RESEARCH　　　　　　　　　　　　　　　　　　　　　　　　**Open Access**

A DFT-based approximate eigenvalue and singular value decomposition of polynomial matrices

Mahdi Tohidian[1], Hamidreza Amindavar[1*] and Ali M Reza[2]

Abstract

In this article, we address the problem of singular value decomposition of polynomial matrices and eigenvalue decomposition of para-Hermitian matrices. Discrete Fourier transform enables us to propose a new algorithm based on uniform sampling of polynomial matrices in frequency domain. This formulation of polynomial matrix decomposition allows for controlling spectral properties of the decomposition. We set up a nonlinear quadratic minimization for phase alignment of decomposition at each frequency sample, which leads to a compact order approximation of decomposed matrices. Compact order approximation of decomposed matrices makes it suitable in filterbank and multiple-input multiple-output (MIMO) precoding applications or any application dealing with realization of polynomial matrices as transfer function of MIMO systems. Numerical examples demonstrate the versatility of the proposed algorithm provided by relaxation of paraunitary constraint, and its configurability to select different properties.

1 Introduction

Polynomial matrices have been used for a long time for modeling and realization of multiple-input multiple-output (MIMO) systems in the context of control theory [1]. Nowadays, polynomial matrices have a wide spectrum of applications in MIMO communications [2-6], source separation [7], and broadband array processing [8]. They also have a dominant role in development of multirate filterbanks [9].

More recently, there have been much interest in polynomial matrix decomposition such as QR decomposition [10-12], eigenvalue decomposition (EVD) [13,14], and singular value decomposition (SVD) [5,11]. Lambert [15] has utilized Discrete Fourier transform (DFT) domain to change the problem of polynomial EVD to pointwise EVD. Since EVD is obtained at each frequency separately, eigenvectors are known at each frequency up to a scaling factor. Therefore, this method requires many frequency samples to avoid abrupt changes in adjacent eigenvectors.

Although, many methods of designing principle component filterbanks have been developed that are equivalent to EVD of pseudo circulant polynomial matrices [16,17], the next pioneering work on polynomial matrix EVD is presented by McWhirter et al. [13]. They use an extension of Jacobi algorithm known as SBR2 for EVD of para-Hermitian polynomial matrices which guarantees exact paraunitarity of eigenvector matrix. Since final goal of SBR2 algorithm is to have strong decorrelation, the decomposition does not necessarily satisfy spectral majorization property. SBR2 algorithm has also been modified for QR decomposition and SVD [10,11].

Jacobi-type algorithms are not the only proposed methods for polynomial matrix decomposition. Another iterative method for spectrally majorized EVD is presented in [14] which is based on the maximization of zeroth-order diagonal energy. Spectral majorization property of this algorithm is verified via simulation. Followed by the work of [6], a DFT-based approximation of polynomial SVD is also proposed in [18] which uses model order truncation by phase optimization.

In this article, we present polynomial EVD and SVD based on DFT formulation. It transforms the problem of polynomial matrix decomposition to the problem

*Correspondence: hamidami@aut.ac.ir
[1]Department of Electrical Engineering, Amirkabir University of Technology, P.O. Box 15914, Tehran, Iran
Full list of author information is available at the end of the article

of, pointwise in frequency, constant matrix decomposition. At first it seems that applying inverse DFT on the decomposed matrices leads to polynomial EVD and SVD of the corresponding polynomial matrix. However, we will show later in this article that in order to have compact order decomposition, phase alignment of decomposed constant matrices in DFT domain results in polynomial matrices with considerably lower order. For this reason, a quadratic nonlinear minimization problem is set up to minimize the decomposition error for a given finite order constraint. Consequently, the required number of frequency samples and computational complexity of decomposition reduce dramatically. The algorithm provides compact order matrices as an approximation of polynomial matrix decomposition for an arbitrary polynomial order. This is suitable in MIMO communications and filterbank applications, where we deal with realization of MIMO linear time invariant systems. Moreover, formulation of polynomial EVD and SVD in DFT domain enables us to select the property of decomposition. We show that if eigenvalues (singular values) intersect at some frequencies in frequency domain, smooth decomposition, and spectrally majorized decomposition are distinct. The proposed algorithm is able to reach to either of these properties.

The remainder of this article is organized as follows. The relation between polynomial matrix decomposition and DFT matrix decomposition is formulated in Section 2. In Section 3, two important spectral properties of decomposition, namely spectral majorization and smooth decomposition, are provided using appropriate arrangement of singular values (eigenvalues) and corresponding singular vectors (eigenvectors). The equality of polynomial matrix and dft matrix decomposed matrices decompositions are guaranteed via the finite duration constraint, which is investigated in Section 4. The finite duration constraint imposes the phase angles of singular vector (eigenvector) to minimize a nonlinear quadratic function. A solution for this problem is proposed in Section 5. Section 6 presents the results of some computer simulations which are considered to demonstrate performance of the proposed decomposition algorithm.

1.1 Notation

Some notational conventions are as follows: constant values, vectors, and matrices are in regular character lower case, lower case over-arrow, and upper case, respectively. Coefficients of polynomial (scalar, vector, and matrix) are with indeterminate variable n in the square brackets. Any polynomial (scalar, vector, and matrix) is distinguished by bold character and indeterminate variable z in the parenthesis and its DFT by bold character and indeterminate variable k in the brackets.

2 Problem formulation

Denote a $p \times q$ polynomial matrix $\mathbf{A}(z)$ such that each element of $\mathbf{A}(z)$ is a polynomial. Equivalently, we can indicate this type of matrix by coefficient matrix $A[n]$,

$$\mathbf{A}(z) = \sum_{n=N_{\min}}^{N_{\max}} A[n] z^{-n} \qquad (1)$$

where $A[n]$ is only non-zero in the interval $[N_{\min}, N_{\max}]$. Define the effective degree of $\mathbf{A}(z)$ as $N_{\max} - N_{\min}$ (or the length of $A[n]$ as $N_{\max} - N_{\min} + 1$).

The polynomial matrix multiplication of a $p \times q$ matrix $\mathbf{A}(z)$ and a $q \times t$ matrix $\mathbf{B}(z)$ is defined as

$$\begin{aligned} \mathbf{C}(z) &= \mathbf{A}(z)\mathbf{B}(z) \\ \mathbf{c}_{ij}(z) &= \sum_{k=1}^{q} \mathbf{a}_{ik}(z)\mathbf{b}_{kj}(z). \end{aligned}$$

We can obtain the coefficient matrix of product by matrix convolution of $A[n]$ and $B[n]$, that is defined as

$$\begin{aligned} C[n] &= A[n] * B[n] \\ c_{ij}[n] &= \sum_{k=1}^{q} a_{ik}[n] * b_{kj}[n] \end{aligned}$$

where $*$ denotes the linear convolution operator.

Denote para-conjugate of a polynomial matrix as

$$\tilde{\mathbf{A}}(z) = \mathbf{A}_*^T(z^{-1}) = \sum_{N_{\min}}^{N_{\max}} A^H[n] z^n.$$

in which, $*$ as a subscript denotes the complex conjugate of coefficients in the polynomial matrix $\mathbf{A}(z)$.

A matrix is said to be para-Hermitian if $\tilde{\mathbf{A}}(z) = \mathbf{A}(z)$ or equivalently $A[n] = A^H[-n]$. We call a polynomial matrix paraunitary if $\tilde{\mathbf{U}}(z)\mathbf{U}(z) = \mathbf{I}$, where $\mathbf{I}$ is a $q \times q$ identity matrix.

Thin EVD of a $p \times p$ para-Hermitian polynomial matrix $\mathbf{A}(z)$ is of the form

$$\mathbf{A}(z) = \mathbf{U}(z)\mathbf{\Lambda}(z)\tilde{\mathbf{U}}(z), \qquad (2)$$

and thin SVD of a $p \times q$ arbitrary polynomial matrix is of the form,

$$\mathbf{A}(z) = \mathbf{U}(z)\mathbf{\Sigma}(z)\tilde{\mathbf{V}}(z) \qquad (3)$$

where $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are $p \times r$ and $q \times r$ paraunitary matrices, respectively. $\mathbf{\Lambda}(z)$ and $\mathbf{\Sigma}(z)$ represent $r \times r$ diagonal matrices where r is the rank of $\mathbf{A}(z)$.

We can equivalently write EVD of a para-Hermitian matrix and SVD of a polynomial matrix in coefficient matrix form

$$A[n] = U[n] * \Lambda[n] * U^H[-n] \qquad (4)$$

$$A[n] = U[n] * \Sigma[n] * V^H[-n] \qquad (5)$$

in which, $U[n]$, $V[n]$, $\Lambda[n]$, and $\Sigma[n]$ are the coefficient matrices corresponding to $\mathbf{U}(z)$, $\mathbf{V}(z)$, $\mathbf{\Lambda}(z)$, and $\mathbf{\Sigma}(z)$.

