

A DIVIDE-AND-CONQUER METHOD FOR THE TAKAGI FACTORIZATION*

WEI XU[†] AND SANZHENG QIAO[†]

Abstract. This paper presents a divide-and-conquer method for computing the symmetric singular value decomposition, or Takagi factorization, of a complex symmetric and tridiagonal matrix. An analysis of accuracy shows that our method produces accurate Takagi values and orthogonal Takagi vectors. Our preliminary numerical experiments have confirmed our analysis and demonstrated that our divide-and-conquer method is much more efficient than the implicit QR method even for moderately large matrices.

Key words. divide-and-conquer method, symmetric SVD, Takagi factorization

AMS subject classifications. 15A18, 65F20, 65F25, 65F50

DOI. 10.1137/050624558

1. Introduction. The Takagi factorization of a complex symmetric matrix A can be written as [7]

$$A = V\Sigma V^T,$$

where V is a unitary matrix, V^T is the transpose of V , and Σ is a nonnegative diagonal matrix. The columns of V are called the Takagi vectors of A and the diagonal elements of Σ are its Takagi values. Since $V^T = \bar{V}^H$, where \bar{V} denotes the complex conjugate of V , the Takagi factorization is a symmetric form of the singular value decomposition (SVD); but there are differences. A pair of left-right singular vectors are unique up to a complex scaling factor with unit modulus, while the Takagi vectors are unique up to a sign change. Therefore, if \mathbf{v}_i is a Takagi vector, then $(\mathbf{v}_i, \bar{\mathbf{v}}_i)$ is a pair of left-right singular vectors, but a left singular vector is not necessarily a Takagi vector; see an example below.

Similar to the computation of the SVD, a standard algorithm for computing the Takagi factorization consists of two stages. The first stage reduces a complex symmetric matrix A of order n to a complex symmetric tridiagonal matrix:

$$(1) \quad A = PTP^T \equiv P \begin{bmatrix} a_1 & b_1 & & 0 \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ 0 & & b_{n-1} & a_n \end{bmatrix} P^T,$$

where P is a unitary matrix of order n and T is tridiagonal. For example, the Lanczos tridiagonalization method with partial orthogonalization [9, 12] can be used. The second stage computes the Takagi factorization $T = Q\Sigma Q^T$ of the complex symmetric tridiagonal T . Combining the two stages, we have

$$A = P(Q\Sigma Q^T)P^T = V\Sigma V^T,$$

*Received by the editors February 17, 2005; accepted for publication (in revised form) by P. C. Hansen August 12, 2007; published electronically February 20, 2008.

<http://www.siam.org/journals/simax/30-1/62455.html>

[†]Department of Computing and Software, McMaster University, Hamilton, ON L8S 4K1, Canada (xuw5@mcmaster.ca, qiao@mcmaster.ca).

where $V = PQ$.

In this paper, we focus on the computation of the Takagi factorization of the complex symmetric tridiagonal T using the divide-and-conquer method based on rank-one tearing of TT^H . It is known that the divide-and-conquer method is one of the most efficient methods for computing the eigenvalues and eigenvectors of a large, normally of order larger than dozens, Hermitian tridiagonal matrix [3]. Apparently, the Takagi vectors of T —that is, the columns of Q —are the eigenvectors of the positive semidefinite Hermitian matrix TT^H , since $TT^H = Q\Sigma Q^T \bar{Q}\Sigma Q^H = Q\Sigma^2 Q^H$. However, an eigenvector of TT^H may not be a Takagi vector of T . For example, let

$$T = \begin{bmatrix} 1 & i \\ i & -1 \end{bmatrix}, \quad \text{where } i = \sqrt{-1};$$

then

$$TT^H = \begin{bmatrix} \frac{\sqrt{2}}{4} + \frac{\sqrt{6}}{4}i & -\frac{\sqrt{2}}{2}i \\ -\frac{\sqrt{6}}{4} + \frac{\sqrt{2}}{4}i & \frac{\sqrt{2}}{2}i \end{bmatrix} \begin{bmatrix} 4 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{4} - \frac{\sqrt{6}}{4}i & -\frac{\sqrt{6}}{4} - \frac{\sqrt{2}}{4}i \\ -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}i \end{bmatrix}$$

is an eigenvalue decomposition of TT^H . Using the algorithm in [10], we can obtain the Takagi factorization

$$T = Q\Sigma Q^T = \begin{bmatrix} -\frac{\sqrt{2}}{2} & \frac{1}{2} + \frac{1}{2}i \\ -\frac{\sqrt{2}}{2}i & \frac{1}{2} - \frac{1}{2}i \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2}i \\ \frac{1}{2} + \frac{1}{2}i & \frac{1}{2} - \frac{1}{2}i \end{bmatrix}.$$

In fact, it is shown in [7, Corollary 4.4.5] that if A is complex symmetric and the eigenvalues of AA^H are distinct, and if $AA^H = U\Sigma^2 U^H$, then there exists a diagonal matrix $D = \text{diag}(e^{i\theta_1}, \dots, e^{i\theta_n})$ with real θ_i such that $A = V\Sigma V^T$ with $V = UD$. The diagonal entries of D are determined by the relation $A\bar{U} = U\Sigma D^2$. In the above example, $e^{i\theta_1} = -1/2 + \sqrt{3}i/2$ and $e^{i\theta_2} = -\sqrt{2}/2 - \sqrt{2}i/2$. However, if σ_i^2 is a multiple eigenvalue of AA^H , then, following the proof of Theorem 4.4.3 in [7], we can construct the Takagi vector \mathbf{v}_i corresponding to the singular value σ_i of A from the eigenvector \mathbf{u}_i corresponding to σ_i^2 using

$$\mathbf{v}_i = \alpha_i(A\bar{\mathbf{u}}_i + \sigma_i \mathbf{u}_i),$$

where $\alpha_i = 1/\|A\bar{\mathbf{u}}_i + \sigma_i \mathbf{u}_i\|_2$ is the normalization factor. The details of the transformation will be described in section 3.

