Iterative Algorithms for Computing the Takagi Factorization of Complex Symmetric Matrices

Xuezhong Wang, Lu Liang and Maolin Che,

***Abstract*—The main aim of this paper is to establish iterative algorithms for computing the Takagi factorization of complex symmetric matrices. Similar to the classical iterative algorithms of computing the eigenpairs of real symmetric matrices, we derive power-like iterations for computing the Takagi values and associated Takagi vectors of complex symmetric matrices, i.e., the power-like method, the orthogonal-like iteration and the complex symmetric QR-like iteration. We analyze the convergence of these algorithms under some mild conditions. We also investigate the Jacobi-like methods for computing the Takagi factorization of complex symmetric matrices like Jacobi's methods for real symmetric eigenvalue problems. We illustrate our algorithms via numerical examples.**

***Index Terms*—Complex symmetric matrix, Takagi factorization, Singular value decomposition, Power-like method, Orthogonal-like iteration, Complex symmetric QR-like iteration, Jacobi-like methods.**

I. Introduction

RECENTLY, the study of complex symmetric matrices can be divided into three categories: solving complex symmetric linear systems (see [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]), computing singular value decomposition (SVD) of complex symmetric matrices (see [11], [12], [13], [14]) and solving complex-symmetric eigenvalue problems (see [15], [16], [17], [18], [19], [20]). A complex symmetric matrix can be diagonalized by a unitary matrix, which is referred to as the *Autonne-Takagi* factorization, sometimes shortened by *Takagi* factorization. It is originally proved by Leon Autonne [21] and Teiji Takagi [22]. One advantage of Takagi factorization of a complex symmetric matrix is that it reflects the symmetry of the complex symmetric matrix and thus saves the storage and computation about half.

The Takagi factorization of a complex symmetric matrix has many applications, such as the Grunsky inequalities [23], computation of the near-best uniform polynomial or rational approximation of a high degree polynomial on a disk [24], the complex independent component analysis problems [25], and nuclear magnetic resonance [26].

Unfortunately, Matlab and LAPACK [27] do not support complex symmetric structures and treat it as a general complex. To compute the SVD of a complex symmetric matrix in LAPACK, the matrix is first reduced to a bidiagonal matrix, in which the symmetric structure is lost. Similar to the computation of the SVD (see [28]), a standard algorithm for computing the Takagi factorization of a complex symmetric matrix consists of two stages. The first stage is to reduce a $n \times n$ complex symmetric matrix to a complex symmetric tridiagonal matrix, and the second stage is to compute the Takagi fctorization of the complex symmetric tridiagonal matrix from the first stage. For the first stage, Qiao, Liu and Xu [29] derive a block Lanczos method for tridiagonalizing complex symmetric matrices. There are two methods for implementing the second stage: the divide-and-conquer method [30] and a twisted factorization method [13].

X. Wang is with the School of Mathematics and Statistics, Hexi University, Zhangye, 734000, P. R. of China. E-mail:xuezhongwang77@126.com.

L. Liang is with the School of Energy Science and Engineering, Central South University, Changsha, 410083, P. R. of China. E-mail: Lianglu612@yeah.net.

M. Che is with the School of Economic Mathematics, Southwest University of Finance and Economics, Chengdu, 611130, P. R. of China. E-mail:chncml@outlook.com.

As we know, these methods for computing Takagi factorization of complex symmetric matrices are the direct method. Bunse-Gerstner and Gragg [11] derive an iterative algorithm for computing the Takagi factorization of complex symmetric matrices. In this paper, we focus on the computation of the Takagi factorization of the complex symmetric matrices by iterative methods, analogy to the symmetric QR iteration and Jacobi's methods for real symmetric matrices.

Throughout this paper, we use small letters $x, u, v, \ldots$ for scalars, small bold letters $\mathbf{x}, \mathbf{u}, \mathbf{v}, \ldots$ for vectors and $A, B, C, \ldots$ for matrices. For a given integer n, denote $1 : n$ or $[n]$ by the set of $1, 2, \ldots, n$. For a given matrix $A \in \mathbb{C}^{m\times n}$, we use $|A|$, $\|A\|_2$ and $\|A\|_F$ for the absolute values, the largest singular value and the Frobenius norm of A. In detailed algorithmic descriptions, we write $A(i, j)$ or use the Matlab [31] notation $A(i : j, k : l)$ to denote the submatrix of A lying in rows i through j and columns k through l. For a given vector $\mathbf{v} \in \mathbb{C}^n$, $\mathrm{diag}(\mathbf{v})$ is a $n\times n$ diagonal matrix where its diagonal entries are the same as the entries of $\mathbf{v}$. More generality, for given k matrices $A_k \in \mathbb{C}^{n_k\times n_k}$, $D = \mathrm{diag}(A_1, A_2, \ldots, A_k)$ is a $k \times k$ block diagonal matrix with ith diagonal block equal A_i. $\mathbf{0}^{m\times n}$ is a $m \times n$ zero matrix and I_n is the $n \times n$ identity matrix.

The rest of our paper is organized as follows. Section 2 introduces the Takagi factorization of complex symmetric matrices. We consider how to design iterative algorithms for computing the Takagi factorization of a complex symmetric matrix and analyze the convergence of these algorithms in Section 3. In Section 4, we illustrate our algorithms via numerical examples. We conclude our paper in Section 5.

II. Preliminaries

In this section, we introduce the definition of the Takagi factorization of complex symmetric matrices and establish the relationship between the Takagi factorization and the SVD of complex symmetric matrices.

A. Takagi factorization of complex symmetric matrices

The *Takagi* factorization [32] of a complex symmetric matrix A can be written as:

$$A = V\Sigma V^\top, \quad \text{or} \quad \begin{cases} A\overline{\mathbf{v}_i} = \sigma_i \mathbf{v}_i, \\ \overline{A}\mathbf{v}_i = \sigma_i \overline{\mathbf{v}_i}, \end{cases} \tag{II.1}$$

where $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$ is a unitary matrix, $V^\top$ 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*. More general, V is called the Takagi vector matrix of A associated with the Takagi value matrix Σ. Since $V^\top = \overline{V}^*$, where $\overline{V}$ and V^* denote the complex conjugate and the complex conjugated transpose of V, respectively, the Takagi factorization is a symmetric form of the singular value decomposition (SVD); but they are different. The relationships between the Takagi vectors and left-right singular vectors are listed as follows:

(a) If $\mathbf{v}$ is a Takagi vector, then $(\mathbf{v}, \overline{\mathbf{v}})$ is a pair of left-right singular vectors;
(b) A left singular vector is not necessarily a Takagi vector, Xu and Qiao [30] state an example to illustrate this case.

In order to analyze power-like ierations for computing the Takagi factorization of complex symmetric matrices, it is convenient to define an *invariant Takagi subspace* of a complex symmetric matrix, which is generalized from a Takagi vector.

Definition II.1. *An invariant Takagi subspace of a complex symmetric matrix A is a subspace $\mathscr{X}$ of $\mathbb{C}^n$, with the property that $\mathbf{x} \in \mathscr{X}$ implies that $A\overline{\mathbf{x}} \in \mathscr{X}$. We also write it as $A\overline{\mathscr{X}} \subseteq \mathscr{X}$, where $\overline{\mathscr{X}} = \{\overline{\mathbf{x}} \mid \mathbf{x} \in \mathscr{X}\}$.*

If all the Takagi values of A satisfy $\sigma_1 \geq \cdots \geq \sigma_p > \sigma_{p+1} \geq \cdots \geq \sigma_n$, according to Definition II.1, the subspace $\operatorname{span}\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_p\}$ is called a p dominant invariant Takagi subspace of a complex symmetric matrix A, where $\mathbf{v}_i$ is the Takagi vector corresponding to σ_i with $i = 1, 2, \ldots, p$.

Meanwhile, for a given complex symmetric matrix $A \in \mathbb{C}^{n\times n}$, if the Takagi factorization of A is $A = V\Sigma V^\top$, for any unitary diagonal matrix $D \in \mathbb{C}^{n\times n}$ with $d_{ii} = \exp(\iota\varphi_i)$, then

$$\begin{aligned} A &= (VD^{-1/2})D^{1/2}\Sigma D^{1/2}(VD^{-1/2})^\top \\ &= (VD^{-1/2})(D\Sigma)(VD^{-1/2})^\top \\ &= (VD^{-1/2})(D\Sigma)(VD^{-1/2})^\top, \end{aligned}$$

where $\varphi_i \in (-\pi, \pi]$ for all $i = 1, 2, \ldots, n$. Here, $D\Sigma$ and ΣD are complex diagonal matrices and the absolute values of $D\Sigma$ and ΣD are the same as Σ.