In general, EVD and SVD of a finite-order polynomial matrix are not finite order. As an example, suppose EVD of para-Hermitian polynomial matrix

$$A(z) = \begin{bmatrix} 2 & z^{-1}+1 \\ z+1 & 2 \end{bmatrix}. \tag{6}$$

Eigenvalues and eigenvectors of the polynomial matrix in (6) are neither of finite order nor rational

$$\Lambda(z) = \begin{bmatrix} 2 + (z^{-1}+2+z)^{1/2} & 0 \\ 0 & 2 - (z^{-1}+2+z)^{1/2} \end{bmatrix}$$

$$\mathbf{U}(z) = \frac{1}{\sqrt{2}} \begin{bmatrix} (z^{-1}+1)(z^{-1}+2+z)^{-1/2} & z^{-1} \\ 1 & (z^{-1}+1)(z^{-1}+2+z)^{-1/2} \end{bmatrix}.$$

The same results can be found for polynomial QR decomposition in [12].

We mainly explain the proposed algorithm for polynomial SVD, yet wherever it seems necessary we explain the result for both decomposition.

The decomposition in (3) can also be approximated by samples of discrete-time Fourier transform, yields a decomposition off the form

$$\mathbf{A}[k] = \mathbf{U}'[k]\,\boldsymbol{\Sigma}'[k]\,\mathbf{V}'^H[k], \quad k = 0, 1, \ldots, K-1. \tag{7}$$

Such a decomposition can be obtained by taking the K-point DFT of coefficient matrix $A[n]$,

$$\mathbf{A}[k] = \mathbf{A}(z)|_{z=w_K^k} = \sum_{N_{\min}}^{N_{\max}} A[n]\,w_K^{kn}, \quad k = 0, 1, \ldots, K-1, \tag{8}$$

where $w_K = \exp(-j2\pi/K)$.

DFT formulation plays an important role in decomposition of polynomial matrices because it replaces the problem of polynomial SVD that involves many protracted steps with K conventional SVD that are pointwise in frequency. It also enables us to control spectral properties of the decomposition. However, it causes two inherent drawbacks:

1. Regardless of what is the trajectory of polynomial singular values in frequency domain, conventional SVD order singular values irrespectively of the ordering in neighboring frequency samples.
2. In frequency domain, samples of polynomial singular vectors are known up to a scalar complex exponential by using the SVD at each frequency sample, which yields to discontinuous variation between neighboring frequency samples.

The first issue is directly dealt with the spectral properties of the decomposition. In Section 3, we would explain why arranging singular values in decreasing order yields to approximate spectral majorization, while smooth decomposition requires *rearrangement* of singular values and their corresponding singular vectors.

For the second issue, suppose conventional SVD of an arbitrary constant matrix A. If the pair $\vec{u}$ and $\vec{v}$ are the left and right singular vectors corresponding to a non-zero singular value, for an arbitrary scalar phase angle θ, the pair $e^{j\theta}\vec{u}$ and $e^{j\theta}\vec{v}$ are also left and right singular vectors corresponding to the same singular value. Although this non-uniqueness is trivial in conventional SVD, it plays a crucial role in polynomial SVD. When we perform SVD at each frequency of DFT matrix as in (7), these non-uniquenesses in phase exist at each frequency regardless of other frequency samples.

Denote $\vec{u}_i[k]$ and $\vec{v}_i[k]$ the ith column vector of the desired matrices $\mathbf{U}(z)$ and $\mathbf{V}(z)$. Then all the vectors of the form

$$\begin{cases} \vec{u}'_i[k] = e^{j\theta_i[k]}\vec{u}_i[k] \\ \vec{v}'_i[k] = e^{j\theta_i[k]}\vec{v}_i[k], \quad i = 1, 2, \ldots, r \\ \sigma'_i[k] = \sigma_i[k] \end{cases} \tag{9}$$

have the chance to appear as the ith column of $\mathbf{U}'[k]$ and $\mathbf{V}'[k]$, and ith diagonal element of $\boldsymbol{\Sigma}'[k]$, respectively. Moreover, in many applications, specially those which are related to MIMO precoding, we can relax constraints of the problem by letting singular values to be complex (see applications of polynomial SVD in [4,18])

$$\begin{cases} \vec{u}'_i[k] = e^{j\theta_i^u[k]}\vec{u}_i[k] \\ \vec{v}'_i[k] = e^{j\theta_i^v[k]}\vec{v}_i[k] \qquad , \quad i = 1, 2, \ldots, r. \\ \sigma'_i[k] = e^{j(\theta_i^v[k]-\theta_i^u[k])}\sigma_i[k] \end{cases} \tag{10}$$

Given this situation, singular values have not all their conventional meaning. For instance, the greatest singular value is conventionally 2-norm of the corresponding matrix, which is not true for complex singular values. The process of compensating singular vectors for these phases is what we call *phase alignment* and is developed in Section 4.

Based on what was mentioned above, Algorithm 1 gives the descriptive pseudo code for DFT-based SVD. Modifications of the algorithm for EVD of para-Hermitian matrices are straightforward. If at each frequency sample all singular values are in decreasing order, REARRANGE function (which is described in Algorithm 2) is only required for smooth decomposition, otherwise for spectral majorization, no further arrangement is required. For the phase alignment, first we need to compute phase angles which is indicated in the algorithm by DOGLEG function and is described in Algorithm 3.

3 Spectral majorized decomposition versus smooth decomposition

Two of the most appealing decomposition properties are *smooth decomposition* [19] and *spectral majorization* [13]. These two objectives do not always occur at the same

---

Algorithm 1 Approximate SVD

$\left[U[n], \Sigma[n], V[n]\right] \leftarrow \text{ASVD}\left(A[n]\right)$
for $k = 0, 1, \cdots, K - 1$
    Compute $\mathbf{A}[k]$ from (8):
        $\mathbf{A}[k] = \sum_{N_{\min}}^{N_{\max}} A[n] \, w_K^{kn}$
    Decompose $\mathbf{A}[k]$ from (7):
        $\left[\mathbf{U}'[k], \mathbf{\Sigma}[k], \mathbf{V}'[k]\right] \leftarrow \text{SVD}\left(\mathbf{A}[k]\right)$
end(for)
If smooth decomposition is required use Algorithm 2:
    $\left[\mathbf{U}'[k], \mathbf{\Sigma}'[k], \mathbf{V}'[k]\right] \leftarrow \textbf{REARRANGE}\left(\mathbf{U}'[k], \mathbf{\Sigma}'[k], \mathbf{V}'[k]\right)$
for $i = 1, 2, \cdots, r$
    Compute phase angles using Algorithm 3:
        $\left[\theta_i^u[k], \theta_i^v[k]\right] \leftarrow \textbf{DOGLEG}\left(\vec{\mathbf{u}}_i'[k], \vec{\mathbf{v}}_i'[k]\right)$
    for $k = 0, 1, \ldots, K - 1$
        Phase alignment using (9) or (10)
    end(for)
end(for)
for $n = 0, 1, \ldots, M - 1$
    Compute decomposed polynomial matrices:
        $U[n] \leftarrow \sum_{k=0}^{K} \mathbf{U}[k] \, W^{-kn}$
        $V[n] \leftarrow \sum_{k=0}^{K} \mathbf{V}[k] \, W^{-kn}$
        $\Sigma[n] \leftarrow$ Diagonal elements of $U^H[-n] * A[n] * V[n]$
end(for)

---

time, hence we should choose which one we are willing to use as our main objective.

In many filterbank applications which are dealt with principle components filterbank, spectral majorization and strong decorrelation are both required [16]. Since smooth decomposition leads to more compact decomposition, in cases that the only objective is strong decorrelation, exploiting smooth decomposition is reasonable. The DFT-based approach of polynomial matrix decomposition is capable of decomposing a matrix with either of these properties with small modification.

Polynomial EVD of a para-Hermitian matrix is said to have spectral majorization property if [13,16]

$$\lambda_1(e^{j\omega}) \geq \lambda_2(e^{j\omega}) \geq \cdots \geq \lambda_r(e^{j\omega}), \quad \forall \omega.$$

Note that, eigenvalues corresponding to para-Hermitian matrices are real in all frequencies.

We can extend the definition to the polynomial SVD, replacing singular values with eigenvalues in the definition, we have

$$\sigma_1(e^{j\omega}) \geq \sigma_2(e^{j\omega}) \geq \cdots \geq \sigma_r(e^{j\omega}), \quad \forall \omega.$$

If we let singular values to be complex, we can replace absolute value of singular values in the definition.