The basic idea behind our method is to apply the divide-and-conquer method to TT^H to compute its eigenvectors and eigenvalues. The square roots of the eigenvalues of TT^H are the Takagi values of T . Since an eigenvector of TT^H may not be a Takagi vector of T , we then transform the eigenvectors of TT^H into the Takagi vectors of T . However, explicitly computing TT^H is too expensive and also destroys the tridiagonal structure of T . We will introduce an implicit method for computing the eigenvalue decomposition of TT^H .

The rest of this paper is organized as follows. Section 2 describes a divide-and-conquer method for computing the eigenvalue decomposition of TT^H without explicitly forming TT^H . In section 3, we propose a method for transforming the eigenvectors of TT^H into the Takagi vectors of T . We analyze the sensitivity of the Takagi vectors of T in section 4. Finally, our preliminary numerical experiments are demonstrated in section 5 to show the stability, accuracy, and efficiency of our algorithm.

2. Divide-and-conquer scheme. Let the Takagi factorization of the complex symmetric tridiagonal matrix T in (1) be

$$Q^H T \bar{Q} = \Sigma = \text{diag}(\sigma_1, \dots, \sigma_n) \quad \text{or} \quad T = Q \Sigma Q^T.$$

In the first step, we tear the tridiagonal matrix T into two tridiagonal submatrices of half size. For simplicity, we assume that n is a power of 2 and $m = n/2$; then

$$(2) \quad T = \begin{bmatrix} T_1 & b_m \mathbf{e}_m \mathbf{e}_1^T \\ b_m \mathbf{e}_1 \mathbf{e}_m^T & T_2 \end{bmatrix},$$

where

$$T_1 = \begin{bmatrix} a_1 & b_1 & & 0 \\ b_1 & \ddots & \ddots & \\ & \ddots & \ddots & b_{m-1} \\ 0 & & b_{m-1} & a_m \end{bmatrix}, \quad T_2 = \begin{bmatrix} a_{m+1} & b_{m+1} & & 0 \\ b_{m+1} & \ddots & \ddots & \\ & \ddots & \ddots & b_{n-1} \\ 0 & & b_{n-1} & a_n \end{bmatrix},$$

and \mathbf{e}_1 and \mathbf{e}_m are unit vectors, $[1, 0, \dots, 0]^T$ and $[0, \dots, 0, 1]^T$, respectively. In this section, we present a divide-and-conquer method for computing the eigenvalue decomposition of TT^H given the eigenvalue decompositions of $T_1 T_1^H$ and $T_2 T_2^H$. Our method is based on the rank-one modification of symmetric eigenvalue decomposition.

2.1. Dividing the matrix. We first establish the relations between the eigenvalues and eigenvectors of $T_i T_i^H$, $i = 1, 2$, and those of TT^H as follows. From (2), we get

$$\begin{aligned} TT^H &= \begin{bmatrix} T_1 & b_m \mathbf{e}_m \mathbf{e}_1^T \\ b_m \mathbf{e}_1 \mathbf{e}_m^T & T_2 \end{bmatrix} \begin{bmatrix} T_1^H & \bar{b}_m \mathbf{e}_m \mathbf{e}_1^T \\ \bar{b}_m \mathbf{e}_1 \mathbf{e}_m^T & T_2^H \end{bmatrix} \\ &= \begin{bmatrix} T_1 T_1^H + |b_m|^2 \mathbf{e}_m \mathbf{e}_m^T & b_m \mathbf{e}_m \mathbf{e}_1^T T_2^H + \bar{b}_m T_1 \mathbf{e}_m \mathbf{e}_1^T \\ \bar{b}_m T_2 \mathbf{e}_1 \mathbf{e}_m^T + b_m \mathbf{e}_1 \mathbf{e}_m^T T_1^H & T_2 T_2^H + |b_m|^2 \mathbf{e}_1 \mathbf{e}_1^T \end{bmatrix} \\ &= \begin{bmatrix} T_1 T_1^H & 0 \\ 0 & T_2 T_2^H \end{bmatrix} + \begin{bmatrix} |b_m|^2 \mathbf{e}_m \mathbf{e}_m^T & b_m \mathbf{e}_m \mathbf{e}_1^T T_2^H \\ \bar{b}_m T_2 \mathbf{e}_1 \mathbf{e}_m^T & 0 \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & \bar{b}_m T_1 \mathbf{e}_m \mathbf{e}_1^T \\ b_m \mathbf{e}_1 \mathbf{e}_m^T T_1^H & |b_m|^2 \mathbf{e}_1 \mathbf{e}_1^T \end{bmatrix} \\ &= \begin{bmatrix} T_1 T_1^H & 0 \\ 0 & T_2 (I_m - \mathbf{e}_1 \mathbf{e}_1^T) T_2^H \end{bmatrix} + \begin{bmatrix} b_m \mathbf{e}_m \\ T_2 \mathbf{e}_1 \end{bmatrix} \begin{bmatrix} \bar{b}_m \mathbf{e}_m^T & \mathbf{e}_1^T T_2^H \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 & \bar{b}_m T_1 \mathbf{e}_m \mathbf{e}_1^T \\ b_m \mathbf{e}_1 \mathbf{e}_m^T T_1^H & |b_m|^2 \mathbf{e}_1 \mathbf{e}_1^T \end{bmatrix} \\ &= \begin{bmatrix} T_1 (I_m - \mathbf{e}_m \mathbf{e}_m^T) T_1^H & 0 \\ 0 & T_2 (I_m - \mathbf{e}_1 \mathbf{e}_1^T) T_2^H \end{bmatrix} + \begin{bmatrix} b_m \mathbf{e}_m \\ T_2 \mathbf{e}_1 \end{bmatrix} \begin{bmatrix} \bar{b}_m \mathbf{e}_m^T & \mathbf{e}_1^T T_2^H \end{bmatrix} \\ &\quad + \begin{bmatrix} T_1 \mathbf{e}_m \\ b_m \mathbf{e}_1 \end{bmatrix} \begin{bmatrix} \mathbf{e}_m^T T_1^H & \bar{b}_m \mathbf{e}_1^T \end{bmatrix} \\ (3) \quad &= \begin{bmatrix} T_1 (I_m - \mathbf{e}_m \mathbf{e}_m^T) T_1^H & 0 \\ 0 & T_2 (I_m - \mathbf{e}_1 \mathbf{e}_1^T) T_2^H \end{bmatrix} + \mathbf{z}_1 \mathbf{z}_1^H + \mathbf{z}_2 \mathbf{z}_2^H, \end{aligned}$$

where

$$\mathbf{z}_1 = \begin{bmatrix} b_m \mathbf{e}_m \\ T_2 \mathbf{e}_1 \end{bmatrix} \quad \text{and} \quad \mathbf{z}_2 = \begin{bmatrix} T_1 \mathbf{e}_m \\ b_m \mathbf{e}_1 \end{bmatrix}.$$