III. Algorithms

In this section, we derive four algorithms for computing the Takagi values and associated Takagi vectors of complex symmetric matrices and analyze the convergence of the algorithms. In details, we propose the power-like method of the Takagi factorization for complex symmetric matrices, this method can compute the largest Takagi value and associated Takagi vector of complex symmetric matrices; secondly, we extend the power-like method to compute a p dominant invariant Takagi subspace of a complex symmetric matrix with $p > 1$; we also get a complex symmetric QR-like iteration for computing the Takagi factorization of complex symmetric matrices, similar to the symmetric QR algorithm (see, e.g., [33, Chapter 5.3]) for real symmetric matrices. Under some wild conditions, we show that present three algorithms are effectiveness. Finally, Jacobi-like methods are presented to compute the Takagi factorization of complex symmetric matrices.

For the Takagi factorization of complex symmetric matrices, we have the following lemma.

Lemma III.1. ([34, Lemma 2]) *For given two complex symmetric matrices $A, B \in \mathbb{C}^{n\times n}$. Suppose that the Takagi factorization of A is $A = V\Sigma V^\top$. If there exists a unitary matrix $Q \in \mathbb{C}^{n\times n}$ such that $B = Q^* A\overline{Q}$, then the Takagi values of A and B are the same and the Takagi factorization of B is $B = (Q^*V)\Sigma(Q^*V)^\top$.*

Proof: Since the Takagi factorization of A is $A = V\Sigma V^\top$, where $U \in \mathbb{C}^{n\times n}$ is unitary and Σ is a positive semi-definite diagonal matrix, according to the assumptions, we have

$$B = Q^* A\overline{Q} = Q^* V\Sigma V^\top \overline{Q} = (Q^*V)\Sigma(Q^*V)^\top.$$

Since $Q^*Q = QQ^* = I_n$,

$$(Q^*V)(Q^*V)^* = V^*(QQ^*)V = I_n$$

and $B = (Q^*V)\Sigma(Q^*V)^\top$ is the Takagi factorization of B. We complete the proof. ■

The following lemma should be noted that the Takagi factorization of complex symmetric matrices can be determined via its singular value decomposition of matrix A.

Lemma III.2. ([11, Theorem 2.1]) *Let $A = U\Sigma V^*$ be singular value decomposition of the complex symmetric matrix $A \in \mathbb{C}^{n\times n}$ with the singular values of A satisfies $\sigma_{\mu(1)} > \sigma_{\mu(2)} > \cdots > \sigma_{\mu(k)}$ and $\rho(l)$ the multiplicity of the singular value $\sigma_{\mu(l)}$, so that $\sum_{l=1}^{k} \rho(l) = n$. Let U_l and V_l be the $n \times \rho(l)$ sub-matrices of U and V containing singular vectors corresponding to $\sigma_{\mu(l)}$. Then $W_l = U_l^\top V_l$ is a $\rho(l) \times \rho(l)$ symmetric unitary matrix if $\sigma_{\mu(l)} > 0$. If for all $l \in [k]$, $W_l = Q_l Q_l^\top$ is an symmetric SVD of W_l and D is the unitary block diagonal matrix with $D = \operatorname{diag}(Q_1, Q_2, \ldots, Q_k)$ then $A = (U\overline{D})\Sigma(U\overline{D})^\top$ is a Takagi factorization of A.*

A. Power-like methods

Given a complex symmetric matrix A with $\sigma_1 > \sigma_2 \geq \cdots \geq \sigma_n$, let $\mathbf{v}_1$ be the Takagi vector corresponding to σ_1, then, we have

$$B = (I - \mathbf{v}_1\mathbf{v}_1^*)A(I - \overline{\mathbf{v}_1}\mathbf{v}_1^\top) = (I - \mathbf{v}_1\mathbf{v}_1^*)A(I - \mathbf{v}_1\mathbf{v}_1^*)^\top$$

is also complex symmetric matrix and

$$\begin{aligned} B &= (I - \mathbf{v}_1\mathbf{v}_1^*)A(I - \mathbf{v}_1\mathbf{v}_1^*)^\top \\ &= (I - \mathbf{v}_1\mathbf{v}_1^*)V\Sigma V^\top(I - \mathbf{v}_1\mathbf{v}_1^*)^\top \\ &= V\Sigma_B V^\top, \end{aligned}$$

where V is the same matrix in (II.1) and

$$\Sigma_B = \operatorname{diag}(0, \sigma_2, \sigma_3, \ldots, \sigma_n).$$

It is obvious that Takagi values of B are same as those of A except for σ_1. Now, we design the power-like method for computing the Takagi pair $(\sigma_1, \mathbf{v}_1)$ of A. The algorithm is summarized as Algorithm III-A. Here, we select

$$\frac{\|A\overline{\mathbf{x}_k} - \lambda_k\mathbf{x}_k\|_2}{\||A||\overline{\mathbf{x}_k}| + |\lambda_k||\mathbf{x}_k|\|_2} < tol$$

or

$$\frac{\left\|A\overline{\widetilde{\mathbf{v}}_k} - \widetilde{\sigma}_k \widetilde{\mathbf{v}}_k\right\|_2}{\left\||A||\overline{\widetilde{\mathbf{v}}_k}| + |\widetilde{\sigma}_k||\widetilde{\mathbf{v}}_k|\right\|_2} < tol, \qquad \text{(III.1)}$$

as the convergence criterion of Algorithm III-A, where tol (>0) is arbitrarily small.

Algorithm III.1 Power-like method for complex symmetric matrices

Input: Given a complex symmetric matrix A with $\sigma_1 > \sigma_2 \geq \cdots \geq \sigma_n$
Output: the Takagi value σ_1 and the Takagi vector $\mathbf{v}_1$
Given an initial vector $\mathbf{x}_0 \in \mathbb{C}^n$ with $\|\mathbf{x}_0\|_2 = 1$
for $k = 0, 1, 2, \ldots$ **do**
$\quad\mathbf{y}_{k+1} = A\overline{\mathbf{x}_k}$
$\quad\mathbf{x}_{k+1} = \mathbf{y}_{k+1}/\|\mathbf{y}_{k+1}\|_2$
$\quad\lambda_{k+1} = \mathbf{x}_{k+1}^* A\overline{\mathbf{x}_{k+1}}$
$\quad$Set $\widetilde{\sigma}_{k+1} = |\lambda_{k+1}|$ and $\widetilde{\mathbf{v}}_{k+1} = \exp\left(\iota\frac{\arg(\lambda_{k+1})}{2}\right)\mathbf{x}_{k+1}$
end for

We first apply this algorithm to the case of A is a diagonal matrix with $A = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ satisfies $\sigma_1 > \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. In this case the Takagi vectors are the column vectors $\mathbf{e}_i$ $(i = 1, 2, \ldots, n)$ of the identity matrix. According to the results about the power method for diagonalizable matrices, the convergence of Algorithm III-A is easy to prove.

We prove the convergence of Algorithm III-A if A ia a diagonal positive semi-definite matrices. To analyze a more general case, we rewrite A as $A = V\Sigma V^\top$, where V is a unitary matrix and $\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ is nonnegative. Let $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$, where the columns $\mathbf{v}_i$ are the Takagi vectors satisfy $\|\mathbf{v}_i\|_2 = 1$. Since $\overline{A} = \overline{V}\Sigma V^*$, for $k = 0, 1, 2, \ldots$, it is easy to check that

$$\begin{aligned}(A\overline{A})^k &= (V\Sigma V^\top \overline{V}\Sigma V^*)(V\Sigma V^\top \overline{V}\Sigma V^*) \\ &\quad \ldots (V\Sigma V^\top \overline{V}\Sigma V^*) \\ &\quad = V\Sigma^{2k}V^*, \\ (A\overline{A})^k A &= (V\Sigma V^\top \overline{V}\Sigma V^*)(V\Sigma V^\top \overline{V}\Sigma V^*) \\ &\quad \ldots (V\Sigma V^\top \overline{V}\Sigma V^*)V\Sigma V^\top \\ &\quad = V\Sigma^{2k+1}V^\top,\end{aligned}$$

which follows from $V^\top\overline{V}$ and V^*V are the identity matrix I_n.