A polynomial matrix have no discontinuity in frequency domain, hence we modify definition of smooth decomposition presented in [19] to fit with our problem and avoid unnecessary discussions.

Polynomial EVD (SVD) of a matrix is said to possess smooth decomposition if eigenvectors (singular vectors) have no discontinuity in frequency domain, that is

$$\left|\frac{d}{d\omega}\mathbf{u}_{il}(e^{j\omega})\right| < \infty, \quad \forall \omega \text{ and } \begin{matrix} i = 1, 2, \ldots, r \\ l = 1, 2, \ldots, p, \end{matrix} \quad (11)$$

where $\mathbf{u}_{il}$ is the lth element of $\vec{\mathbf{u}}_i$.

If eigenvalues (singular values) of a polynomial matrix intersect at some frequencies, the spectral majorization and smooth decomposition are not simultaneously realizable. As an example, suppose $\mathbf{A}(z)$ is a polynomial matrix with $\vec{\mathbf{u}}_1(z)$ and $\vec{\mathbf{u}}_2(z)$ are eigenvectors corresponding to distinct eigenvalues $\lambda_1(z)$ and $\lambda_2(z)$, respectively. Lets assume $\vec{\mathbf{u}}_1(e^{j\omega})$ and $\vec{\mathbf{u}}_2(e^{j\omega})$ have no discontinuity in frequency domain, and $\lambda_1(e^{j\omega})$ and $\lambda_2(e^{j\omega})$ intersect at some frequencies. Denote

$$\lambda_1'(e^{j\omega}) = \begin{cases} \lambda_1(e^{j\omega}) & \lambda_1(e^{j\omega}) \geq \lambda_2(e^{j\omega}) \\ \lambda_2(e^{j\omega}) & \lambda_1(e^{j\omega}) < \lambda_2(e^{j\omega}) \end{cases},$$

$$\lambda_2'(e^{j\omega}) = \begin{cases} \lambda_2(e^{j\omega}) & \lambda_1(e^{j\omega}) \geq \lambda_2(e^{j\omega}) \\ \lambda_1(e^{j\omega}) & \lambda_1(e^{j\omega}) < \lambda_2(e^{j\omega}) \end{cases}, \quad (12)$$

Algorithm 2 Rearrangement for smooth decomposition

$\left[\mathbf{U}'[k],\Sigma[k],\mathbf{V}'[k]\right]\leftarrow\textbf{REARRANGE}\left(\mathbf{U}'[k],\Sigma[k],\mathbf{V}'[k]\right)$
for $k=1,2,\ldots,K$
    Define $S=\{1,2,\ldots,r\}$
    for $i=1,2,\ldots,r$
        $\iota\leftarrow\underset{j\in S}{\arg\max}\dfrac{|c_{ij}^{u'}[k]|+|c_{ij}^{v'}[k]|}{2}$
        $\vec{\mathbf{u}}_i''[k]\leftarrow\vec{\mathbf{u}}_\iota'[k]$
        $\vec{\mathbf{v}}_i''[k]\leftarrow\vec{\mathbf{v}}_\iota'[k]$
        $\sigma_i''[k]\leftarrow\sigma_\iota[k]$
        $S\leftarrow S-\{\iota\}$
    end(for)
    $\mathbf{U}'[k]\leftarrow\mathbf{U}''[k]$
    $\mathbf{V}'[k]\leftarrow\mathbf{V}''[k]$
    $\Sigma[k]\leftarrow\Sigma''[k]$
end(for)

---

and

$$\vec{\mathbf{u}}_1'(e^{j\omega})=\begin{cases}\vec{\mathbf{u}}_1(e^{j\omega}) & \lambda_1(e^{j\omega})\geq\lambda_2(e^{j\omega})\\\vec{\mathbf{u}}_2(e^{j\omega}) & \lambda_1(e^{j\omega})<\lambda_2(e^{j\omega})\end{cases},$$

$$\vec{\mathbf{u}}_2'(e^{j\omega})=\begin{cases}\vec{\mathbf{u}}_2(e^{j\omega}) & \lambda_1(e^{j\omega})\geq\lambda_2(e^{j\omega})\\\vec{\mathbf{u}}_1(e^{j\omega}) & \lambda_1(e^{j\omega})<\lambda_2(e^{j\omega})\end{cases}. \tag{13}$$

Obviously, $\vec{\mathbf{u}}_1'(e^{j\omega})$ and $\vec{\mathbf{u}}_2'(e^{j\omega})$ are eigenvectors corresponding to distinct eigenvalues $\lambda_1'(e^{j\omega})$ and $\lambda_2'(e^{j\omega})$, respectively. Note that, $\lambda_1'(e^{j\omega})\geq\lambda_2'(e^{j\omega})$ for all frequencies, which means $\lambda_1(e^{j\omega})$ and $\lambda_2(e^{j\omega})$ are spectrally majorized. However, $\vec{\mathbf{u}}_1'(e^{j\omega})$ and $\vec{\mathbf{u}}_2'(e^{j\omega})$ are discontinuous at intersection frequencies of $\lambda_1(e^{j\omega})$ and $\lambda_2(e^{j\omega})$, which implies that they are not smooth anymore. In this situation, although $\lambda_1'(e^{j\omega})$, $\lambda_2'(e^{j\omega})$, $\vec{\mathbf{u}}_1'(e^{j\omega})$, and $\vec{\mathbf{u}}_2'(e^{j\omega})$ are not even analytic, we can approximate them with finite order polynomials.

If a decomposition has spectral majorization, its eigenvalues (singular values) are of decreasing order in all frequencies. Therefore, they are in decreasing order in any arbitrary frequency sample set, including DFT frequencies. Obviously the converse is only approximately true. Hence, for polynomial EVD to possess spectral majorization approximately, it suffices to arrange sampled eigenvalues (singular values) of (7) in decreasing order. Since we only justify spectral majorization at DFT frequency samples, the resulting EVD (SVD) may possess the property only approximately. Similar results can be seen in [14,20].

To have smooth singular vectors, we propose an algorithm based on inner product of consecutive frequency samples of singular vectors. We can accumulate smoothing requirement in (11) for all r elements as

$$\left\|\frac{d}{d\omega}\vec{\mathbf{u}}_i(e^{j\omega})\right\|<\infty,\quad\forall\omega\ \text{and}\quad i=1,2,\ldots,r. \tag{14}$$

Let B be the upper bound of norm of derivative and $\Re\{\cdot\}$ be the real value of a complex value.

For an arbitrary $\Delta\omega$ we have

$$\left\|\vec{\mathbf{u}}_i(e^{j(\omega+\Delta\omega)})-\vec{\mathbf{u}}_i(e^{j\omega})\right\|^2=2-2\Re\left\{\vec{\mathbf{u}}_i^H(e^{j(\omega+\Delta\omega)})\vec{\mathbf{u}}_i(e^{j\omega})\right\}$$
$$<(\Delta\omega B)^2\quad\forall\omega, \tag{15}$$

that is, for a smooth singular vector $\Re\left\{\vec{\mathbf{u}}_i(e^{j(\omega+\Delta\omega)})\ \vec{\mathbf{u}}_i(e^{j\omega})\right\}$ can be made to be as close to unity as desired by making $\Delta\omega$ sufficiently small. In our problem $\vec{\mathbf{u}}_i(e^{j\omega})$ is sampled uniformly with $\Delta\omega=\frac{2\pi}{K}$. Since EVD is performed at each frequency sample independently, $\vec{\mathbf{u}}_i[k]$ and $\vec{\mathbf{u}}_i[k+1]$ are not necessarily two consecutive frequency samples of a smooth eigenvector. Therefore, we should rearrange eigenvalues and eigenvectors to yield smooth decomposition. This can be done for each sample of eigenvector $\vec{\mathbf{u}}_i[k]$ by seeking for the eigenvector of successor sample $\vec{\mathbf{u}}_j[k+1]$ with the most value of $\Re\left\{\vec{\mathbf{u}}_i^H[k]\ \vec{\mathbf{u}}_j[k+1]\right\}$.

