/4}$.

As in the partial orthogonalization Algorithm 3, whenever we perform orthogonalization in iteration $j$, we carry out orthogonalization in the subsequent iteration $j+1$. As shown in (9), since the computation of $\omega_{k,j+1}$ requires $\omega_{k-1,j}$, $\omega_{k,j}$, and $\omega_{k+1,j}$, we expand each interval $[l_k, u_k]$ to $[l_k - 1, u_k + 1]$ for the subsequent iteration.

The following algorithm shows the modified partial orthogonalization scheme.

ALGORITHM 4 (MODIFIED PARTIAL ORTHOGONALIZATION). *Given a starting vector* $\mathbf{b}$ *and a subroutine for matrix-vector multiplication* $\mathbf{y} = A\mathbf{x}$ *for any* $\mathbf{x}$*, where* $A$ *is an* $n$*-by-*$n$ *complex symmetric matrix. Using the modified partial orthogonalization, this algorithm computes the diagonals of the complex symmetric and tridiagonal matrix* $T$ *in (2) and a unitary* $Q$ *such that* $T = Q^{\mathrm{H}} A \bar{Q}$.

```
q_0 = 0; β_0 = 0; ω_{1,1} = 1;
q_1 = b/||b||_2;
for j = 1 to n
    y = A q̄_j;
    α_j = q_j^H y;
    r_j = y − α_j q_j − β_{j−1} q_{j−1};
    β_j = ||r_j||_2;
    Compute ω_{k,j+1} for k = 1,...,j − 1 using (12);
    Set ω_{j,j+1} to ψ_j using (13);
    Set ω_{j+1,j+1} = 1;
    k = 1;
    while k ≤ j
        if |ω_{k,j+1}| > √ε
            Find the neighborhood [l_k, u_k] of k;
            k = u_k + 1;
        else
            k = k+1;
        end
    end
    for each interval [l_k, u_k]
        Orthogonalize r_j against q_{l_k},...,q_{u_k};
        Reset ω_{l_k,j+1},...,ω_{u_k,j+1} using (15);
        Adjust the neighborhood to [l_k − 1, u_k + 1] for the next iteration;
    end
    if orthogonalization was performed
        Recalculate β_j = ||r_j||_2;
    end
    if β_j = 0, quit; end
    q_{j+1} = r_j/β_j;
end.
```

Figure 5 shows that not all previous vectors are selected in some iterations in the modified partial orthogonalization procedure.

## 6. EXPERIMENTS

Algorithms 1, 2, 3, and 4 were programmed in MATLAB. The random complex symmetric matrices in the following examples were generated as follows. First, a set of $n$ random numbers with a normal Gaussian distribution with zero mean and variance 1 was generated. Their absolute values were chosen as the singular values $\sigma_1, ..., \sigma_n$. Then, a random unitary matrix $\widehat{Q}$ of order $n$ was generated to form a complex symmetric matrix $A = \widehat{Q}\Sigma\widehat{Q}^{\mathrm{T}}$. All starting vectors $\mathbf{b}$ were $[1, ..., 1]^{\mathrm{T}}$.

**Example 1**. Various sizes of random complex symmetric matrices were generated and ran on Algorithm 2 (with selective orthogonalization). The orthogonality of $Q$ was measured by $\|I - Q^{\mathrm{H}}Q\|_{\mathrm{F}}/n^2$ and the error in the tridiagonalization was measured by $\|Q^{\mathrm{H}}A\bar{Q} - T\|_{\mathrm{F}}/n^2$. Table 1 shows the results. For comparison, without orthogonalization, typically, $Q$ completely loses orthogonality around size $n = 20$.