For the case of $(A\overline{A})^k$, let us write

$$\mathbf{x}_0 = V(V^*\mathbf{x}_0) \equiv V[\xi_1, \xi_2, \ldots, \xi_n]^\top.$$

It finally leads to

$$(A\overline{A})^k\mathbf{x}_0 = (V\Sigma^{2k}V^*)V\begin{bmatrix}\xi_1\\ \xi_2\\ \vdots\\ \xi_n\end{bmatrix} = V\begin{bmatrix}\xi_1\sigma_1^{2k}\\ \xi_2\sigma_2^{2k}\\ \vdots\\ \xi_n\sigma_n^{2k}\end{bmatrix}$$

$$= \xi_1\sigma_1^{2k}V\begin{bmatrix}1\\ \frac{\xi_2\sigma_2^{2k}}{\xi_1\sigma_1^{2k}}\\ \vdots\\ \frac{\xi_n\sigma_n^{2k}}{\xi_1\sigma_1^{2k}}\end{bmatrix}.$$

As before, the vector in brackets converges to $\mathbf{e}_1$, so $(A\overline{A})^k\mathbf{x}_0$ gets closer and closer to a multiple of $V\mathbf{e}_1 = \mathbf{v}_1$, the Takagi vector corresponding to σ_1.

Meanwhile, for the case of $(A\overline{A})^kA$, denotes

$$\mathbf{x}_0 = \overline{V}(V^\top\mathbf{x}_0) \equiv V([\xi_1, \xi_2, \ldots, \xi_n]^\top).$$

It leads

$$(A\overline{A})^kA\mathbf{x}_0 = (V\Sigma^{2k+1}V^\top)\overline{V}\begin{bmatrix}\xi_1\\ \xi_2\\ \vdots\\ \xi_n\end{bmatrix} = V\begin{bmatrix}\xi_1\sigma_1^{2k+1}\\ \xi_2\sigma_2^{2k+1}\\ \vdots\\ \xi_n\sigma_n^{2k+1}\end{bmatrix}$$

$$= \xi_1\sigma_1^{2k+1}V\begin{bmatrix}1\\ \frac{\xi_2\sigma_2^{2k+1}}{\xi_1\sigma_1^{2k+1}}\\ \vdots\\ \frac{\xi_n\sigma_n^{2k+1}}{\xi_1\sigma_1^{2k+1}}\end{bmatrix}.$$

Similarly, the vector in brackets converges to $\mathbf{e}_1$, so $(A\overline{A})^kA\mathbf{x}_0$ approximates closer and closer to a multiple of $V\mathbf{e}_1 = \mathbf{v}_1$, the Takagi vector corresponding to σ_1. Combining these two cases, we prove the convergence of Algorithm III-A, if the Takagi values of complex symmetric matrices satisfy $\sigma_1 > \sigma_2 \geq \cdots \geq \sigma_n$.

It is well known that the core computation of Algorithm III-A is complex matrix-vector multiplication [28, Problem 4.2.1]. Let $A = B + \iota C$ and $\mathbf{z} = \mathbf{x} + \iota\mathbf{y}$, where $B, C \in \mathbb{R}^{n\times n}$ are two symmetric matrices and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, then

$$A\mathbf{z} = (B + \iota C)(\mathbf{x} + \iota\mathbf{y}) = (B\mathbf{x} - C\mathbf{y}) + \iota(B\mathbf{y} + C\mathbf{x}).$$

Then, in each step, we need to implement Algorithm 1.2.3 in [28] eight times. Hence, the computation complexity of Algorithm III-A is $\mathcal{O}(n^2)$ flops, when $A \in \mathbb{C}^{n\times n}$ is complex symmetric.

B. Orthogonal-like iteration

Our next improvement is to present a algorithm which can converge to a $p(>1)$-dimensional invariant Takagi subspace, rather than one Takagi vector at each time. It is called orthogonal-like iteration (and sometimes Takagi subspace iteration or Simultaneous Iteration). This algorithm is summarized in Algorithm III-B.

We select

$$\sum_{i\neq j} |\Lambda_k(i,j)|^2 < tol, \qquad \text{(III.2)}$$

as the convergence criterion of Algorithm III-B, where tol (>0) is arbitrarily small and $\Lambda_k(i,j)$ is the (i,j)-entry of Λ_k.

Here is a detail analysis of this algorithm. Assume that $\sigma_p > \sigma_{p+1}$. If $p = 1$, then this method and its analysis are identical to the power-like method. When $p > 1$, we deduce that $\operatorname{span}\{X_{k+1}\} = \operatorname{span}\{Y_{k+1}\} = \operatorname{span}\{A\overline{X_{k+1}}\}$, so we have

$$\begin{aligned}\operatorname{span}\{X_{2k}\} &= \operatorname{span}\left\{(A\overline{A})^kX_0\right\} = \operatorname{span}\left\{V\Sigma^{2k}V^*X_0\right\},\\ \operatorname{span}\{X_{2k+1}\} &= \operatorname{span}\left\{(A\overline{A})^kAX_0\right\}\\ &= \operatorname{span}\left\{V\Sigma^{2k+1}V^\top X_0\right\}.\end{aligned}$$

Algorithm III.2 Orthogonal-like iteration for complex symmetric matrices

Input: Given a complex symmetric matrix A with $\sigma_1 \geq \cdots \geq \sigma_p > \sigma_{p+1} \geq \cdots \geq \sigma_n$
Output: $\Sigma_p \in \mathbb{R}^{p\times p}$ is a nonnegative diagonal matrix and $V_p \in \mathbb{C}^{n\times p}$ such that $A\overline{V_p} = V_p\Sigma_p$ and $V_p^* V_p = I_p$
Given an initial matrix $X_0 \in \mathbb{C}^{n\times p}$ with $X_0^* X_0 = I_p$
for $k = 0, 1, 2, \ldots$ **do**
  $Y_{k+1} = A\overline{X_k}$
  Factor $Y_{k+1} = X_{k+1}R_{k+1}$ (QR decomposition)
  $\Lambda_{k+1} = X_{k+1}^* A\overline{X_{k+1}}$
  Set $\widetilde{\Sigma}_{k+1} = |\Lambda_{k+1}|$ and $\widetilde{V}_{k+1} = X_{k+1}D_{k+1}$, where $D_{k+1} \in \mathbb{C}^{p\times p}$ is diagonal and $D_{k+1,jj} = \exp\left(\iota\frac{\arg(\mathbf{x}_{k+1,j}^* A\overline{\mathbf{x}_{k+1,j}})}{2}\right)$ with $j \in [p]$ and $X_{k+1} = (\mathbf{x}_{k+1,1}, \mathbf{x}_{k+1,2}, \ldots, \mathbf{x}_{k+1,p})$.
end for

Note that

$$V\Sigma^{2k}V^*X_0 = V\mathrm{diag}(\sigma_1^{2k}, \sigma_2^{2k}, \ldots, \sigma_n^{2k})V^*X_0$$
$$= \sigma_p^{2k}V\mathrm{diag}((\frac{\sigma_1}{\sigma_p})^{2k}, \ldots, 1\ldots, (\frac{\sigma_n}{\sigma_p})^{2k})V^*X_0,$$
$$V\Sigma^{2k+1}V^\top X_0 = V\mathrm{diag}(\sigma_1^{2k+1}, \sigma_2^{2k+1}, \ldots, \sigma_n^{2k+1})V^\top X_0$$
$$= \sigma_p^{2k+1}V\mathrm{diag}((\frac{\sigma_1}{\sigma_p})^{2k+1}, \ldots, 1\ldots, (\frac{\sigma_n}{\sigma_p})^{2k+1})V^\top X_0.$$

Since $\frac{\sigma_i}{\sigma_p} \geq 1$ if $i \leq p$, and $\frac{\sigma_i}{\sigma_p} < 1$ if $i > p$, we denote

$$\mathrm{diag}(\frac{\sigma_1}{\sigma_p})^{2k}, \ldots, 1\ldots, (\frac{\sigma_n}{\sigma_p})^{2k})V^*X_0 = \begin{pmatrix} P_{2k} \\ Q_{2k} \end{pmatrix},$$
$$\mathrm{diag}((\frac{\sigma_1}{\sigma_p})^{2k+1}, \ldots, 1\ldots, (\frac{\sigma_n}{\sigma_p})^{2k+1})V^\top X_0 = \begin{pmatrix} P_{2k+1} \\ Q_{2k+1} \end{pmatrix},$$

where $Q_k(\in \mathbb{C}^{(n-p)\times p})$ approaches zero like $(\sigma_{p+1}/\sigma_p)^k$, and $P_k(\in \mathbb{C}^{p\times p})$ does not approach zero. Indeed, if P_0 has full rank, then P_k will have full rank too. Let the Takagi vectors matrix be $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n) \equiv (V_p, \widehat{V}_p)$, i.e., $V_p = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_p), \widehat{V}_p = (\mathbf{v}_{p+1}, \mathbf{v}_{p+2}, \ldots, \mathbf{v}_n)$. Then

$$V\Sigma^{2k}V^*X_0 = \sigma_p^{2k}V\begin{pmatrix} P_{2k} \\ Q_{2k} \end{pmatrix}$$
$$= \sigma_p^{2k}(V_pP_{2k} + \widehat{V}_pQ_{2k}),$$
$$V\Sigma^{2k+1}V^\top X_0 = \sigma_p^{2k+1}V\begin{pmatrix} P_{2k+1} \\ Q_{2k+1} \end{pmatrix}$$
$$= \sigma_p^{2k+1}(V_pP_{2k+1} + \widehat{V}_pQ_{2k+1}).$$