Define inner product $c_{ij}^u[k]$ as

$$c_{ij}^u[k]=\vec{\mathbf{u}}_i^H[k-1]\ \vec{\mathbf{u}}_j[k].$$

Since, $\vec{\mathbf{u}}_i'[k]$ is a scalar phase multiplication of $\vec{\mathbf{u}}_i[k]$, computation of $\Re\{c_{ij}^u[k]\}$ is not possible before phase alignment. Due to (15), for sufficiently small $\Delta\omega$, two consecutive samples of a smooth singular vector can be as close as desired and we can approximate

$$\Re\left\{c_{ij}^u[k]\right\}\approx|c_{ij}^u[k]|=|c_{ij}^{u'}[k]|,$$

which allows us to use inner product of $\vec{\mathbf{u}}'[k]$ instead of $\vec{\mathbf{u}}[k]$. From (12) and (13), it can be seen that before the intersection of eigenvalues, consecutive eigenvectors which are sorted by conventional EVD in decreasing order, are from the same smooth eigenvector and so $|c_{11}^{u'}[k]|$ and $|c_{22}^{u'}[k]|$ are near unity. However, if $k-1$ and k are two frequency sample before and after the intersection, respectively, due to decreasing order of eigenvalues, smoothed eigenvectors are swapped after intersection. Therefore, $|c_{11}^{u'}[k]|$ and $|c_{22}^{u'}[k]|$ are some values near zero, instead $|c_{12}^{u'}[k]|$ and $|c_{21}^{u'}[k]|$ are near unity.

Algorithm 2 describes a simple rearrangement procedure to track eigenvectors (singular vectors) for smooth decomposition.

4 Finite duration constraint

Phase alignment is critical to have compact order decomposition. Another aspect of this fact is revealed in the coefficient's domain perspective of (7). In this domain, the multiplication is replaced by circular convolution

$$A[\ ((n))_K]=U'[\ ((n))_K]\circledast\Sigma'[\ ((n))_K]\circledast V'^H[\ ((-n))_K]$$
$$=U[\ ((n))_K]\circledast\Sigma[\ ((n))_K]\circledast V^H[\ ((-n))_K] \tag{16}$$

in which $\circledast$ is the circular convolution operator and $((n))_K$ denotes n module K.

Polynomial SVD corresponds to linear convolution in the coefficients domain, however the decomposition obtained from DFT corresponds to circular convolution. Recalling from discrete-time signal processing, it is well known that we can equivalently utilize circular convolution instead of linear convolution if convoluted signals are zero-padded adequately. That is, for $x_1[n]$ and $x_2[2]$ are two signals with the length of N_1 and N_2, respectively, apply zero padding such that zero padded signals have the length $N_1 + N_2 - 1$ [21]. Hence, if the last $M - 1$ coefficients of $U[n]$, $\Sigma[n]$, and $V[n]$, are zero, the following results are hold:

$$A[((n))_K] = U[((n))_K] \circledast \Sigma[((n))_K] \circledast V^H[((-n))_K] \Rightarrow A[n] = U[n] * \Sigma[n] * V^H[-n],$$

$$U[((n))_K] \circledast U^H[((-n))_K] = \delta[((n))_K] I \Rightarrow U[n] * U^H[-n] = \delta[n] I, \qquad (17)$$

$$V[((n))_K] \circledast V^H[((-n))_K] = \delta[((n))_K] I \Rightarrow V[n] * V^H[-n] = \delta[n] I.$$

Therefore, the problem is to obtain the phase set $\{\vec{\theta}_i[k]\}$ and correcting the singular vectors using (9). The phase set $\{\vec{\theta}_i[k]\}$ should be such that the resulting coefficients satisfy (17).

Without loss of generality, let $U[n]$ and $V[n]$ be causal, i.e., $U[n] = V[n] = 0$ for $n < 0$. $U[n]$ and $V[n]$ (which are supposed to be of length M) should be multi-sequence zero-padded at least with $(M - 1)$ zeros.

$$U[n] = \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{U}[k] w_K^{-kn} = \mathbf{0} \qquad (18)$$

for $n = M, M + 1, \ldots, K - 1$, in which $K \geq 2M - 1$. If these conditions are satisfied, circular convolution can be used instead of linear convolution.

Since the available matrix of singular vectors at each frequency is $\mathbf{U}'[k]$, inserting (9) in (18) for each singular vector separately leads to

$$\sum_{k=0}^{K-1} \mathbf{u}'_i[k] e^{-j\theta_i^u[k]} w_K^{-kn} = \mathbf{0} \qquad (19)$$

for $n = M, M + 1, \ldots, K - 1$.

Without loss of generality, let $\theta_i[0] = 0$. In a more compact form we can express these $(K - M)$-folded equations in matrix form

$$\mathcal{F}_M(\vec{\mathbf{u}}'_i) \vec{\mathbf{x}}(\vec{\theta}_i^u) = -\vec{f}_M(\vec{\mathbf{u}}'_i) \qquad i = 1, 2, \ldots, r \qquad (20)$$

in which $\vec{\mathbf{x}}(\vec{\theta}_i^u) = [\exp(j\theta_i^u[1]), \exp(j\theta_i^u[2]), \ldots, \exp(j\theta_i^u[K-1])]^T$, $\vec{f}_M(\vec{\mathbf{u}}'_i) = [\vec{\mathbf{u}}'^T_i[0], \vec{\mathbf{u}}'^T_i[0], \ldots, \vec{\mathbf{u}}'^T_i[0]]^T$ is a $p(K - M) \times 1$ vector, and

$$\mathcal{F}_M(\vec{\mathbf{u}}'_i) =$$
$$\begin{bmatrix} \vec{\mathbf{u}}'_i[1] w_K^{-M} & \vec{\mathbf{u}}'_i[2] w_K^{-2M} & \cdots & \vec{\mathbf{u}}'_i[K-1] w_K^{-(K-1)M} \\ \vec{\mathbf{u}}'_i[1] w_K^{-(M+1)} & \vec{\mathbf{u}}'_i[2] w_K^{-2(M+1)} & \cdots & \vec{\mathbf{u}}'_i[K-1] w_K^{-(K-1)(M+1)} \\ \vdots & \vdots & & \vdots \\ \vec{\mathbf{u}}'_i[1] w_K^{-(K-1)} & \vec{\mathbf{u}}'_i[2] w_K^{-2(K-1)} & \cdots & \vec{\mathbf{u}}'_i[K-1] w_K^{-(K-1)^2} \end{bmatrix}.$$

For polynomial EVD, Equation (20) is enough, however, for polynomial SVD we have two options. To approximate SVD with approximately positive singular values, we must

augment $\mathcal{F}_M(\vec{\mathbf{u}}'_i)$ and $\vec{f}_M(\vec{\mathbf{u}}'_i)$ with similar defined matrix and vector for $\mathbf{v}'_i[k]$

$$\mathcal{F}_M(\vec{\mathbf{u}}'_i, \vec{\mathbf{v}}'_i) = \begin{bmatrix} \mathcal{F}_M(\vec{\mathbf{u}}'_i) \\ \mathcal{F}_M(\vec{\mathbf{v}}'_i) \end{bmatrix} \text{ and } \vec{f}_M(\vec{\mathbf{u}}'_i, \vec{\mathbf{v}}'_i) = \begin{bmatrix} \vec{f}_M(\vec{\mathbf{u}}'_i) \\ \vec{f}_M(\vec{\mathbf{v}}'_i) \end{bmatrix},$$

then solve

$$\mathcal{F}_M(\vec{\mathbf{u}}'_i, \vec{\mathbf{v}}'_i) \vec{\mathbf{x}}(\vec{\theta}_i) = -\vec{f}_M(\vec{\mathbf{u}}'_i, \vec{\mathbf{v}}'_i) \quad i = 1, 2, \ldots, r. \qquad (21)$$

An additional degree of freedom is obtained by letting singular values to be complex. However, an straightforward solution which yield to singular values and singular vectors of order M is complicated. Instead, we impose the finite duration constraint only two singular vectors

$$\begin{cases} \mathcal{F}_M(\vec{\mathbf{u}}'_i) \vec{\mathbf{x}}(\vec{\theta}_i^v) = -\vec{f}_M(\vec{\mathbf{u}}'_i) \\ \mathcal{F}_M(\vec{\mathbf{v}}'_i) \vec{\mathbf{x}}(\vec{\theta}_i^v) = -\vec{f}_M(\vec{\mathbf{v}}'_i) \end{cases} \quad i = 1, 2, \ldots, r. \qquad (22)$$

If $K \geq 2M - 1$, then the last $M - 1$ coefficients of resulting polynomial vectors are zero. Therefore, according to (17), $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are paraunitary. On the other hand, if $K \geq 2M + N_{\max} - N_{\min} - 1$, circular convolution relation of coefficient

$$\Sigma[((n))_K] = U^H[((-n))_K] \circledast A[((n))_K] \circledast V[((n))_K]$$

results in the linear convolution $\Sigma[n] = U^H[-n] * A[n] * V[n]$. This guarantee that $\mathbf{\Sigma}(z)$ is a diagonal polynomial matrix of order $2M + N_{\max} - N_{\min} - 2$. Obviously, if $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are paraunitary and $\mathbf{\Sigma}(z) = \tilde{\mathbf{U}}(z) \mathbf{A}(z) \mathbf{V}(z)$ is a diagonal matrix, $\mathbf{A}(z) = \mathbf{U}(z) \mathbf{\Sigma}(z) \tilde{\mathbf{V}}(z)$ is the polynomial SVD of $\mathbf{A}(z)$.