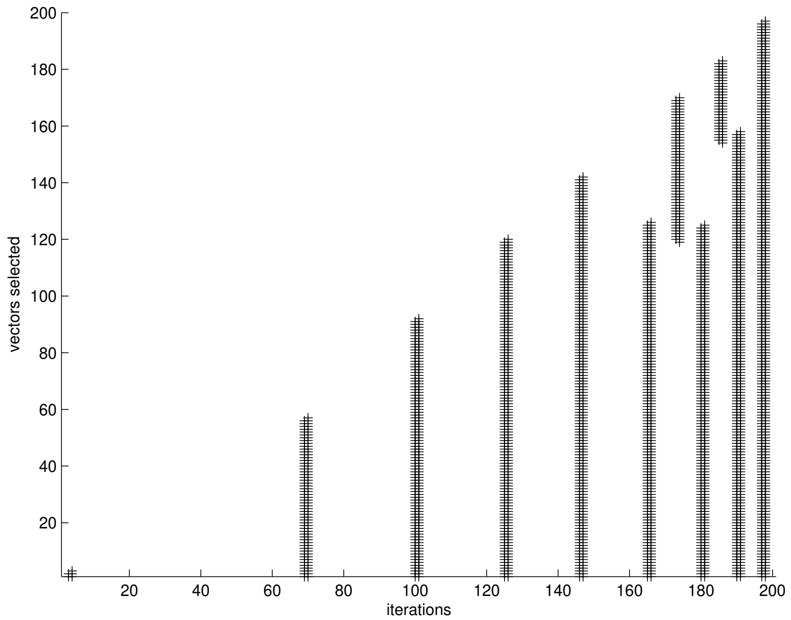


**Figure 5.** Vectors selected for orthogonalization in the modified partial orthogonalization scheme. Column $j$ plots the vectors selected in iteration $j$.

| size $n$ | number of vectors selected for orthogonalization | orthogonality $\|I - Q^{\rm H}Q\|_{\rm F}/n^2$ | factorization $\|Q^{\rm H}A\bar{Q} - T\|_{\rm F}/n^2$ |
|---|---|---|---|
| 20 | 11 | $2.02E-11$ | $1.29E-10$ |
| 40 | 73 | $1.16E-10$ | $2.60E-09$ |
| 100 | 1122 | $1.86E-11$ | $2.35E-09$ |
| 200 | 6162 | $1.07E-08$ | $1.15E-06$ |

**Table 1.** Efficiency and accuracy of Algorithm 2.

| size $n$ | number of vectors selected for orthogonalization | orthogonality $\|I - Q^{\mathrm{H}}Q\|_{\mathrm{F}}/n^2$ | factorization $\|Q^{\mathrm{H}}A\bar{Q} - T\|_{\mathrm{F}}/n^2$ |
|---|---|---|---|
| 20 | 7 | $1.24E-15$ | $8.01E-16$ |
| 40 | 106 | $2.75E-12$ | $2.74E-12$ |
| 100 | 567 | $7.43E-13$ | $7.30E-13$ |
| 200 | 2168 | $4.54E-13$ | $4.44E-13$ |
| 500 | 14735 | $2.04E-13$ | $2.01E-13$ |
| 1000 | 51579 | $8.21E-14$ | $8.12E-14$ |
| 2000 | 211373 | $4.03E-14$ | $3.96E-14$ |

**Table 2.** Efficiency and accuracy of Algorithm 3.

| size $n$ | number of vectors selected for orthogonalization | orthogonality $\|I - Q^{\mathrm{H}}Q\|_{\mathrm{F}}/n^2$ | factorization $\|Q^{\mathrm{H}}A\bar{Q} - T\|_{\mathrm{F}}/n^2$ |
|---|---|---|---|
| 20 | 5 | $3.16E-15$ | $2.78E-15$ |
| 40 | 98 | $1.21E-12$ | $1.20E-12$ |
| 100 | 408 | $8.88E-13$ | $8.78E-13$ |
| 200 | 2014 | $4.54E-13$ | $4.46E-13$ |
| 500 | 13771 | $1.98E-13$ | $1.86E-13$ |
| 1000 | 47676 | $9.44E-14$ | $8.66E-14$ |
| 2000 | 197630 | $5.96E-14$ | $5.38E-14$ |

**Table 3.** Efficiency and accuracy of Algorithm 4.

**Example 2**. In comparison with the selective orthogonalization in Example 1, various sizes of random complex symmetric matrices were generated and ran on Algorithm 3 (with partial orthogonalization). The measurements of the orthogonality of $Q$ and the error in the tridiagonalization are the same as those in Example 1. Table 2 shows the results.