Thus $\mathrm{span}\{X_k\}$ converges to

$$\mathrm{span}\{X_{2k}\} = \mathrm{span}\left\{(A\overline{A})^kX_0\right\} = \mathrm{span}\left\{V_pX_{2k} + \widehat{V}_pY_{2k}\right\} \Rightarrow \mathrm{span}\left\{V_pX_{2k}\right\} = \mathrm{span}\{V_p\},$$
$$\mathrm{span}\{X_{2k+1}\} = \mathrm{span}\left\{(A\overline{A})^kAX_0\right\} = \mathrm{span}\left\{V_pX_{2k+1} + \widehat{V}_pY_{2k+1}\right\} \Rightarrow \mathrm{span}\{V_pX_{2k+1}\} = \mathrm{span}\{V_p\}.$$

Hence, the invariant Takagi subspace spanned by the first p Takagi vectors, as desired.

It is well known that the core computation of Algorithm III-B is complex matrix-matrix multiplication and computing the *thin* QR decomposition [28, Theorem 5.2.3] of Y_{k+1}. Let $A = B + \iota C$ and $Z = X + \iota Y$ where $B, C \in \mathbb{R}^{n\times n}$ are two symmetric matrices and $X, Y \in \mathbb{R}^{n\times p}$, then

$$AZ = (B + \iota C)(X + \iota Y) = (BX - CY) + \iota(BY + CX).$$

Then, in each step, we need to implement Algorithm 1.2.3 in [28] $8p$ times. Hence, the computation complexity of Algorithm III-A is $\mathcal{O}(pn^2)$ flops, if $A \in \mathbb{C}^{n\times n}$ is complex symmetric.

Thus, we can let $p = n$ and $|X_0| = I_n$ in the orthogonal-like iteration (Algorithm III-B). The next theorem shows that under certain assumptions, we can use orthogonal-like iteration to compute the Takagi factorization of complex symmetric matrices.

Theorem III.1. *Suppose that A is a complex symmetric matrix. Running orthogonal-like iteration (Algorithm* III-B*) on matrix A with $p = n$ and $|X_0| = I_n$. If the Takagi values of A have distinct values and the principal submatrices $V(1:j, 1:j)$ have full rank, then $A_i = X_i^* A\overline{X_i}$ converges to $D\Sigma$, where $D \in \mathbb{C}^{n\times n}$ satisfies $|D| = I_n$, i.e., $\widetilde{A}_i = \widetilde{V}_i^* A\overline{\widetilde{V}_i}$ converges to Σ. The Takagi values will appear in decreasing order.*

Proof: The assumption about nonsingularity of $V(1:j, 1:j)$ for all j implies that X_0 is nonsingular. Note that X_k is a square unitary matrix, so the Takagi values of A and $A_k = X_k^* A\overline{X_k}$ are the same. Write $X_k = (X_{1k}, X_{2k})$, where X_{1k} has p columns and X_{2k} has $n-p$ columns, thus

$$A_k = X_k^* A\overline{X_k} = \begin{pmatrix} X_{1k}^* A\overline{X_{1k}} & X_{1k}^* A\overline{X_{2k}} \\ X_{2k}^* A\overline{X_{1k}} & X_{2k}^* A\overline{X_{2k}} \end{pmatrix}.$$

Since $\mathrm{span}\{X_{1k}\}$ converges to an invariant Takagi subspace of A, $\mathrm{span}\{A\overline{X_{1k}}\}$ converges to the same subspace, $X_{2k}^* A\overline{X_{1k}}$ and $(X_{1k}^* A\overline{X_{2k}})^\top = X_{2k}^* A\overline{X_{1k}}$ converge to $X_{2k}^* X_{1k} = \mathbf{0}^{(n-p)\times p}$. Since this is true for all $p < n$, every off-diagonal entry of A_k converges to zero, so A_k converges to a complex diagonal matrix. ■

C. Complex symmetric QR-like iteration

Now, our goal is to attain a complex symmetric QR-like iteration for computing the Takagi factorization of complex symmetric matrices, which is needed in the proof of Theorem III.1. Algorithm III-C can realize this process.

Algorithm III.3 QR-like algorithm for computing the Takagi factorization of complex symmetric matrices

Input: Given a complex symmetric matrix A_0 and $U_0 \leftarrow I_n$
Output: $\Lambda \in \mathbb{C}^{n\times n}$ is a complex diagonal matrix and $U \in \mathbb{C}^{n\times n}$ such that $A\overline{U} = U\Lambda$ and $U^*U = I_n$
for $k = 0, 1, 2, \ldots$ **do**
  Factor $A_k = Q_kR_k$ (the QR decomposition)
  Compute $A_{k+1} \leftarrow R_k\overline{Q_k}$ and $U_{k+1} \leftarrow U_kQ_k$
end for

In practice, the matrices X_{k+1} in Algorithm III-B and the matrices Q_k in Algorithm III-C do not need to be computed explicitly. Here, we choose

$$\sum_{i\neq j} |A_{k,ij}|^2 < tol \qquad \text{(III.3)}$$

as the convergence criterion of Algorithm III-C, where tol (> 0) is arbitrarily small and $A_k(i,j)$ is the (i,j)-entry of A_k with $i, j \in [n]$. We note that the absolute value of A_k converges to Σ, i.e., $\overline{D}A_k\overline{D} \rightarrow \Sigma$ and and $U_kD \rightarrow V$ as

$k \to \infty$, where the ith entry of D is $\exp\left(\iota\frac{\arg(a_i)}{2}\right)$ and a_i is the ith entry of A_k with $i \in [n]$.

Since $A_{k+1} = R_k\overline{Q_k} = Q_k^* A_k \overline{Q_k}$, A_{k+1} and A_k have the same Takagi values. We claim that the A_k computed by QR-like iteration is identical to the matrix $X_k^* A \overline{X_k}$ implicitly computed by orthogonal-like iteration.

Lemma III.3. *Suppose that $A \in \mathbb{C}^{n\times n}$ is complex symmetric matrix and $A_k = X_k^* A \overline{X_k}$, where X_k is the matrix computed from orthogonal-like iteration (Algorithm* III-B*), then A_k converges to $D\Sigma$ if all the Takagi values are different, where $D \in \mathbb{C}^{n\times n}$ is diagonal with $|D| = I_n$. The choice of D depends on all matrices X_k and A.*

Proof: We use induction. Assume that $A_k = X_k^* A \overline{X_k}$. From Algorithm III-C, we can derive $A\overline{X_k} = X_{k+1}R_{k+1}$, where X_{k+1} is unitary and R_{k+1} is upper triangular. Then $X_k^* A \overline{X_k} = X_k^*(X_{k+1}R_{k+1})$ is the product of an unitary matrix $Q = X_k^* X_{k+1}$ and an upper triangular matrix $R = R_{k+1} = X_{k+1}^* A \overline{X_k}$; this must be the QR decomposition $A_k = QR$, since the QR decomposition is unique (except for possibly multiplying each column of Q and row of R by -1). Then

$$\begin{aligned} X_{k+1}^* A \overline{X_{k+1}} &= (X_{k+1}^* A \overline{X_k})(\overline{X_k^\top X_{k+1}}) \\ &= R_{k+1}(\overline{X_k^\top X_{k+1}}) \\ &= R\overline{Q}. \end{aligned}$$

This is precisely how the QR iteration maps A_k to A_{k+1}, $X_{k+1}^* A \overline{X_{k+1}} = A_{k+1}$ as desired. ■

It is well known that we can use the symmetric QR iteration (for example, see [33, Chapter 5.3]) to find all eigenvalues and the eigenvectors of a real symmetric matrix. The symmetric QR iteration can be divided into two phases:

1. Given a real symmetric matrix A, find an orthogonal Q such that $QAQ^\top = T$ is tridiagonal.
2. Employing the symmetric QR iteration to matrix T and getting a tridiagonal matrices sequence $T = T_0, T_1, T_2, \ldots$, which converges to a diagonal form.