Once the set of phase $\{\theta_i^u[k], \theta_i^v[k]\}$ are obtained from (20), (21), or (22), phase alignment of $\vec{\mathbf{u}}'_i[k]$ and $\vec{\mathbf{v}}'_i[k]$ can be done using (10) and inverse DFT of $\mathbf{U}[k]$ and $\mathbf{V}[k]$ yield to coefficient matrices $U[n]$ and $V[n]$. For obtaining singular values, we have two options, we can either set $K \geq 2M - 1$ and phase align $\boldsymbol{\sigma}'_i[k]$ using (10). After inverse DFT of $\mathbf{\Sigma}[k]$, we should truncate $\Sigma[n]$ to have duration

M. Another option which yields to more accurate results is by calculating $\tilde{\mathbf{U}}(z)\mathbf{A}(z)\mathbf{V}(z)$ and replacing off-diagonal elements with zero.

Next, we provide a minimization approach to determine the unknown set $\{\theta_i[k]\}$.

5 Gradient descent solution

In general, there may exist no phase vector $\vec{\theta}$ which satisfies (20). Even when there exists a phase vector that satisfies the finite duration constraint, the solution is not straightforward. For these reasons, we can view (20) as a minimization problem [6]. We use energy of the highest order coefficients (the coefficients that we equate to zero in (18)) as the objective to the minimization problem

$$J(\vec{\theta}_i) = \left\| \mathcal{F}_M(\vec{\mathbf{u}}_i')\vec{\mathbf{x}}(\vec{\theta}_i^u) + \vec{f}_M(\vec{\mathbf{u}}_i') \right\|^2, \quad i = 1, 2, \ldots, r. \tag{23}$$

An alternative minimization technique as a solution for this phase optimization problem is proposed in [6], which we describe it in this section.

Throughout this section, we focus on solving $\vec{\theta}_i = \arg\min J(\vec{\theta}_i)$ as a least square solution for a single singular vector $\vec{\mathbf{u}}_i[k]$, so we drop the subscript "*i*" from the quantity $\vec{\theta}_i$ and use $\mathcal{F}$ and $\vec{f}$, instead of $\mathcal{F}_M(\vec{\mathbf{u}}_i')$ and $\vec{f}_M(\vec{\mathbf{u}}_i')$ to simplify the notation. The objective $J(\vec{\theta})$ is intentionally presented as a function of $\vec{\theta}$ to emphasize the fact that our problem is classified as an unconstrained optimization.

We exploit the trusted region strategy for the problem (23). By utilizing the information about the gradient vector and Hessian matrix in each step, trusted region strategy constructs a model function m_k which have a similar behavior close to the current point $\vec{\theta}^{(k)}$. The model m_k is usually defined as the second-order Taylor series expansion (or its approximation) of $J(\vec{\theta} + \vec{\varphi})$ around $\vec{\theta}$, that is

$$m_k(\vec{\varphi}) = J(\vec{\theta}) + \vec{\varphi}^T \nabla J(\vec{\theta}) + \vec{\varphi}^T \nabla^2 J(\vec{\theta})\vec{\varphi},$$

where $\nabla J(\vec{\theta})$ and $\nabla^2 J(\vec{\theta})$ are the gradient vector and the Hessian matrix corresponding to $J(\vec{\theta})$, respectively. The model m_k is designed to be a good approximation of $J(\vec{\theta})$ near the current point and is not trustworthy on regions far from the current point. Consequently, the restriction in minimization of m_k on a region around $\vec{\theta}^{(k)}$ is crucial, that is

$$\vec{\varphi} = \arg\min_{\vec{\varphi}} m_k(\vec{\varphi}) \quad \|\vec{\varphi}\| < R. \tag{24}$$

where *R* is the trusted region radius.

The decision about shrinking of the trusted region is determined by comparing the actual reduction inobjective

function and predicted reduction. Given a step $\vec{\varphi}$, the ratio

$$\rho = \frac{J(\vec{\theta}) - J(\vec{\theta} + \vec{\varphi})}{m_k(0) - m_k(\vec{\varphi})} \tag{25}$$

is used as a criterion to indicate if the trusted region is small enough.

Among methods which approximate the solution of the constrained minimization (24) dogleg procedure is the only one which leads to analytical approximation. It also promises to achieve at least as much reduction in m_k as is possible by Cauchy point (the minimizer of m_k along the steepest descent direction $-\nabla J(\vec{\theta})$, subject to the trusted region) [22]. However, this procedure requires Hessian matrix (or an approximation of it) to be positive definite.

5.1 Hessian matrix modification

The gradient vector and Hessian matrix corresponding to $J(\vec{\theta})$ are as follows

$$\begin{aligned} \vec{g}(\vec{\theta}) &= 2\Im\left\{\mathbf{X}(\vec{\theta})\mathcal{F}^H(\mathcal{F}\vec{\mathbf{x}}(\vec{\theta}) + \vec{f})\right\}, \\ H(\vec{\theta}) &= 2\Re\left\{\mathbf{X}(\vec{\theta})\mathcal{F}^H\mathcal{F}\mathbf{X}(\vec{\theta})^H\right\} \\ &\quad -2\Re\left\{\text{diag}\left(\mathbf{X}(\vec{\theta})\mathcal{F}^H(\mathcal{F}\vec{\mathbf{x}}(\vec{\theta}) + \vec{f})\right)\right\}, \end{aligned} \tag{26}$$

where $\mathbf{X}(\vec{\theta})$ is a diagonal matrix with the *k*th diagonal element $\exp(-j\theta[k])$ and $k = 0, 1, \ldots, K-1$.

In general, Hessian matrix in (26) does not promise to be always positive definite. Therefore, we should modify Hessian matrix to yield a positive definite approximation.

We provide a simple modification which brings some desirable features by omitting the second term from the Hessian matrix and diagonal loading to guarantee positive definiteness

$$H(\vec{\theta}) \approx 2\Re\left\{\mathbf{X}(\vec{\theta})\mathcal{F}^H\mathcal{F}\mathbf{X}(\vec{\theta})^H\right\} + \alpha I. \tag{27}$$

The term $2\Re\left\{\mathbf{X}(\vec{\theta})\mathcal{F}^H\mathcal{F}\mathbf{X}(\vec{\theta})^H\right\}$ is positive semi-definite and in many situations, it is much more significant than the second term of Hessian matrix in (26). Hence, with diagonal loading αI (*I* is with conformable size and α is very small), the modified Hessian matrix guarantees (27) to be positive definite and provides the desired properties in contrast to use the exact Hessian matrix.

5.2 Dogleg method

Dogleg method starts with the unconstrained minimization of (24)

$$\phi^h = -H^{-1}\vec{g} \tag{28}$$

When the trusted region radius is so large that $\|\phi^H\| \leq R$, it is the exact solution of (24) and we select it as the dogleg method answer. On the other hand, for small *R* the solution of (24) is $-R\vec{g}/\|\vec{g}\|$. For intermediate values of *R*, the optimal solution lies on a curved trajectory between these two points [22].

Algorithm 3 Trusted region dogleg algorithm

```
Obtain θ⃗₀ from (33)
Given R₀ < R̄
for  i = 0,1,2,⋯
    Obtain φ⃗ₕ from (28)
    if  ‖φ⃗ʰ‖ < R₀
        φ⃗ᵢ ← φ⃗ᵘ
    else
        Obtain φ⃗ᵍ from (29)
        if  ‖φ⃗ᵍ‖ > Rᵢ
            φ⃗ᵢ ← Rᵢ φ⃗ᵍ/‖φ⃗ᵍ‖
        else
            φ⃗ᵈ ← φ⃗ʰ − φ⃗ᵍ
            Solve ‖φ⃗ᵈ‖τ²+2(φ⃗ᵍ)ᴴφ⃗ᵈτ+‖φ⃗ᵍ‖−Rᵢ²=0
            Select the root τ > 0
            φ⃗ᵢ ← φ⃗ᵍ + τφ⃗ʰ
    Evaluate ρᵢ from (25)
    if  ρᵢ < ¼
        Rᵢ₊₁ ← 0.25Rᵢ
    elseif  ρ > ¾ and ‖φ⃗ᵢ‖ = Rᵢ
        Rᵢ₊₁ ← min(2Rᵢ,R̄)
    else
        Rᵢ₊₁ ← Rᵢ
    if  ρᵢ > 0
        θᵢ₊₁ ← θᵢ + φᵢ
    else
        θᵢ₊₁ ← θᵢ
end(for)
```

The dogleg method approximates this trajectory by a path consisting of two line segments. The first line segment starts from the origin to the unconstrained minimized point along the steepest descent direction

$$\phi^g = -\frac{\vec{g}^T\vec{g}}{\vec{g}^T H\vec{g}}\vec{g}. \tag{29}$$

The second line segment starts from ϕ^g to ϕ^h. These two line segments form an approximate trajectory which its intersection with the sphere $\|\phi\| = R$ is the approximate solution of (24) when $\|\phi^h\| > R$.