From (3), if the eigenvalue decompositions

$$(4) \quad T_1 T_1^H = U_1 \Sigma_1^2 U_1^H \quad \text{and} \quad T_2 T_2^H = U_2 \Sigma_2^2 U_2^H$$

of the positive semidefinite Hermitian matrices $T_1 T_1^H$ and $T_2 T_2^H$ are available, then we can find the eigenvalue decomposition of TT^H by four rank-one modifications. Thus, if the Takagi factorizations of T_1 and T_2 are available, then we can compute the Takagi values of T and the eigenvectors of TT^H by four rank-one modifications. Later in section 3, we will show how to transform the eigenvectors into the Takagi vectors.

Now, we discuss the rank-one modification. Cuppen [2, Theorem 2.1] characterizes the eigenvalues and eigenvectors of the real symmetric rank-one modification. We generalize it to the complex case. The proof is analogous to the one in [2], so it is omitted.

THEOREM 2.1. *Suppose D^2 is a real diagonal matrix $\text{diag}(d_1^2, \dots, d_n^2)$, where $d_1^2 > d_2^2 > \dots > d_n^2$, $\mathbf{z} \in \mathbb{C}^n$ is a vector with no zero entries, and $\rho > 0$ a real scalar; then the eigenvalues of the matrix $D^2 + \rho \mathbf{z} \mathbf{z}^H$ are the n roots $\delta_1^2 > \delta_2^2 > \dots > \delta_n^2$ of the rational function*

$$(5) \quad w(\delta^2) = 1 + \rho \mathbf{z}^H (D^2 - \delta^2 I)^{-1} \mathbf{z} = 1 + \rho \sum_{j=1}^n \frac{|z_j|^2}{d_j^2 - \delta^2}.$$

The corresponding normalized eigenvectors, $\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_n$ of $D^2 + \rho \mathbf{z} \mathbf{z}^H$ are given by

$$(6) \quad \mathbf{g}_j = (D^2 - \delta_j^2 I)^{-1} \mathbf{z} / \|(D^2 - \delta_j^2 I)^{-1} \mathbf{z}\|_2,$$

and d_j^2 strictly separate the eigenvalues δ_j^2 :

$$d_n^2 < \delta_n^2 < d_{n-1}^2 < \delta_{n-1}^2 < \dots < d_1^2 < \delta_1^2 < d_1^2 + \rho \mathbf{z}^H \mathbf{z}.$$

Applying the above theorem, we can compute the eigenvalue decomposition of TT^H from those in (4) via four rank-one modifications. Specifically, suppose that the eigenvalue decomposition $T_1 T_1^H = U_1 \Sigma_1^2 U_1^H$ in (4) is available; then

$$\begin{aligned} T_1 (I_m - \mathbf{e}_m \mathbf{e}_m^T) T_1^H &= T_1 T_1^H - T_1 \mathbf{e}_m \mathbf{e}_m^T T_1^H \\ &= U_1 (\Sigma_1^2 - (U_1^H T_1 \mathbf{e}_m)(U_1^H T_1 \mathbf{e}_m)^H) U_1^H. \end{aligned}$$

Applying Theorem 2.1 to $-\Sigma_1^2 + (U_1^H T_1 \mathbf{e}_m)(U_1^H T_1 \mathbf{e}_m)^H$, we obtain the eigenvalue decomposition of $T_1 T_1^H - T_1 \mathbf{e}_m \mathbf{e}_m^T T_1^H$. Similarly, the eigenvalue decomposition of $T_2 T_2^H - T_2 \mathbf{e}_1 \mathbf{e}_1^T T_2^H$ can be obtained from $T_2 T_2^H = U_2 \Sigma_2^2 U_2^H$ by applying Theorem 2.1. Thus, we suppose

$$(7) \quad T_1 T_1^H - T_1 \mathbf{e}_m \mathbf{e}_m^T T_1^H = \hat{U}_1 \hat{\Sigma}_1^2 \hat{U}_1^H \quad \text{and} \quad T_2 T_2^H - T_2 \mathbf{e}_1 \mathbf{e}_1^T T_2^H = \hat{U}_2 \hat{\Sigma}_2^2 \hat{U}_2^H.$$

Applying the above decompositions to (3), we have

$$(8) \quad TT^H = \begin{bmatrix} \hat{U}_1 & \\ & \hat{U}_2 \end{bmatrix} \left(\begin{bmatrix} \hat{\Sigma}_1^2 & \\ & \hat{\Sigma}_2^2 \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{u}}_1 \\ \hat{\mathbf{u}}_2 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}_1 \\ \hat{\mathbf{u}}_2 \end{bmatrix}^H + \begin{bmatrix} \hat{\mathbf{v}}_1 \\ \hat{\mathbf{v}}_2 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{v}}_1 \\ \hat{\mathbf{v}}_2 \end{bmatrix}^H \right) \begin{bmatrix} \hat{U}_1 & \\ & \hat{U}_2 \end{bmatrix}^H,$$

where $\hat{\mathbf{u}}_1 = b_m \hat{U}_1^H \mathbf{e}_m$, $\hat{\mathbf{u}}_2 = \hat{U}_2^H T_2 \mathbf{e}_1$, $\hat{\mathbf{v}}_1 = \hat{U}_1^H T_1 \mathbf{e}_m$, and $\hat{\mathbf{v}}_2 = b_m \hat{U}_2^H \mathbf{e}_1$. This shows that the eigenvalue decomposition of TT^H can be obtained by two more rank-one modifications.