It is obvious that QR iteration keeps all the T_k are tridiagonal matrices, since $QAQ^\top$ is symmetric and upper Hessenberg, it must also be lower Hessenberg, i.e., tridiagonal.

Similar to this process, we consider how to efficiently implement Algorithm III-C for computing the Takagi factorization of complex symmetric matrices. The desired process is also divided into three phases:

1. Given a complex symmetric matrix A, find an unitary Q such that $Q^* A \overline{Q} = T$ is tridiagonal and complex symmetric, for instance, Qiao, Liu and Xu [29] derive an algorithm for implementing this purpose.
2. Apply the complex symmetric QR-like iteration to T to get a sequence $T = T_0, T_1, T_2, \ldots$ of tridiagonal matrices converging to a complex diagonal form $\widehat{\Sigma}$. (**Note that:** Bunse-Gerstner and Gragg [11] derive a unitary matrix Q by computing the QR decomposition of T^*T to implement T converging to a diagonal form; however, we derive a unitary matrix Q by computing the QR decomposition of T.)
3. Convert the final complex diagonal form $\widehat{\Sigma}$ derived above to a positive semi-definite diagonal form Σ and derive the Takagi matrix V.

We can see that QR-like iteration keeps all the T_k tridiagonal and complex symmetric, since $Q^* A \overline{Q}$ is complex symmetric and upper Hessenberg, it must also be lower Hessenberg, i.e., tridiagonal. This keeps each QR iteration very cheap.

However, how to describe QR iteration with a shift for computing the Takagi factorization of complex symmetric matrices remains open.

D. Jacobi-like method

Jacobi's method [1] is historically the oldest method for the eigenvalue problem, dating to 1846. For a real symmetric matrix A, Jacobi's method does not start by reducing A to a tridiagonal form as the symmetric QR method or divide-conquer method, instead of working on the original dense matrix. Now, we generate a Jacobi-like method for computing the Takagi factorization of complex symmetric matrices.

Given a complex symmetric matrix $A = A_0 \in \mathbb{C}^{n\times n}$. Jacobi's like method produces a unitary matrices sequences $A_1, A_2, \ldots$, which eventually converges to a diagonal matrix determined by the Takagi values. A_{i+1} is obtained from A_i by the formula $A_{i+1} = J_i^\top A_i J_i$, where J_i is a unitary matrix. Thus

$$\begin{aligned} A_m = J_{m-1}^\top A_{m-1} J_{m-1} = J_{m-1}^\top J_{m-2}^\top A_{m-2} J_{m-2} J_{m-1} = \\ \cdots = J_{m-1}^\top J_{m-2}^\top \ldots J_0^\top A_0 J_0 \ldots J_{m-2} J_{m-1} \equiv J^\top A J, \end{aligned}$$

where $J = J_0 \ldots J_{m-2} J_{m-1}$ is a unitary matrix.

If we choose each J_i appropriately, then A_m approaches a diagonal matrix Λ for large m. Thus we can write $\Lambda \approx J^\top A J$ or $\overline{J}\Lambda J^* = (\overline{J})\Lambda(\overline{J})^\top \approx A$. Therefore, according to (II.1) and Lemma III.1, the columns of $\bar{J}$ are approximate Takagi vectors of A.

We make $J^\top A J$ nearly diagonal by iteratively choosing J_i to make one pair of off-diagonal entries of $A_{i+1} = J_i^\top A_i J_i$. To do this, we first consider how to find a unitary matrix Q for computing the Takagi factorization of a 2×2 complex symmetric matrix A.

It is obvious that a 2×2 unitary matrix Q can be written as $\begin{pmatrix} ce^{\iota\varphi_1} & se^{\iota\varphi_2} \\ -se^{-\iota\varphi_3} & ce^{\iota\varphi_4} \end{pmatrix}$, where $\begin{pmatrix} c & s \\ -s & c \end{pmatrix}$ is an orthogonal matrix and $\varphi_1 + \varphi_3 = \varphi_2 - \varphi_4 \pm \pi$ with $\varphi_i \in (-\pi, \pi]$, $(i = 1, 2, 3, 4)$. For a given 2×2 complex symmetric matrix A, we can derive a 2×2 unitary matrix Q such that $Q^\top A Q$ is a diagonal form by Algorithm III-C or some methods given in [11], [30], [13]. Without loss of generality, we assume that c and s are nonnegative.

We derive a Jacobi-like method for computing the Takagi factorization of complex symmetric matrices, summarized in Algorithm III-D.

Now, we analyze the convergence of Algorithm III-D and give the convergent criterion. We first define the quantity $\text{off}(A)$ [28, Secion 8.5.1] as

$$\text{off}(A) = \sqrt{\sum_{i=1}^{n}\sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}|^2}.$$

We see that $\text{off}(A)$ is also the Frobenius norm of the off-diagonal elements. The idea behind Jacobi's-like method is to systematically reduce $\text{off}(A)$. The basic step in a Jacobi-like Takagi value procedure involves these three phases:

[1]Demmel [33] describes some results about Jacobi's method.

Algorithm III.4 Jacobi-like method for computing the Takagi factorization of complex symmetric matrices

Input: Given a complex symmetric matrix A and $U \leftarrow I_n$
Output: A unitary matrix $U \in \mathbb{C}^{n\times n}$ such that $U^\top AU$ is a diagonal form
$U \leftarrow I_n$
for $k = 0, 1, 2, \ldots$ **do**
Step 1: Choose (p, q) such that $|a_{pq}| = \max_{i\neq j} |a_{ij}|$ with $p < q$
Step 2: Write $\widehat{A} = \begin{pmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{pmatrix}$
Step 3: Compute a unitary matrix Q such that $Q^\top \widehat{A} Q$ is a diagonal form
Step 4: Form J such that $J \leftarrow I_n$ except for

$$\begin{pmatrix} J_{pp} & J_{pq} \\ J_{qp} & J_{qq} \end{pmatrix} \leftarrow Q$$

Step 5: Compute $A \leftarrow J^\top AJ$ and $U \leftarrow UJ$
end for

(i) Choose an index pair (p, q) that satisfies $1 \le p < q \le n$;
(ii) Compute a 6-tuple $(c, s, \varphi_1, \varphi_2, \varphi_3, \varphi_4)$ such that

$$\begin{pmatrix} ce^{\iota\varphi_1} & se^{\iota\varphi_2} \\ -se^{-\iota\varphi_3} & ce^{\iota\varphi_4} \end{pmatrix}^\top \begin{pmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{pmatrix} \begin{pmatrix} ce^{\iota\varphi_1} & se^{\iota\varphi_2} \\ -se^{-\iota\varphi_3} & ce^{\iota\varphi_4} \end{pmatrix} \tag{III.4}$$

is a diagonal form;
(iii) Overwrite A with $B = J^\top AJ$ where $J = I_n$ except for

$$(J_{pp}, J_{pq}, J_{qp}, J_{qq}) = (ce^{\iota\varphi_1}, se^{\iota\varphi_2}, -se^{-\iota\varphi_3}, ce^{\iota\varphi_4}).$$

Observe that the matrix B agrees with A except in rows and columns p and q. Moreover, for a given matrix $A \in \mathbb{C}^{n\times n}$, when $P, Q \in \mathbb{C}^{n\times n}$ are unitary, the Frobenius norm of A and PAQ are the same, see, e.g., [28, Secion 2.3.5]. Hence, we find that

$$\begin{aligned} |a_{pp}|^2 + |a_{qq}|^2 + 2|a_{pq}|^2 &= |b_{pp}|^2 + |b_{qq}|^2 + 2|b_{pq}|^2 \\ &= |b_{pp}|^2 + |b_{qq}|^2. \end{aligned}$$

It follows that

$$\begin{aligned} \text{off}(B)^2 =& \|B\|_F^2 - \sum_{i=1}^n |b_{ii}|^2 = \|A\|_F^2 - \sum_{i=1}^n |a_{ii}|^2 \\ &+ (|b_{pp}|^2 + |b_{qq}|^2 - |a_{pp}|^2 - |a_{qq}|^2) \\ =& \text{off}(A)^2 - 2|a_{pq}|^2. \end{aligned} \tag{III.5}$$

In this sense, A moves closer to diagonal form with each Jacobi-like step. Since a_{pq} is the largest off-diagonal entry in each Jacobi-like step of Algorithm III-D, then

$$\text{off}(A)^2 \le N(|a_{pq}|^2 + |a_{qp}|^2)$$

where $N = \frac{n(n-1)}{2}$. It follows from (III.5) that

$$\text{off}(B)^2 \le \left(1 - \frac{1}{N}\right) \text{off}(A)^2.$$

By induction, if $A^{(k)}$ denotes the matrix after k Jacobi-like updates, then

$$\text{off}(A^{(k)})^2 \le \left(1 - \frac{1}{N}\right)^k \text{off}(A^{(0)})^2 = \left(1 - \frac{1}{N}\right)^k \text{off}(A)^2.$$

It implies that the classical Jacobi-like procedure converges with a linear rate.