5.3 Alternating minimization
Another solution of (23) is provided by converting the problem of multivariate minimization to a sequence of single-variate minimization problem via alternating minimization [6]. In each iteration, a series of single-variate minimization is performed, while other parameters are held unchanged. Each Iteration consists of $K-1$ steps,

which at each step one parameter $\theta[k]$ is updated. Suppose we are at step k of ith iteration. At this step $k-1$ first parameters were updated in the current iteration, and $K-k-2$ last parameters were updated from the previous iteration. These parameters are held fixed, while $\theta[k]$ is minimized at the current step,

$$\theta_i[k] = \arg\min_{\theta[k]} J\left(\theta_i[1],...,\theta_i[k-1],\theta[k],\theta_{i-1}[k+1],...,\theta_{i-1}[K-1]\right). \tag{30}$$

The cost function is guaranteed to be non-incremental at each step; however, this method is also converges to a local minima which highly depend on the initial guess of the algorithm. For solving (30) it is suffices to make the kth element of gradient vector in (26) equal to zero. Suppose the calculation are performed for phase alignment of $\vec{u}'[k]$ $k = 0, 1, \ldots, K-1$,

$$\frac{\partial J}{\partial\theta[k]} = \Im\left\{e^{-j\theta[k]}t_i[k]\right\}$$
$$= 2\,|t_i[k]|^2\sin(\angle t_i[k]-\theta[k]) = 0, \tag{31}$$

where $\angle t_u^{(i)}[k]$ is the phase angle of $t_u^{(i)}[k]$ and

$$t_i[k] = \sum_{l=0}^{k-1}e^{j\theta_i[l]}\vec{u}'^H[k]\,\vec{u}'[l]\,\frac{w_K^{(k-l)M}-1}{1-w_K^{(k-l)}}$$
$$+ \sum_{l=k+1}^{K-1}e^{j\theta_{i-1}[l]}\vec{u}'^H[k]\,\vec{u}'[l]\,\frac{w_K^{(k-l)M}-1}{1-w_K^{(k-l)}}.$$

Fortunately, Equation (31) has a closed form solution

$$\theta_i[k] = \begin{cases} \angle t_i[k] \\ \angle t_i[k]+\pi \end{cases}. \tag{32}$$

However, only the second case of (32) has positive second partial deviation. Therefore, the global minima of (30) is

$$\theta_i[k] = \angle t_i[k]+\pi.$$

5.4 Initial guess
All algorithms of unconstrained minimization require to be supplied by a starting point, which we denoted by $\vec{\theta}_0$. To avoid getting stuck in local minima, we should select a good initial guess. This can be accomplished by minimizing a different but similar cost function denoted by $J'(\vec{\theta})$

$$J'(\vec{\theta}) = \left\|\vec{x}(\vec{\theta}_i) - \mathcal{F}^\dagger\vec{f}\right\|^2$$

in which † represents pseudo inverse.

Solving $J'(\vec{\theta})$ yields to a simple initial guess

$$\vec{\theta}_0 = \angle\mathcal{F}^\dagger\vec{f}. \tag{33}$$

Based on what have been mentioned in this section, a pseudo-code description of the trusted region dogleg algorithm is given by Algorithm 3. In this algorithm,

we start with the initial guess of (33) and a trusted region radius upper bound $\bar{R}$. Then we continue the trusted region minimization procedure as described in this section.

6 Simulation results

In this section, we present some examples to demonstrate the performance of the proposed algorithm. For the first example, our algorithm is applied to a polynomial matrix example from [11]

$$\mathbf{A}(z) = \begin{bmatrix} 2 & 0 & 2z^{+1} \\ z^{+1} & 1 & 0 \\ 0 & z^{-1} & 1 \end{bmatrix}. \tag{34}$$

Frequency behavior of singular values can be seen in Figure 1. There is no intersection of singular values, so the setup of the algorithm either for spectral majorization or frequency smoothness leads to identical decomposition.

For having approximately positive singular values, we use (21). Define the average energy of highest order coefficients for the pair of polynomial singular vectors $\vec{\mathbf{u}}_i$ and $\vec{\mathbf{v}}_i$ as $E_i^{u,v} = J(\vec{\theta}_i)/(K - M)$ (we expect energy of highest order coefficients to be zero or at least minimized). A plot of E_i versus iteration for each pair of singular vectors is depicted in Figure 2. The decomposition length is $M = 9$ (order is 8) and we use $K = 2M + (N_{\max} - N_{\min}) = 20$ number of DFT points.

As it is seen, the use of dogleg method with approximate Hessian matrix leads to a fast convergence in contrast with using alternative minimization and Cauchy-point (which is always selected along the gradient direction). Of course we should consider that due to matrix inversion, computational complexity of Dogleg method is $O(K^3)$ while computational complexity of alternative minimization and Cauchy point is $O(K^2)$.
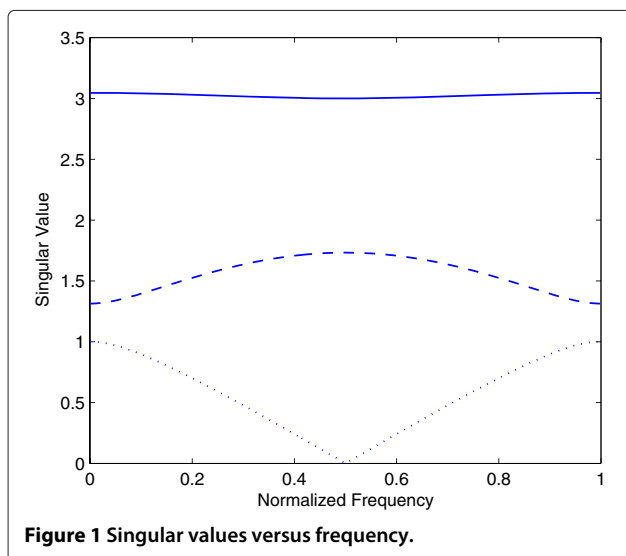
The final value of average highest order coefficient for three pair of singular vectors are 5.54×10^{-5}, 3.5×10^{-3}, and 0.43, respectively. The first singular vector satisfies finite duration constraint almost exactly. The second singular vector fairly satisfies this constraint. However, highest order coefficients of last singular vector, possess considerable amount of energy, that seems to cause decomposition error.

Denote the relative error of the decomposition as

$$E^A = \frac{\left\| \mathbf{A}(z) - \mathbf{U}(z)\Sigma(z)\tilde{\mathbf{V}}(z) \right\|_F}{\|\mathbf{A}(z)\|_F}$$

in which $\|\cdot\|_F$ is the extension of Frobenius norm for polynomial matrices and is defined by

$$\|\mathbf{A}(z)\|_F = \sqrt{\sum_n \|A[n]\|_F^2}.$$

Since in our optimization procedure we only seek for finite duration approximation, $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are only approximately paraunitary. Therefore, we also define relative error of paraunitarity as

$$E^U = \frac{\left\| \tilde{\mathbf{U}}(z)\mathbf{U}(z) - \mathbf{I} \right\|_F}{r}.$$

An upper bound for E^U can be obtained as

$$E^U \leq 2\sqrt{\frac{K-M}{K}\sum_{i=1}^{r} E_i^{(u)}(1 - E_i^{(u)})}$$
$$+ \sqrt{\frac{K-M}{K}\sum_{i=1}^{r}(E_i^{(u)})^2},$$

which means as average energy on $K - M$ highest order goes to zero, E^U diminishes.

The relative error of this decomposition is $E^A = 1.18 \times 10^{-2}$ while the error of $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are $E^U = 3.3 \times 10^{-2}$ and $E^V = 3.08 \times 10^{-2}$, respectively. The paraunitarity error is relatively high in contrast with decomposition error. This is due to the difference between the first two singular values and the last singular value.