The numerical computation of the rank-one modification, i.e., the roots of the rational function (5) and the eigenvectors (6) will be treated in section 4.

2.2. Deflation. In this subsection, we remove the assumptions of distinctiveness of the diagonal entries d_i and no zero entries in the modification vector \mathbf{z} by applying the deflation technique given in [14]. We first consider the case when \mathbf{z} has zero entries. It can be easily verified that (d_i^2, \mathbf{e}_i) is an eigenpair of $D^2 + \rho \mathbf{z} \mathbf{z}^H$ if $z_i = 0$. In this case, the problem can be deflated by one for each zero entry in \mathbf{z} . Next, we consider the case when there are two equal diagonal elements in D^2 , say $d_i^2 = d_j^2$. Let P be a Givens rotation such that

$$P \begin{bmatrix} z_i \\ z_j \end{bmatrix} = \begin{bmatrix} * \\ 0 \end{bmatrix};$$

then

$$P \left(\begin{bmatrix} d_i^2 & 0 \\ 0 & d_j^2 \end{bmatrix} + \begin{bmatrix} z_i \\ z_j \end{bmatrix} \begin{bmatrix} z_i \\ z_j \end{bmatrix}^H \right) P^H = \begin{bmatrix} d_i^2 & 0 \\ 0 & d_j^2 \end{bmatrix} + \begin{bmatrix} * \\ 0 \end{bmatrix} \begin{bmatrix} * \\ 0 \end{bmatrix}^H.$$

Thus, when $d_i^2 = d_j^2$ for some $i \neq j$, we can assume $z_i = 0$ or $z_j = 0$. So, the case of equal diagonal elements in D is changed to the case of zero entries in \mathbf{z} .

Due to the rounding errors, we regard two elements d_i^2 and d_j^2 equal if the difference between them is less than a predetermined tolerance tol . How do we determine the tolerance? In our deflation procedure, when d_i^2 and d_j^2 are numerically equal, we find a Givens rotation to transform $[z_i, z_j]^T$ into $[*, 0]^T$. Let $c = \bar{z}_i / \sqrt{|z_i|^2 + |z_j|^2}$ and $s = -\bar{z}_j / \sqrt{|z_i|^2 + |z_j|^2}$; then

$$\begin{bmatrix} c & -s \\ \bar{s} & \bar{c} \end{bmatrix} \begin{bmatrix} d_i^2 \\ d_j^2 \end{bmatrix} \begin{bmatrix} \bar{c} & s \\ -\bar{s} & c \end{bmatrix} = \begin{bmatrix} d_i^2 \\ d_j^2 \end{bmatrix} + E,$$

where

$$E = (d_i^2 - d_j^2) \begin{bmatrix} -|s|^2 & cs \\ \bar{c}\bar{s} & |s|^2 \end{bmatrix}.$$

We set the tolerance tol so that $\|E\|_F \leq \epsilon \|\text{diag}(d_i^2, d_j^2)\|_F$ when $|d_i^2 - d_j^2| \leq tol$, where ϵ is the machine precision. Taking the Frobenius norm on E and $\text{diag}(d_i^2, d_j^2)$, we get

$$\|E\|_F = \sqrt{2} |s| |d_i^2 - d_j^2| \quad \text{and} \quad \|\text{diag}(d_i^2, d_j^2)\|_F \leq \sqrt{2} d_{\max}^2,$$

where $d_{\max} = \max(d_i, d_j)$. Thus, we set the tolerance

$$tol = \frac{d_{\max}^2}{|s|} \epsilon.$$

3. Takagi factorization. As described in the previous section, given the Takagi factorizations of T_1 and T_2 in (2), we can compute the eigenvalue decomposition $TT^H = U \Sigma^2 U^H$ through four rank-one modifications. Let $T = Q \Sigma Q^T$ be the Takagi factorization of T . It is obvious that the Takagi values of T are the square roots of

the eigenvalues of TT^H . It then remains to convert the eigenvectors of TT^H into the Takagi vectors of T . Specifically, given an eigenvector \mathbf{u}_i of TT^H , we want to convert it into a vector \mathbf{q}_i satisfying $T\bar{\mathbf{q}}_i = \sigma_i\mathbf{q}_i$. First, in the case when the eigenvalues are distinct, the eigenvectors of TT^H are uniquely defined up to a scaling factor with unit modulus, which implies that the Takagi vector \mathbf{q}_i is a scalar multiple of the corresponding eigenvector \mathbf{u}_i . Let $T\bar{\mathbf{u}}_i = \xi\sigma_i\mathbf{u}_i$ for some scalar ξ such that $|\xi| = 1$, denote $\xi = e^{2i\phi}$ and define

$$(9) \quad \mathbf{q}_i \equiv e^{i\phi}\mathbf{u}_i.$$

Then

$$T\bar{\mathbf{q}}_i = e^{-i\phi}T\bar{\mathbf{u}}_i = e^{-i\phi}e^{2i\phi}\sigma_i\mathbf{u}_i = e^{i\phi}\sigma_ie^{-i\phi}\mathbf{q}_i = \sigma_i\mathbf{q}_i$$

as desired. Specifically, ξ can be obtained by $\xi = (\mathbf{u}_i^H T\bar{\mathbf{u}}_i)/\sigma_i$ if $\sigma_i \neq 0$; otherwise $\xi = 1$.