Next, we consider how to improve the Jacobi-like method for computing the Takagi factorization of complex symmetric matrices. First of all, we state that the trouble of Algorithm III-D is that the updates involve $\mathcal{O}(n)$ complex flops while the search for the optimal (p, q) is $\mathcal{O}(n^2)$. We can overcome this trouble by row-by-row fashion, (see [28, Section 8.5.4]). The row Jacobi-like method for complex symmetric matrices is summarized in Algorithm 3.5.

Algorithm III.5 Row Jacobi-like method for computing the Takagi factorization of complex symmetric matrices

Input: Given a complex symmetric matrix A and $U \leftarrow I_n$
Output: A unitary matrix $U \in \mathbb{C}^{n\times n}$ such that $U^\top AU$ is a diagonal form
for $k = 0, 1, 2, \ldots$ **do**
for $p = 1 : n - 1$ **do**
for $q = p + 1 : n$ **do**
Step 1: Write $\widehat{A} = \begin{pmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{pmatrix}$
Step 2: Compute a unitary matrix Q such that $Q^\top \widehat{A} Q$ is a diagonal form
Step 3: Form J such that $J \leftarrow I_n$ except for

$$\begin{pmatrix} J_{pp} & J_{pq} \\ J_{qp} & J_{qq} \end{pmatrix} \leftarrow Q$$

Step 5: Compute $A \leftarrow J^\top AJ$ and $U \leftarrow UJ$
end for
end for
end for

When Algorithm III-D or 3.5 is terminated, we need to convert the final diagonal form to Σ and the unitary matrix U to the Takagi vector matrix V. Let $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ the final diagonal form, the strategy is:
(i) Form a diagonal matrix D as

$$\text{diag}\left(\exp\left(\iota\text{arg}\frac{\lambda_1}{2}\right), \exp\left(\iota\text{arg}\frac{\lambda_1}{2}\right), \ldots, \exp\left(\iota\text{arg}\frac{\lambda_1}{2}\right)\right);$$

(ii) Compute Σ and V as

$$\Sigma = |\Lambda|, \quad V = \overline{U}D.$$

When Algorithm III-C is terminated, we also need to convert the final diagonal form to the Takagi value matrix Σ and U to the Takagi vector matrix V by the above strategy.

In following, for a given 2×2 complex symmetric matrix A, we consider how to more efficiently compute a unitary Q such that $Q^\top AQ$ is a diagonal form, similar to the process about computing the Schur decomposition of 2×2 real symmetric matrices, (see [28, Section 8.5.2]), that is, implementing Step 2 in Algorithm III-D or 3.5.

Let the 2×2 complex symmetric matrix A as

$$A = \begin{pmatrix} a_1 & a_2 \\ a_2 & a_3 \end{pmatrix} = \begin{pmatrix} r_1e^{\iota\theta_1} & r_2e^{\iota\theta_2} \\ r_2e^{\iota\theta_2} & r_3e^{\iota\theta_3} \end{pmatrix}, \tag{III.6}$$

where $a_i \in \mathbb{C}$, nonnegative $r_i \in \mathbb{R}$ and $\theta_i \in (-\pi, \pi]$ with $i = 1, 2, 3$. When the matrix $\begin{pmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{pmatrix}$ in (III.4) is replaced by A, given in (III.6), and there exists a 2×2 unitary matrix Q such that $B = Q^\top AQ$ is a diagonal matrix, then $b_{12} =$

$b_{21} = 0$, that is,

$$\begin{aligned} b_{12} &= a_1 ce^{\iota\varphi_1} se^{\iota\varphi_2} + a_2 ce^{\iota\varphi_1} ce^{\iota\varphi_4} - a_2 se^{\iota\varphi_2} se^{-\iota\varphi_3} \\ &\quad - a_3 ce^{\iota\varphi_4} se^{-\iota\varphi_3} \\ &= r_1 cse^{\iota(\theta_1+\varphi_1+\varphi_2)} + r_2 c^2 e^{\iota(\theta_2+\varphi_1+\varphi_4)} \\ &\quad - r_2 s^2 e^{\iota(\theta_2+\varphi_2-\varphi_3)} - r_3 cse^{\iota(\theta_3+\varphi_4-\varphi_3)} \\ &= 0, \end{aligned}$$

where c and s are nonnegative, and $\varphi_1 + \varphi_3 = \varphi_2 - \varphi_4 \pm \pi$ with $\varphi_i \in (-\pi, \pi]$ and $i = 1, 2, 3, 4$. We can find a unitary matrix $Q \in \mathbb{C}^{2\times 2}$ such that $Q^\top AQ$ is a diagonal form through the process:

(i) Compute two nonnegative scalars c and s such that

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix} |A\overline{A}| \begin{pmatrix} c & s \\ -s & c \end{pmatrix}^\top$$

is a nonnegative diagonal matrix;

(ii) Choose a pair $(\varphi_1, \varphi_2, \varphi_3, \varphi_4)$ such that

$$r_1 cse^{\iota(\theta_1+\varphi_1+\varphi_2)} + r_2 c^2 e^{\iota(\theta_2+\varphi_1+\varphi_4)} - r_2 s^2 e^{\iota(\theta_2+\varphi_2-\varphi_3)} \\ -r_3 cse^{\iota(\theta_3+\varphi_4-\varphi_3)} = 0$$

where $\varphi_1 + \varphi_3 = \varphi_2 - \varphi_4 \pm \pi$ with $\varphi_i \in (-\pi, \pi]$ and $i = 1, 2, 3, 4$.

IV. Numerical Examples

In this section, the computations are implemented in Matlab Version 2013a on a laptop with Intel Core i5-4200M CPU (2.50GHz) and 7.89GB RAM. We test the accuracy and efficiency of our algorithms by computing the Takagi factorization or the Takagi-like factorization of random complex symmetric matrices. Suppose that the computational accuracy tol is $10e-10$. All floating point numbers in each example have 4 significant digits after the decimal point.

We assume that the dimension of testing matrices in each example is 100 and program our Algorithms III-A, III-B and III-C in Matlab for random generated complex symmetric matrices. Random complex symmetric matrices with predetermined singular values in the following examples were generated as follows [2]. First, a vector $\mathbf{s}$, which includes n Takagi values, is initialized. Then, a random unitary matrix V was generated by the QR decomposition of a random $n\times n$ complex matrix. Finally, a complex symmetric matrix A was obtained by the product $V\Sigma V^\top$, where $\Sigma = \mathrm{diag}(\mathbf{s})$.

Example IV.1. *In this example, we use Algorithm* III-A *to compute the largest Takagi value and associated Takagi vector of a random complex symmetric matrix A, where the largest Takagi value of A is larger than others of A and its termination condition is given in* (III.2).

Let σ be the largest Takagi value of A and $\mathbf{v}$ be the Takagi vector corresponding to σ. Meanwhile, denoting $\widehat{\mathbf{v}}$ and $\widehat{\sigma}$ as the computed Takagi vector and Takagi value, respectively, we compute the error, determined by the computed Takagi value, via

$$\gamma_\sigma = \|\sigma - \widehat{\sigma}\|_2\,,$$

and the orthogonality of the computed Takagi vector $\widehat{\mathbf{v}}$ made by

$$\gamma_o = \|\widehat{\mathbf{v}}^*\widehat{\mathbf{v}} - 1\|_2\,.$$

[2]Xu and Qiao [13] use a similar strategy to generate random complex symmetric tridiagonal matrices.