A plot of relative errors E^A, E^U, and E^V for various amount of M is shown in Figure 3. The number of frequency samples is fixed at $K = 2M + 2(N_{\max} - N_{\min})$.

The number of frequency samples K is an optional choice, however as discussed in Section 4, it should satisfy $K \geq 2M + N_{\max} - N_{\min} - 1$. In order to demonstrate the effect of number of frequency samples on the decomposition error, a plot of relative error versus different amount of K is depicted in Figure 4. Increasing the number of frequency samples does not lead to reduction of relative error. Moreover, it increases computational burden. Therefore, a value near $2M + (N_{\max} - N_{\min}) - 1$ is a reasonable choice for the number of frequency samples.
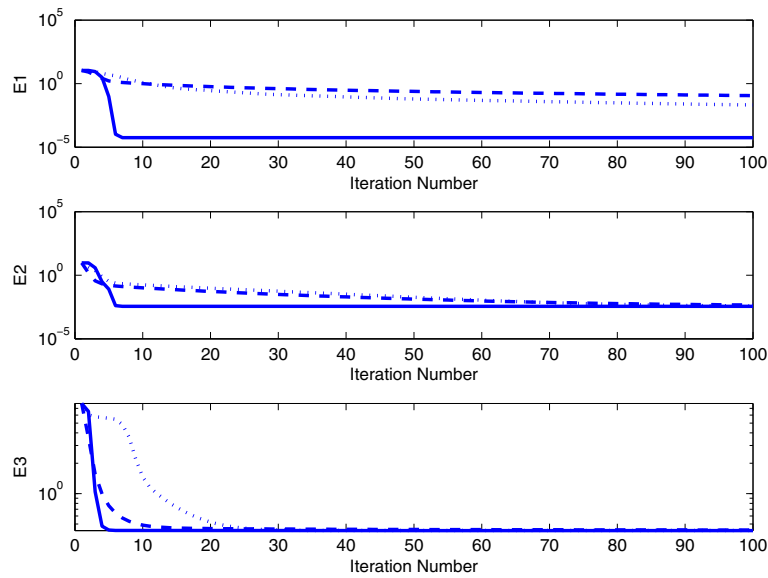

Figure 1 Singular values versus frequency.
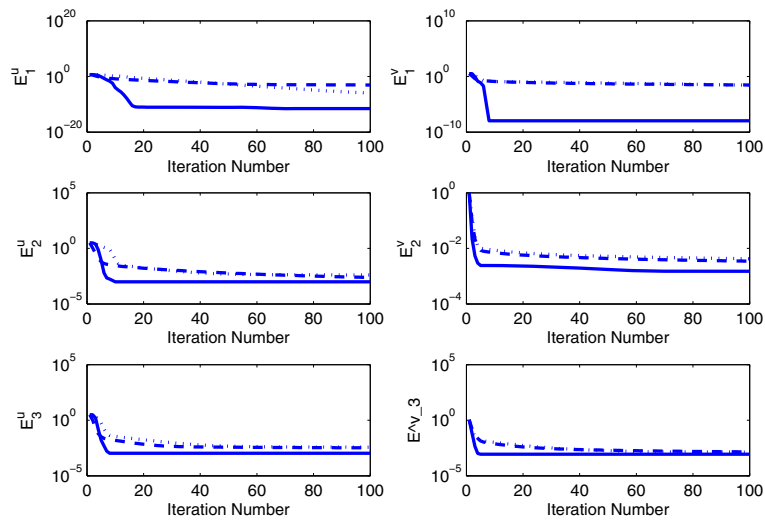
Figure 2 Average highest order coefficients energy E_i versus iteration number for a decomposition with approximately positive singular values. Dotted line: Cauchy points. Dashed line: Alternative minimization. Solid Line: proposed algorithm.

Now, lets relax the problem by allowing singular values to be complex and using (22). A plot of E_i^u and E_i^v versus iteration for each pair of singular vectors is depicted in Figure 5. The decomposition length is $M = 9$ (order is 8) and we use $K = 2M + (N_{max} - N_{min}) = 20$ number of DFT points.

Again Dogleg method converges very rapidly while alternative minimization and Cauchy point converge slowly. The final value of average energy for three left singular vectors are 1.23×10^{-10}, 9.7×10^{-4}, and 10^{-3},

respectively. This is while these values for right singular vectors are 1.12×10^{-10}, 1.4×10^{-3}, and 8.7^{-4}, respectively.

Note that the average energy of highest order coefficients for the third pair of singular vectors alleviate meaningfully. Figure 1 shows that the third singular value goes to zero and then returns to positive values. If we constrain singular values to be positive, a phase jump of π radian, is imposed to one of third singular vectors near the frequency which singular vector goes to zero. However, by
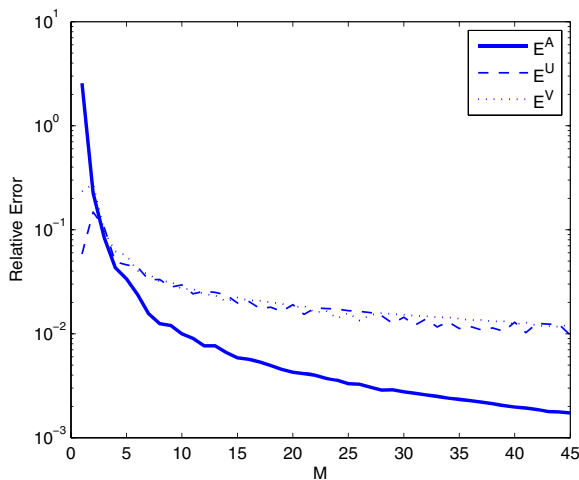
Figure 3 Relative error versus *M* for a decomposition with approximately positive singular values. $K = 2M = 2$.
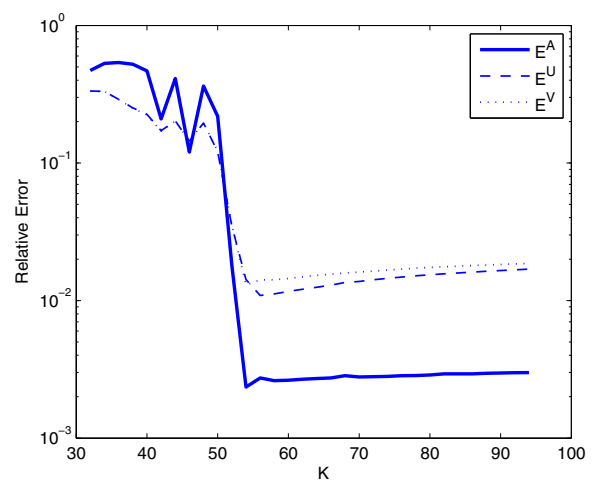
Figure 4 Relative error versus *K* for a decomposition with approximately positive singular values. $M = 31$.

Figure 5 Average highest order coefficients energy E_i versus iteration number for a decomposition with complex singular values. Dotted line: Cauchy points. Dashed line: Alternative minimization. Solid Line: proposed algorithm.

letting singular values to be complex, the zero crossing occur which requires no discontinuity of singular vectors.

The relative error of this decomposition is $E^A = 4.9 \times 10^{-3}$ while the error of $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are $E^U = 2.5 \times 10^{-3}$ and $E^V = 3.5 \times 10^{-3}$, respectively. In contrast with constraining singular values to be positive, having complex singular values decrease decomposition and paraunitarity error significantly.

Plots of relative errors E^A, E^U, and E^V for various amount of M and K are shown in Figures 6 and 7, respectively. Letting singular values be complex causes

significant reduction of all relative errors. As it was mentioned, Figure 7 shows that increasing K from $2M + N_{\max} - N_{\min} - 1$ causes no improvement in relative errors while it makes additional computational burden.