Next, in the case of multiple eigenvalues, $T\bar{\mathbf{u}}_i$ may not equal $\xi\sigma_i\mathbf{u}_i$. We construct

$$(10) \quad \mathbf{q}_i = \alpha_i(T\bar{\mathbf{u}}_i + \sigma_i\mathbf{u}_i),$$

where $\alpha_i = 1/\|T\bar{\mathbf{u}}_i + \sigma_i\mathbf{u}_i\|_2$ is the normalization factor. Then

$$T\bar{\mathbf{q}}_i = \alpha_i T(\overline{T\bar{\mathbf{u}}_i + \sigma_i\mathbf{u}_i}) = \alpha_i(T\bar{T}\mathbf{u}_i + \sigma_i T\bar{\mathbf{u}}_i) = \alpha_i(\sigma_i^2\mathbf{u}_i + \sigma_i T\bar{\mathbf{u}}_i) = \sigma_i\mathbf{q}_i.$$

Finally, we check the orthogonality of the Takagi vectors of T converted from the eigenvectors of TT^H . It is obvious that the orthogonality is maintained among the Takagi vectors corresponding to distinct Takagi values because of the orthogonality of the eigenvectors corresponding to distinct eigenvalues. Now, assume that $\mathbf{q}_i, \dots, \mathbf{q}_{i+k-1}$ are the Takagi vectors corresponding to a multiple Takagi value σ_i of multiplicity $k > 1$. The construction of \mathbf{q}_i shows that the subspace spanned by $\mathbf{q}_i, \dots, \mathbf{q}_{i+k-1}$ is the same as the one spanned by $\mathbf{u}_i, \dots, \mathbf{u}_{i+k-1}$, since $\mathbf{q}_i, \dots, \mathbf{q}_{i+k-1}$ are the eigenvectors associated with σ_i^2 . Thus, \mathbf{q}_{i+t} ($t = 0, \dots, k-1$) are orthogonal to \mathbf{q}_j , the Takagi vector corresponding to σ_j , if $\sigma_j \neq \sigma_i$. However, the Takagi vectors corresponding to the equal Takagi values may lose their orthogonality. So, the modified Gram–Schmidt orthogonalization is applied to these vectors to restore the orthogonality. Suppose that \mathbf{q}_{i+t} is one of the Takagi vectors corresponding to σ_i computed from (10), then we orthogonalize it against the previous t vectors $\mathbf{q}_i, \dots, \mathbf{q}_{i+t-1}$ using the modified Gram–Schmidt method.

Now, we give the divide-and-conquer algorithm for computing the Takagi factorization of a complex symmetric tridiagonal matrix.

ALGORITHM 3.1. *Given a complex symmetric and tridiagonal matrix T , this algorithm computes the Takagi factorization $T = Q\Sigma Q^T$. There are two stages in this algorithm. The first stage computes the eigenvalue decomposition $TT^H = U\Sigma^2U^H$; the second stage computes the Takagi vectors \mathbf{q}_i of T from the eigenvectors \mathbf{u}_i of TT^H .*

1. Partition T as (2). If T_1 and T_2 are small enough, then directly compute the eigenvalue decompositions

$$T_1T_1^H = U_1\Sigma_1U_1^H \quad \text{and} \quad T_2T_2^H = U_2\Sigma_2U_2^H.$$

If T_1 and T_2 are large, apply this algorithm to T_1 and T_2 .

2. Apply the deflation and the rank-one modification Theorem 2.1 to $T_1T_1^H - T_1\mathbf{e}_m\mathbf{e}_m^T T_1^H$ and $T_2T_2^H - T_2\mathbf{e}_1\mathbf{e}_1^T T_2^H$ to obtain their eigenvalue decompositions (7). Thus, TT^H has the form (8).

3. Compute the eigenvalue decomposition of TT^H via two rank-one modifications using the deflation and Theorem 2.1.
4. The Takagi values of T are the square roots of the eigenvalues of TT^H .
5. For a single Takagi value, its corresponding Takagi vector \mathbf{q}_i is computed using (9); for a multiple Takagi value, its Takagi vector \mathbf{q}_i is computed using (10) and then orthogonalized against the previously computed Takagi vectors corresponding to the same Takagi value by the modified Gram–Schmidt orthogonalization.

Finally, we present a complexity comparison between the divide-and-conquer method and the implicit QR method. Let $t(n)$ be the number of flops required by the divide-and-conquer method, then

$$\begin{aligned} t(n) &= 2t(n/2) && \text{for the two small submatrices } T_1 \text{ and } T_2 \\ &+ O(n^2) && \text{find the eigenvalues and eigenvectors of } D + \rho\mathbf{z}\mathbf{z}^H \\ &+ 2.25cn^3 && \text{update } U. \end{aligned}$$

Thus, updating U is the major cost in our divide-and-conquer method. Ignoring the $O(n^2)$ terms, we get $t(n) \approx 3cn^3$. The constant c represents the deflation and is much smaller than one in practice [3]. In comparison, the implicit QR method in [10] requires about $6n^3$ flops. Hence, our divide-and-conquer method is more efficient than the implicit QR method.

4. Orthogonality of Takagi vectors. In the previous section, we presented a divide-and-conquer algorithm for computing the Takagi factorization of T . It is based on the rank-one update of the symmetric eigenvalue decomposition. Due to the rounding errors, the orthogonality of the eigenvectors computed by Theorem 2.1 may be lost. In this section, we present an analysis of the orthogonality of the computed eigenvectors and propose techniques for assuring good orthogonality. For simplicity, we assume that the given matrix in the rank-one modification is already deflated.

First, we derive a formula for the eigenvectors \mathbf{g}_j in Theorem 2.1. Differentiating both sides of the function $w(t)$ in (5) with respect to t , we get

$$\|(D^2 - \delta^2 I)^{-1} \mathbf{z}\|_2^2 = \sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \delta^2)^2} = \rho^{-1} |w'(\delta^2)|.$$

Then (6) can be rewritten as

$$(11) \quad \mathbf{g}_j = \left[\frac{z_1}{d_1^2 - \delta_j^2}, \frac{z_2}{d_2^2 - \delta_j^2}, \dots, \frac{z_n}{d_n^2 - \delta_j^2} \right] \frac{\sqrt{\rho}}{\sqrt{w'(\delta_j^2)}}.$$

Let $\hat{\delta}_i^2$ be a computed root of w in (5). In the following, by extending the results in [8], we show that if the relative error in $d_j^2 - \hat{\delta}_i^2$ is small for all i and j , then the computed eigenvectors \mathbf{g}_i have good orthogonality.