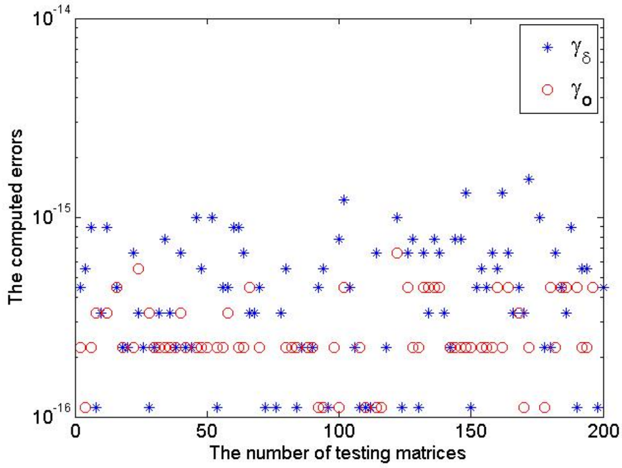

Fig. 1. All results about computing the largest Takagi value and associated Takagi vector of random complex symmetric matrices by Algorithm III-A

Running Algorithm III-A 200 times, Figure 1 show the computational results.

Example IV.2. *In this example, we use Algorithm* III-B *to compute the first p Takagi values and associated Takagi vectors of a random generated complex symmetric matrix A, where the first p Takagi values of A are larger than others of A and the termination condition of algorithm is given in* (III.2).

Let $\mathbf{s}_p$ be the vector determined by the first p Takagi values of A and V_p be the Takagi vector matrix corresponding to $\mathbf{s}_p$. Meanwhile, denoting $\widehat{V}_p$ and $\widehat{\mathbf{s}}_p$ as the computed Takagi vector matrix and the vector of the computed p Takagi values, respectively, we compute the error in the p dominant Takagi factorization measured by

$$\gamma_A = \left\| A\overline{\widehat{V}_p} - \widehat{V}_p \mathrm{diag}(\widehat{\mathbf{s}}_p) \right\|_2,$$

the error in the computed Takagi values had by

$$\gamma_{\mathbf{s}} = \|\mathbf{s}_p - \widehat{\mathbf{s}}_p\|_2\,,$$

and the orthogonality of the computed Takagi vector $\widehat{V}_p$ determined by

$$\gamma_o = \left\| \widehat{V}_p^* \widehat{V}_p - I_p \right\|_2,$$

where the order of all entries in $\mathbf{s}_p$ is the same as the order of all entries in $\widehat{\mathbf{s}}_p$.

Running Algorithm III-B 5 times, Table I show computational results with $p = 5$.

TABLE I
ALL RESULTS ABOUT THE ERROR IN THE LARGEST 5 COMPUTED TAKAGI VALUES AND ASSOCIATED TAKAGI VECTORS OF RANDOM COMPLEX SYMMETRIC MATRICES.

sample	γ_A	$\gamma_{\mathbf{s}}$	γ_o
1	0.001e-11	0.355e-11	0.001e-11
2	0.001e-11	0.260e-11	0.001e-11
3	0.438e-11	0.238e-11	0.001e-11
4	0.301e-10	0.679e-11	0.001e-11
5	0.1120e-10	0.5662e-11	0.0003e-11

Example IV.3. *In this example, we use Algorithm* III-C *to compute the Takagi factorization of a random complex* *symmetric matrix A, where all Takagi values of A are distinct and its termination condition is given in* (III.3).

Let $\mathbf{s}$ be the vector of all the Takagi values of A and V be the Takagi vector matrix corresponding to $\mathbf{s}$. Meanwhile, denoting $\widehat{V}$ and $\widehat{\mathbf{s}}$ as the computed Takagi vector matrix and the vector of the computed Takagi values, respectively, we compute the error in the Takagi factorization measured by

$$\gamma_A = \left\| A\overline{\widehat{V}} - \widehat{V}\mathrm{diag}(\widehat{\mathbf{s}}) \right\|_2,$$

the error in the computed Takagi values made by

$$\gamma_{\mathbf{s}} = \|\mathbf{s} - \widehat{\mathbf{s}}\|_2,$$

and the orthogonality of the computed Takagi vector $\widehat{V}$ characterized by

$$\gamma_o = \left\| \widehat{V}^* \widehat{V} - I_n \right\|_2,$$

where the order of all entries in $\mathbf{s}_p$ is the same as the order of all entries in $\widehat{\mathbf{s}}_p$.

Table II shows that the computed Takagi values and Takagi vectors are accurate.

TABLE II
THE TAKAGI FACTORIZATION OF FIVE TESTING MATRICES WITH DISTINCT TAKAGI VALUES.

sample	γ_A	$\gamma_{\mathbf{s}}$	γ_o
1	0.366e-10	0.356e-10	0.001e-11
2	0.971e-11	0.263e-11	0.002e-11
3	0.437e-11	0.238e-11	0.001e-11
4	0.301e-10	0.067e-10	0.001e-11
5	0.112e-10	0.056e-10	0.003e-11

Example IV.4. *In this example, the testing matrix is chosen from* [3], [8]. *The testing matrix A is*

$$A = (-\omega^2 M + K) + \iota(\omega C_V + C_H),$$

where M and K are the inertia and stiffness matrices, respectively; C_V and C_H are the viscous and hysteretic damping matrices, respectively; and ω is the driving circular frequency.

We take $C_H = \mu K$ with $\mu = 0.02$ being a damping coefficient, $\omega = 2\pi$, $C_V = \frac{1}{2}M$, and K the five point centered difference matrix approximating the negative Laplacian operator with homogeneous Dirichlet boundary condition on a uniform mrsh in the unit square $[0,1] \times [0,1]$ with the mesh size $h = \frac{1}{m+1}$. In this case, the matrix $K \in \mathbb{R}^{m\times m}$ possesses the form $K = I_m \otimes V_m + V_m \otimes I_m$ with $V_m = h^{-2}\mathrm{tridiag}(-1, 2, -1) \in \mathbb{R}^{m\times m}$.

In our example, we assume that $m = 3$. The meanings of γ_A, $\gamma_{\mathbf{s}}$ and γ_o are given in Example IV.3. For different matrices M, all results are given in Table III.

TABLE III
γ_A, $\gamma_{\mathbf{s}}$ AND γ_o FOR COMPLEX SYMMETRIC MATRICES.

M	I_n	$2I_n$	$5I_n$	$10I_n$	$15I_n$
γ_A	0.114e-11	0.114e-11	0.025e-10	0.468e-11	0.083e-10
$\gamma_{\mathbf{s}}$	0.049e-11	0.186e-11	0.112e-10	0.712e-11	0.122e-10
γ_o	0.001e-11	0.002e-11	0.001e-10	0.001e-11	0.002e-11

TABLE IV
γ_A, $\gamma_{\mathbf{s}}$ AND γ_o FOR COMPLEX SYMMETRIC MATRICES.

σ_2	5	10	50	80	100
γ_A	0.157e-10	0.235e-11	0.311e-11	0.236e-11	0.146e-11
$\gamma_{\mathbf{s}}$	0.371e-10	0.294e-11	0.112e-11	0.209e-11	0.219e-11
γ_o	0.001e-10	0.001e-11	0.001e-11	0.001e-11	0.001e-11

Example IV.5. *In this example, the testing complex symmetric matrix is chosen from* [4], [6]. *The testing matrix A is given in*

$$A = H + \sigma_1 I_n + \iota\sigma_2 I_n,$$

In addition, we set $\sigma_1 = 100$ and H is the same as the matrix K given in Example IV.4. We see that A is the coefficient matrix of the complex symmetric linear system leaded by the following form of the Helmholtz equation:

$$-\triangle u + \sigma_1 u + \iota\sigma_2 u = f,$$

where σ_1 and σ_2 are real coefficient functions, u satisfies Dirichlet boundary conditions in $[0,1] \times [0,1]$.

In our example, we assume that $n = 9$. The meanings of γ_A, $\gamma_{\mathbf{s}}$ and γ_o are given in Example IV.3. For different positive scalars σ_2, all results are given in Table IV.

V. CONCLUSIONS

In this paper, we derive iterative algorithms for computing the Takagi factorization of complex symmetric matrices and analyze the convergence of these methods. Numerical examples show that our algorithms are accuracy for implementing this process.

However, we do not consider how to implement efficiently these algorithms for computing the Takagi factorization, if the Takagi values are not always distinct. In the near future, we shall consider how to improve these algorithms for computing the US-eigenpairs of a complex symmetric tensor and the U-eigenpairs of a generic complex tensor, which is referred to Ni, Qi and Bai [35] or the generalized tensor eigenvalue problems [36], [37], [38].

ACKNOWLEDGMENT

The author would like to thank the editor and referees for their detailed comments which greatly improved the presentation of the paper.