**Example 3**. In comparison with the partial orthogonalization in Example 2, various sizes of random complex symmetric matrices were generated and ran on Algorithm 4 (with modified partial orthogonalization). The measurements of the orthogonality of $Q$ and the error in the tridiagonalization are the same as those in Example 1. Table 3 shows that the modified partial orthogonalization is more efficient than the partial orthogonalization.

**Example 4**. Algorithm 2 was applied to a random 1000-by-1000 complex symmetric matrix $A$ for 30 iterations. The four largest Takagi-Ritz values and their corresponding vectors were computed as approximations of the Takagi values and vectors of $A$. Table 4 shows the accuracy of the approximations. A small $1 - |\mathbf{q}_i^{\mathrm{H}}\hat{\mathbf{q}}_i|$ indicates that $\mathbf{q}_i$ and $\hat{\mathbf{q}}_i$ are either in the same direction or in the opposite directions. This example shows that Algorithm 2 is very effective in computing approximations of the largest singular values and vectors of complex symmetric matrices.

**Conclusion**. In this paper, we have presented a simple selective orthogonalization scheme and practical partial orthogonalization and modified partial orthogonalization schemes for the Lanczos tridiagonalization of a complex

| $i$ | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| $\|\hat{\sigma}_i - \sigma_i\|$ | $1.90E-8$ | $1.71E-6$ | $7.27E-4$ | $7.40E-3$ |
| $1 - \|\mathbf{q}_i^{\mathrm{H}}\hat{\mathbf{q}}_i\|$ | $1.59E-8$ | $3.33E-6$ | $1.10E-3$ | $1.11E-1$ |

**Table 4.** Errors in the largest four computed Takagi-Ritz values $\hat{\sigma}_i$ and Takagi-Ritz vectors $\hat{\mathbf{q}}_i$ against the exact Takagi values $\sigma_i$ and vectors $\mathbf{q}_i$ after 30 iterations of Algorithm 2 on a 1000-by-1000 complex symmetric matrix.

symmetric matrix. Experimental results show that all the three orthogonalization schemes effectively stablize the Lanczos tridiagonalization, whereas, partial orthogonalization and modified partial orthogonalization schemes are efficient. However, the selective orthogonalization is useful in computing the approximations of a few largest singular values and their corresponding singular vectors of a large complex symmetric matrix.

## ACKNOWLEDGMENTS

This work is partially supported by Natural Sciences and Engineering Research Council of Canada.

## REFERENCES

1. R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.
2. T. Takagi, "On an algebraic problem related to an analytic theorem of carathédory and fejér and on an allied theorem of landau," *Japan J. Math.* **1**, pp. 82–93, 1924.
3. A. Bunse-Gerstner and W. B. Gragg, "Singular value decompositions of complex symmetric matrices," *Journal of Computational and Applied Mathematics* **21**, pp. 41–54, 1988.
4. F. T. Luk and S. Qiao, "A fast singular value algorithm for hankel matrices," in *Fast Algorithms for Structured Matrices: Theory and Applications, Contemporary Mathematics 323*, V. Olshevsky, ed., pp. 169–177, American Mathematical Society, 2003.
5. G. H. Golub and C. F. V. Loan, *Matrix Computations, 3rd Ed.*, The Johns Hopkins University Press, Baltimore, MD, 1996.
6. J. W. Demmel, *Applied Numerical Linear Algebra*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.
7. C. C. Paige, "Error analysis of the lanczos algorithm for tridiagonalizing a symmetric matrix," *J. Inst. Maths Applics.* **18**, pp. 341–349, 1976.
8. H. D. Simon, "The lanczos algorithm with partial reorthogonalization," *Mathematics of Computation* **42**, pp. 115–142, 1984.