THEOREM 4.1. *Denote $\hat{\delta}_i^2$ and $\hat{\delta}_k^2$ as the computed roots of w in (5). Let the relative errors in $d_j^2 - \hat{\delta}_i^2$ and $d_j^2 - \hat{\delta}_k^2$ be θ_i and θ_k , respectively, that is,*

$$d_j^2 - \hat{\delta}_i^2 = (d_j^2 - \delta_i^2)(1 + \theta_i) \quad \text{and} \quad d_j^2 - \hat{\delta}_k^2 = (d_j^2 - \delta_k^2)(1 + \theta_k),$$

and $|\theta_i|, |\theta_k| \leq \tau \ll 1$ for all j , then

$$|\hat{\mathbf{g}}_i^H \hat{\mathbf{g}}_k| = |\mathbf{g}_i^H E \mathbf{g}_k| \leq \tau(2 + \tau) \left(\frac{1 + \tau}{1 - \tau} \right)^2,$$

where $\hat{\mathbf{g}}_i$ and $\hat{\mathbf{g}}_k$ are computed eigenvectors using (11) and E is a diagonal matrix whose i th diagonal entry is

$$(12) \quad E_{ii} = \frac{\theta_i + \theta_k + \theta_i \theta_k}{(1 + \theta_i)(1 + \theta_k)} \left(\frac{w'(\delta_i^2)w'(\delta_k^2)}{w'(\hat{\delta}_i^2)w'(\hat{\delta}_k^2)} \right)^{1/2}.$$

Proof. From (11), we have

$$\begin{aligned} & -\hat{\mathbf{g}}_i^H \hat{\mathbf{g}}_k \\ &= - \left(\sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \delta_k^2)(d_j^2 - \delta_i^2)(1 + \theta_i)(1 + \theta_k)} \right) \frac{\rho}{(w'(\hat{\delta}_i^2)w'(\hat{\delta}_k^2))^{1/2}} \\ &= \left(\sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \delta_k^2)(d_j^2 - \delta_i^2)} - \sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \delta_k^2)(d_j^2 - \delta_i^2)(1 + \theta_i)(1 + \theta_k)} \right) \\ & \quad \frac{\rho}{(w'(\hat{\delta}_i^2)w'(\hat{\delta}_k^2))^{1/2}} \end{aligned}$$

since $\mathbf{g}_i^H \mathbf{g}_k = 0$. Thus, we have

$$\begin{aligned} & |\hat{\mathbf{g}}_i^H \hat{\mathbf{g}}_k| \\ &= \left| \sum_{j=1}^n \left(\frac{|z_j|^2}{(d_j^2 - \delta_k^2)(d_j^2 - \delta_i^2)} \right) \left(1 - \frac{1}{(1 + \theta_i)(1 + \theta_k)} \right) \frac{\rho}{(w'(\hat{\delta}_i^2)w'(\hat{\delta}_k^2))^{1/2}} \right| \\ &= \left| \sum_{j=1}^n \left(\frac{|z_j|^2}{(d_j^2 - \delta_k^2)(d_j^2 - \delta_i^2)} \right) \left(\frac{\theta_i + \theta_k + \theta_i \theta_k}{(1 + \theta_i)(1 + \theta_k)} \right) \left(\frac{w'(\delta_i^2)w'(\delta_k^2)}{w'(\hat{\delta}_i^2)w'(\hat{\delta}_k^2)} \right)^{1/2} \right| \\ & \quad \left| \frac{\rho}{(w'(\delta_i^2)w'(\delta_k^2))^{1/2}} \right| \\ &= |\mathbf{g}_i^H E \mathbf{g}_k| \leq \|E\|_2, \end{aligned}$$

where E is a diagonal matrix, whose diagonal elements are given by (12).

On the other hand, it is easy to show that

$$(13) \quad \frac{w'(\delta_i^2)}{w'(\hat{\delta}_i^2)} = \frac{\sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \delta_i^2)^2}}{\sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \delta_i^2)^2(1 + \theta_i)^2}} \leq (1 + \tau)^2.$$

Substituting $w'(\delta_i^2)/w'(\hat{\delta}_i^2)$ in (12) with (13), we have

$$\max(|E_{ii}|) \leq \frac{\tau + \tau + \tau^2}{(1 - \tau)^2} (1 + \tau)^2 = \tau(2 + \tau) \left(\frac{1 + \tau}{1 - \tau} \right)^2.$$

This completes the proof. \square

Apparently, if the roots δ_i^2 of w are computed in high accuracy, then the relative errors in $d_j^2 - \delta_i^2$ are small, provided that the eigenvalues δ_i^2 are not clustered. Consequently, from the above theorem, the computed eigenvectors $\hat{\mathbf{g}}_i$ have good orthogonality.

We adopt the stable method in [5] for computing the roots δ_i of $w(\delta^2)$ in (5). It is well known that if two quantities x and y are close, then in finite-precision arithmetic it is more accurate to compute $x^2 - y^2$ via the formula $(x + y)(x - y)$ [6]. To avoid explicitly calculating the differences between squared quantities, we reformulate $w(\delta^2)$ in (5) as

$$w(\delta^2) = 1 + \psi_i(\mu) + \varphi_i(\mu) \equiv f_i(\mu),$$

where

$$\psi_1(\mu) = 0, \quad \varphi_1(\mu) = \sum_{j=1}^n \frac{|z_j|^2}{(\zeta_j - \mu)(d_j + d_i + \rho\mu)},$$

and

$$\psi_i(\mu) = \sum_{j=1}^{i-1} \frac{|z_j|^2}{(\zeta_j - \mu)(d_j + d_{i-1} + \rho\mu)}, \quad \varphi_i(\mu) = \sum_{j=i}^n \frac{|z_j|^2}{(\zeta_j - \mu)(d_j + d_{i-1} + \rho\mu)},$$

for $i > 1$, and

$$\begin{aligned} \zeta_j &= (d_j - d_i)/\rho, & \mu &= (\delta - d_i)/\rho, & \text{when } \delta^2 &\in (d_i^2, (d_{i-1}^2 + d_i^2)/2), \\ \zeta_j &= (d_j - d_{i-1})/\rho, & \mu &= (\delta - d_{i-1})/\rho, & \text{when } \delta^2 &\in [(d_{i-1}^2 + d_i^2)/2, d_{i-1}^2]. \end{aligned}$$

In the above formulation, an important property of $f_i(\mu)$ is that it can be evaluated accurately. Moreover, we have formulated the functions $\psi_i(\mu)$ and $\varphi_i(\mu)$ so that explicit calculation of the differences of squares such as $d_j^2 - d_i^2$ and $\delta^2 - d_i^2$ are avoided. There are many zero finding methods, for example, the rational interpolation [1] and bisection and its variations [11, 13]. Following [5], our algorithm for finding the zeros of $f_i(\mu)$ is based on the rational interpolation strategy [1] and its LAPACK implementation `slassd4`. Thus, from [5], the computed eigenvalues have high relative accuracy. The eigenvectors are computed from the computed eigenvalues following the method for computing the eigenvectors in [5], which guarantees numerical orthogonality. Thus, the computed Takagi vectors are numerically orthogonal since they are obtained by converting the eigenvectors.