REFERENCES

[1] S. R. Arridge, H. Egger, and M. Schlottbom, “Preconditioning of complex symmetric linear systems with applications in optical tomography,” *Appl. Numer. Math.*, vol. 74, pp. 35–48, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.apnum.2013.06.008

[2] O. Axelsson and A. Kucherov, “Real valued iterative methods for solving complex symmetric linear systems,” *Numer. Linear Algebra Appl.*, vol. 7, no. 4, pp. 197–218, 2000. [Online]. Available: http://dx.doi.org/10.1002/1099-1506(200005)7:4¡197::AID-NLA194¿3.0.CO;2-S

[3] Z.-Z. Bai, M. Benzi, and F. Chen, “On preconditioned MHSS iteration methods for complex symmetric linear systems,” *Numer. Algorithms*, vol. 56, no. 2, pp. 297–317, 2011. [Online]. Available: http://dx.doi.org/10.1007/s11075-010-9441-6

[4] D. Bertaccini, “Efficient preconditioning for sequences of parametric complex symmetric linear systems,” *Electron. Trans. Numer. Anal.*, vol. 18, pp. 49–64 (electronic), 2004.

[5] V. E. Howle and S. A. Vavasis, “An iterative method for solving complex-symmetric systems arising in electrical power modeling,” *SIAM J. Matrix Anal. Appl.*, vol. 26, no. 4, pp. 1150–1178, 2005. [Online]. Available: http://dx.doi.org/10.1137/S0895479800370871

[6] C.-X. Li and S.-L. Wu, "A double-parameter GPMHSS method for a class of complex symmetric linear systems from Helmholtz equation," *Math. Probl. Eng.*, pp. Art. ID 894 242, 7, 2014. [Online]. Available: http://dx.doi.org/10.1155/2014/894242

[7] T. Sogabe and S.-L. Zhang, "A COCR method for solving complex symmetric linear systems," *J. Comput. Appl. Math.*, vol. 199, no. 2, pp. 297–303, 2007. [Online]. Available: http://dx.doi.org/10.1016/j.cam.2005.07.032

[8] S.-L. Wu, "Several variants of the Hermitian and skew-Hermitian splitting method for a class of complex symmetric linear systems," *Numer. Linear Algebra Appl.*, vol. 22, no. 2, pp. 338–356, 2015. [Online]. Available: http://dx.doi.org/10.1002/nla.1952

[9] X. Wang and Z. Wang, "Preconditioned parallel multisplitting usaor methods for h-matrices linear systems," *IAENG International Journal of Applied Mathematics*, vol. 44, no. 1, pp. 45–52, 2014.

[10] M. U. Rehman, C. Vuik, and G. Segal, "Preconditioners for the steady incompressible navier-stokes problem," *IAENG International Journal of Applied Mathematics*, vol. 38, no. 4, pp. 223–232, 2008.

[11] A. Bunse-Gerstner and W. B. Gragg, "Singular value decompositions of complex symmetric matrices," *J. Comput. Appl. Math.*, vol. 21, no. 1, pp. 41–54, 1988. [Online]. Available: http://dx.doi.org/10.1016/0377-0427(88)90386-X

[12] M. Ferranti, T. H. Le, and R. Vandebril, "A comparison between the complex symmetric based and classical computation of the singular value decomposition of normal matrices," *Numer. Algorithms*, vol. 67, no. 1, pp. 109–120, 2014. [Online]. Available: http://dx.doi.org/10.1007/s11075-013-9777-9

[13] W. Xu and S. Qiao, "A twisted factorization method for symmetric svd of a complex symmetric tridiagonal matrix," *Numer. Linear Algebra Appl.*, vol. 16, no. 10, pp. 801–815, 2009.

[14] J. B. Erway, R. F. Marcia, and J. A. Tyson, "Generalized diagonal pivoting methods for tridiagonal systems without interchanges," *IAENG International Journal of Applied Mathematics*, vol. 40, no. 4, pp. 269–275, 2010.

[15] S. S. Ahmad and V. Mehrmann, "Perturbation analysis for complex symmetric, skew symmetric, even and odd matrix polynomials," *Electron. Trans. Numer. Anal.*, vol. 38, pp. 275–302, 2011.

[16] P. Arbenz and O. Chinellato, "On solving complex-symmetric eigenvalue problems arising in the design of axisymmetric VCSEL devices," *Appl. Numer. Math.*, vol. 58, no. 4, pp. 381–394, 2008. [Online]. Available: http://dx.doi.org/10.1016/j.apnum.2007.01.019

[17] P. Arbenz and M. E. Hochstenbach, "A Jacobi-Davidson method for solving complex symmetric eigenvalue problems," *SIAM J. Sci. Comput.*, vol. 25, no. 5, pp. 1655–1673 (electronic), 2004. [Online]. Available: http://dx.doi.org/10.1137/S1064827502410992

[18] J. Danciger, "A min-max theorem for complex symmetric matrices," *Linear Algebra Appl.*, vol. 412, no. 1, pp. 22–29, 2006. [Online]. Available: http://dx.doi.org/10.1016/j.laa.2005.06.020

[19] V. I. Hasanov, "An iterative method for solving the spectral problem of complex symmetric matrices," *Comput. Math. Appl.*, vol. 47, no. 4-5, pp. 529–540, 2004. [Online]. Available: http://dx.doi.org/10.1016/S0898-1221(04)90043-0

[20] Q. Zhang and X. Wang, "Complex-valued neural network for hermitian matrices," *Engineering Letters*, vol. 25, no. 3, pp. 312–320, 2017.

[21] L. Autonne, *Sur les matrices hypohermitiennes et sur les matrices unitaires*. Rey, 1915, vol. 38.

[22] T. Takagi, "On an algebraic problem reluted to an analytic theorem of carathéodory and fejér and on an allied theorem of landau," in *Japanese Journal of Mathematics: transactions and abstracts*, vol. 1, no. 0. The Mathematical Society of Japan, 1924, pp. 83–93.

[23] C. Pommerenke, *Univalent functions*. Vandenhoeck & Ruprecht, Göttingen, 1975, with a chapter on quadratic differentials by Gerd Jensen, Studia Mathematica/Mathematische Lehrbücher, Band XXV.

[24] L. N. Trefethen, "Near-circularity of the curve in complex Chebyshev approximation," *J. Approx. Theory*, vol. 31, no. 4, pp. 344–367, 1981. [Online]. Available: http://dx.doi.org/10.1016/0021-9045(81)90102-7

[25] P. Comon, "Independent component analysis, a new concept?" *Signal processing*, vol. 36, no. 3, pp. 287–314, 1994.

[26] A. D. Bain, "Chemical exchange in NMR," *Progress in Nuclear Magnetic Resonance Spectroscopy*, vol. 43, no. 3, pp. 63–103, 2003.

[27] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammerling, A. McKenney *et al.*, *LAPACK Users' Guide*. SIAM, 1999, vol. 9.

[28] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., ser. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, 2013.

[29] S. Qiao, G. Liu, and W. Xu, "Block Lanczos tridiagonalization of complex symmetric matrices," in *Optics & Photonics 2005*. International Society for Optics and Photonics, 2005, pp. 591 010–591 010.

[30] W. Xu and S. Qiao, "A divide-and-conquer method for the Takagi factorization," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 1, pp. 142–153, 2008. [Online]. Available: http://dx.doi.org/10.1137/050624558

[31] I. T. M. Works, *Matlab Reference Guide*. Math Works, Incorporated, 1992.

[32] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge University Press, Cambridge, 2012.

[33] J. W. Demmel, *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997. [Online]. Available: http://dx.doi.org/10.1137/1.9781611971446

[34] R. C. Thompson, "Singular values and diagonal elements of complex symmetric matrices," *Linear Algebra Appl.*, vol. 26, pp. 65–106, 1979. [Online]. Available: http://dx.doi.org/10.1016/0024-3795(79)90173-3

[35] G. Ni, L. Qi, and M. Bai, "Geometric measure of entanglement and U-eigenvalues of tensors," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 1, pp. 73–87, 2014. [Online]. Available: http://dx.doi.org/10.1137/120892891

[36] C. Cui, Y. Dai, and J. Nie, "All real eigenvalues of symmetric tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 4, pp. 1582–1601, 2014.

[37] W. Ding and Y. Wei, "Generalized tensor eigenvalue problems," *SIAM Journal on Matrix Analysis and Applications*, pp. 11–36, 2015.

[38] T. G. Kolda and J. R. Mayo, "An adaptive shifted power method for computing generalized tensor eigenpairs," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 4, pp. 1563–1581, 2014.