Finding a root of $f_i(\mu)$ is an iterative process. The stopping criterion plays an important role in the accuracy of the computed roots. Similar to [4], we propose the stopping criterion:

$$(14) \quad |f_i(\mu)| \leq \epsilon n (|\psi_i(\mu)| + |\varphi_i(\mu)| + 1).$$

In the following, we show that by using this criterion, the computed roots $\hat{\delta}_i^2$ of $w(\delta^2)$ are accurate.

Since $w(\delta_i^2) = 0$, we have

$$\begin{aligned} w(\hat{\delta}_i^2) &= w(\hat{\delta}_i^2) - w(\delta_i^2) = \rho \sum_{j=1}^n \frac{|z_j|^2}{d_j^2 - \hat{\delta}_i^2} - \rho \sum_{j=1}^n \frac{|z_j|^2}{d_j^2 - \delta_i^2} \\ &= \rho(\hat{\delta}_i^2 - \delta_i^2) \sum_{j=1}^n \frac{|z_j|^2}{(d_j^2 - \hat{\delta}_i^2)(d_j^2 - \delta_i^2)}. \end{aligned}$$

According to the stopping criterion (14), since $f_i(\mu)$ can be evaluated accurately, we have

$$|w(\hat{\delta}_i^2)| \leq \epsilon n \left(1 + \rho \sum_{j=1}^n \frac{|z_j|^2}{|d_j^2 - \hat{\delta}_i^2|} \right) \leq \rho \epsilon n \left(\sum_{j=1}^n \frac{|z_j|^2}{|d_j^2 - \hat{\delta}_i^2|} + \sum_{j=1}^n \frac{|z_j|^2}{|d_j^2 - \delta_i^2|} \right)$$

since $1 = -\rho \sum_{j=1}^n \frac{|z_j|^2}{d_j^2 - \delta_i^2}$. Without loss of generality, we assume δ_i^2 and $\hat{\delta}_i^2$ are in the same interval, say (d_i^2, d_{i-1}^2) . It follows that $(d_j^2 - \delta_i^2)(d_j^2 - \hat{\delta}_i^2) > 0$. So,

$$\begin{aligned} |w(\hat{\delta}_i^2)| &= \rho |\hat{\delta}_i^2 - \delta_i^2| \sum_{j=1}^n \frac{|z_j|^2}{|(d_j^2 - \hat{\delta}_i^2)(d_j^2 - \delta_i^2)|} \leq \rho \epsilon n \left(\sum_{j=1}^n \frac{|z_j|^2}{|d_j^2 - \hat{\delta}_i^2|} + \sum_{j=1}^n \frac{|z_j|^2}{|d_j^2 - \delta_i^2|} \right) \\ &\leq \rho \epsilon n (4\|D^2 + \rho \mathbf{z}\mathbf{z}^H\|_2 + |\hat{\delta}_i^2 - \delta_i^2|) \sum_{j=1}^n \frac{|z_j|^2}{|(d_j^2 - \hat{\delta}_i^2)(d_j^2 - \delta_i^2)|}, \end{aligned}$$

since $|d_j^2 - \hat{\delta}_i^2| + |d_j^2 - \delta_i^2| \leq 2|d_j^2 - \delta_i^2| + |\hat{\delta}_i^2 - \delta_i^2| \leq 4\|D^2 + \rho \mathbf{z}\mathbf{z}^H\|_2 + |\hat{\delta}_i^2 - \delta_i^2|$. From the above equation, we can get the upper bound for $|\hat{\delta}_i^2 - \delta_i^2|$:

$$|\hat{\delta}_i^2 - \delta_i^2| \leq \frac{4\epsilon n \|D^2 + \rho \mathbf{z}\mathbf{z}^H\|_2}{1 - \epsilon n}.$$

In conclusion, we apply the rational interpolation zero finding method to $f_i(\mu)$ using the stopping criterion (14). We can then obtain accurate eigenvalues δ_i^2 . Provided that the eigenvalues are not clustered, it results in the high relative accuracy of the difference $d_i^2 - \hat{\delta}_i^2$, which implies good orthogonality of the computed eigenvectors of TT^H .

5. Numerical examples. We programmed our divide-and-conquer Algorithm 3.1 in MATLAB and tested it on three types of complex symmetric and tridiagonal matrices. Our experiments were carried out on a server with two 2.4 GHz Xeon CPUs, 1GB RAM, and 80GB disk. The complex symmetric and tridiagonal matrices with predetermined Takagi values were generated as follows. First, a random vector uniformly distributed on $(0, 1]$ was generated and sorted in descending order as a Takagi value vector d . Then, a random unitary matrix was generated as a Takagi vector matrix V . The product $A = V\Sigma V^T$, where $\Sigma = \text{diag}(d)$, was computed as a complex symmetric matrix. Finally, a complex symmetric and tridiagonal T was obtained by applying the Householder transformations to both sides of A . Denoting \hat{Q} and \hat{d} as the computed Takagi vector matrix and Takagi value vector, respectively, the error in the computed Takagi factorization was measured by

$$\gamma_t = \|\hat{Q}\hat{\Sigma}\hat{Q}^T - T\|_2, \quad \text{where } \hat{\Sigma} = \text{diag}(\hat{d}).$$

The error in the computed Takagi values was measured by

$$\gamma_v = \|d - \hat{d}\|_2,$$

and the orthogonality of the computed Takagi vector matrix \hat{Q} was measured by

$$\gamma_o = \|\hat{Q}\hat{Q}^H - I\|_2.$$

TABLE 1

The Takagi factorization of five 256×256 testing matrices with distinct Takagi values.

Example	γ_o	γ_v	γ_t
1	1.3558E-14	3.1347E-14	4.1149E-12
2	2.1679E-14	1.0854E-14	4.3920E-12
3	9.7087E-14	8.4093E-15	1.1309E-12
4	1.1040E-14	1.2622E-14	5.5019E-12
5	3.0840E-14	1.1658E-14	1.1243E-12

Example 1. Five random complex symmetric and tridiagonal matrices of order 256 were generated as described above. In this example, the Takagi values of each matrix were distinct. Table 1 shows that the computed Takagi values and Takagi vectors are accurate.

Example 2. Five random complex symmetric and tridiagonal matrices of order 256 were generated. In this example, we set the five largest Takagi values equal and the four smallest Takagi values equally. Table 2 shows the results.

TABLE 2

The Takagi factorization of five 256×256 testing matrices with multiple Takagi values of small multiplicity.

Example	γ_o	γ_v	γ_t
1	7.5222E-12	1.1331E-14	1.0564E-12
2	2.5397E-12	1.9208E-14	2.6242E-12
3	2.4214E-12	6.0150E-14	6.1179E-12
4	1.9582E-12	4.8421E-14	3.2142E-12
5	6.3841E-12	1.0580E-14	2.4453E-12

TABLE 3

The Takagi factorization of five 256×256 testing matrices with multiple Takagi values of large multiplicity.

Example	γ_o	γ_v	γ_t
1	7.8816E-13	8.8186E-14	4.0040E-12
2	3.7709E-12	2.4154E-14	8.4231E-12
3	4.3532E-13	1.3427E-14	3.4808E-12
4	6.2713E-12	7.4803E-14	1.7887E-12
5	4.5237E-12	5.1166E-14	6.4702E-12

Example 3. Five random T of order 256 were generated. In this example, however, we set the 31 largest Takagi values equal. Table 3 shows that the computed results are accurate.

For performance, we tested our algorithm on random complex symmetric and tridiagonal matrices of five different sizes. For each size, we generated five matrices and ran our divide-and-conquer (DAC) method and the implicit QR (IQR) method [10]. In our divide-and-conquer method, when the size of the submatrices T_i , for $i = 1, 2$, in (2) is less than or equal to 10, its Takagi factorization is computed directly by the implicit QR method. Table 4 shows the average running time and the average factorization error γ_t of the five matrices of same size. The results in Table 4 demonstrate that our method is significantly more efficient than the implicit QR method even for matrices of moderately large size.

TABLE 4

The performance and accuracy comparison of the divide-and-conquer (DAC) method and the implicit QR (IQR) method.

matrix size	Running time (sec)		γ_t	
	DAC method	IQR method	DAC method	IQR method
100	1.14	1.16	1.3352E-14	2.4668E-14
200	3.01	5.47	2.0272E-12	2.9772E-14
400	9.51	26.05	1.7014E-12	6.4860E-14
800	46.88	187.05	1.1338E-11	9.0250E-14
1600	286.14	2091.12	4.2198E-11	2.1552E-13

6. Conclusion. We have proposed a divide-and-conquer method for computing the Takagi factorization of a complex symmetric and tridiagonal matrix and presented an analysis, which shows that our method computes accurate Takagi values and vectors provided that the Takagi values are not clustered. Our preliminary experiments have demonstrated that our method produces accurate results even for matrices with multiple Takagi values and is much more efficient than the implicit QR method [10].

REFERENCES

[1] J. R. BUNCH, C. P. NIELSEN, AND D. C. SORENSEN, *Rank-one modification of the symmetric eigenproblems*, Numer. Math., 31 (1978), pp. 31–48.

[2] J. J. M. CUPPEN, *A divide and conquer method for the symmetric tridiagonal eigenproblem*, Numer. Math., 36 (1981), pp. 177–195.

[3] J. W. DEMMEL, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[4] M. GU AND S. C. EISENSTAT, *A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 172–191.

[5] M. GU AND S. C. EISENSTAT, *A divide-and-conquer algorithm for the bidiagonal SVD*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 79–92.

[6] N. J. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, Philadelphia, 2002.

[7] R. A. HORN AND C. R. JOHNSON, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1990.

[8] E. R. JESSUP AND D. C. SORENSEN, *A parallel algorithm for computing the singular value decomposition of a matrix*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 530–548.

[9] S. QIAO, G. LIU, AND W. XU, *Block Lanczos tridiagonalization of complex symmetric matrices*, in Advanced Signal Processing Algorithms, Architectures, and Implementations XV, F. T. Luk, ed., in Proceedings of the SPIE, Vol. 5910, 2005, pp. 285–295.

[10] F. T. LUK AND S. QIAO, *A fast singular value algorithm for Hankel matrices*, Fast Algorithms for Structured Matrices: Theory and Applications. Contemp. Math. 323, V. Olshevsky, ed., Amer. Math. Soc., 2003, pp. 169–177.

[11] D. P. O’LEARY AND G. W. STEWART, *Computing the eigenvalues and eigenvectors of symmetric arrowhead matrices*, J. Comput. Phys., 90 (1990), pp. 497–505.

[12] S. QIAO, *Orthogonalization techniques for the Lanczos tridiagonalization of complex symmetric matrices*, Advanced Signal Processing Algorithms, Architectures, and Implementations XIV, F. T. Luk, ed., in Proceedings of the SPIE Vol. 5559, 2004, pp. 423–434.

[13] W. E. SHREVE AND M. R. STABNOW, *An eigenvalue algorithm for symmetric bordered diagonal matrices*, Current Trends in Matrix Theory, F. Uhling and R. Grone, eds., Elsevier, New York, 1987, pp. 339–346.

[14] G. W. STEWART, *Matrix Algorithm, Volume II*, Eigensystems, SIAM, Philadelphia, 2001.