McWhirter and coauthors [11] have reported the relative error of decomposition. Provided that paraunitary matrices $\mathbf{U}(z)$ and $\mathbf{V}(z)$ are of order 33, the relative error of their algorithm is 0.0469. This is while our algorithm only requires paraunitary matrices of order 3 for relative error of 0.035 with positive singular values and relative error of 2.45×10^{-6} with complex singular values. In addition,
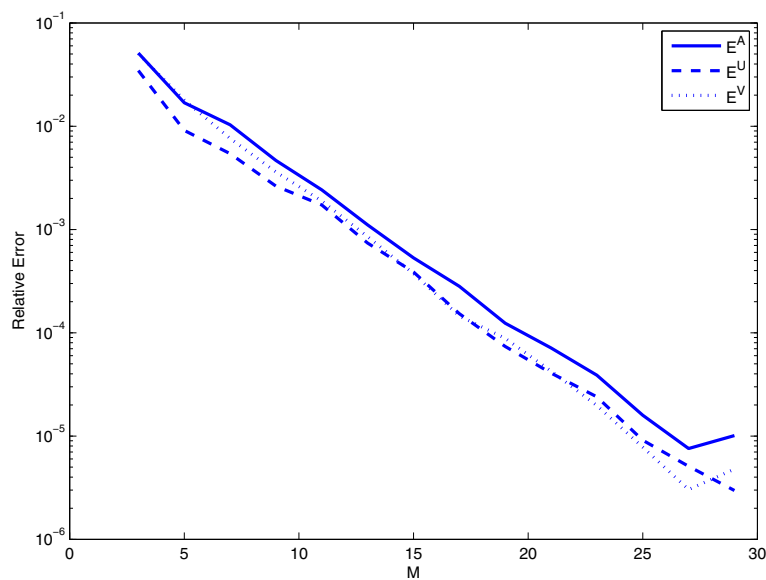
Figure 6 Relative error versus M for a decomposition with complex singular values. $K = 2M + 2$.
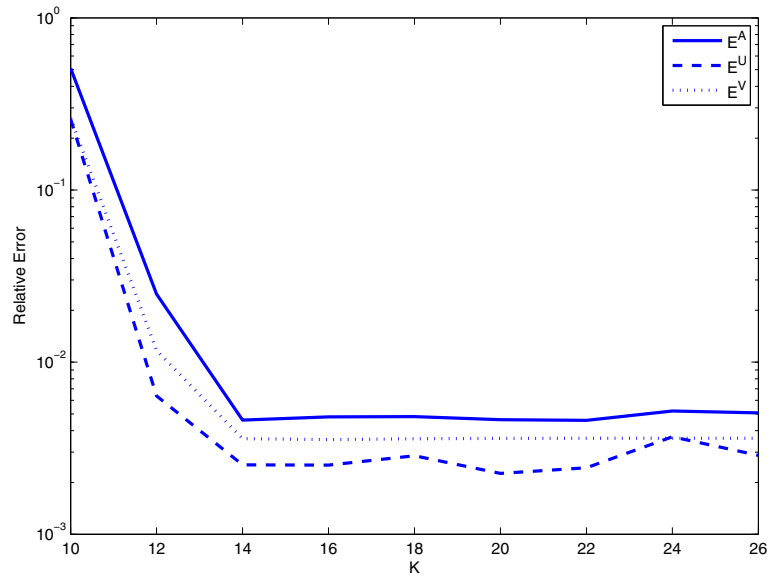
Figure 7 Relative error versus K for a decomposition with complex singular values. $M = 9$.

in the new approach, exploiting paraunitary matrices of order 33, the relative error is 0.0032 with positive singular values and 4.7×10^{-6} with complex singular values.

This large difference is not caused by iteration numbers because we compare results while all algorithms relatively converges, and with continuation of iterations trivial improvement are obtained. The main reason lies on different constraints of the solution presented in [11] in contrast to our proposed method. While they impose paraunitary constraint on $\tilde{\mathbf{U}}(z)\mathbf{A}(z)\mathbf{V}(z)$ to yield a diagonalized $\Sigma(z)$, we impose the finite duration constraint and obtain approximation of $\mathbf{U}(z)$ and $\mathbf{V}(z)$ with fair fitting to the decomposed matrices at each frequency samples. Therefore, we can consider this method as a finite duration polynomial regression of matrices which is obtained by uniformly sampling $\mathbf{U}(z)$ and $\mathbf{V}(z)$ on the unit circle in z-plane.

As a second example, consider EVD of the following para-Hermitian matrix

$$\mathbf{A}(z) = \begin{bmatrix} .5z^2 + 3 + .5z^{-2} & -.5z^2 + .5z^{-2} \\ .5z^2 - .5z^{-2} & -.5z^2 + 1 - .5z^{-2} \end{bmatrix}$$

The exact smooth EVD of this matrix is of finite order

$$\mathbf{U}(z) = \frac{1}{2} \begin{bmatrix} 1 + z^{-1} & 1 - z^{-1} \\ 1 - z^{-1} & 1 + z^{-1} \end{bmatrix}, \tag{35}$$

$$\Lambda(z) = \begin{bmatrix} z^1 + 2 + z^{-1} & 0 \\ 0 & -z^1 + 2 - z^{-1} \end{bmatrix}.$$

Frequency behavior of eigenvalues can be seen in Figure 8. Since eigenvalues intersect at two frequencies,

smooth decomposition and spectrally majorized decomposition result two distinct solutions.

To perform smooth decomposition, we need to track and rearrange eigenvectors to avoid any discontinuity using Algorithm 2. The resulting $|c_{ij}^{u'}[k]|$ are shown in Figure 9 for $k = 0, 1, \ldots, K - 1$. Using these $|c_{ij}^{u'}[k]|$ the Algorithm 2 swap first and second eigenvalues and eigenvectors for $k = 12 : 32$ which results in continuity of eigenvalues and eigenvectors.

Now that all eigenvalues and eigenvectors are rearranged in DFT domain, it's time for phase alignment of
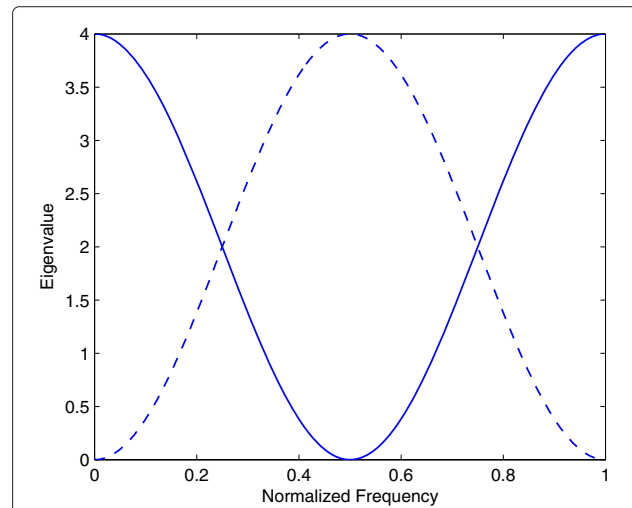

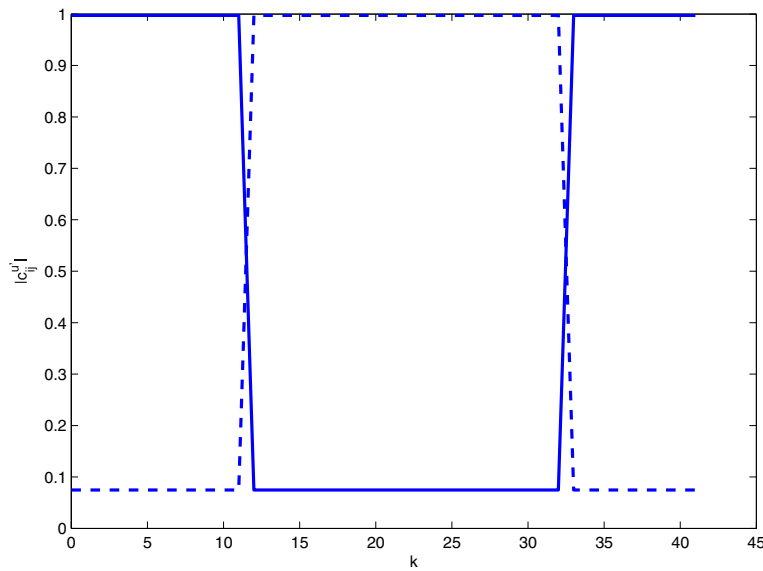**Figure 8 Eigenvalues of smooth decomposition versus frequency.**

Figure 9 Rearrangement of eigenvalues and eigenvectors. $K = 42$. Dashed Line: $c_{12}^{u'}[k]$ and $c_{21}^{u'}[k]$. Solid Line: $c_{11}^{u'}[k]$ and $c_{22}^{u'}[k]$.

eigenvectors. A plot of E_i versus iteration for $M = 3$ and smooth decomposition is depicted in Figure 10. It is predictable that dogleg algorithm converges rapidly while the alternative minimization and Cauchy point has a long way to converge.

Since the energy of highest order coefficients of eigenvectors are trifling, using the proposed method for smooth decomposition results in very high accuracy, as

seem in the figures. Relative error of smooth decomposition versus M is shown in Figure 11.

While using frequency smooth EVD of (35) leads to relative error below 10^{-5} for $M \geq 3$ with a few number of iterations, Spectrally majorized EVD requires a lot more polynomial order to reach a reasonable relative error.

Unlike smooth decomposition which requires rearrangement of eigenvalues and eigenvectors, spectral
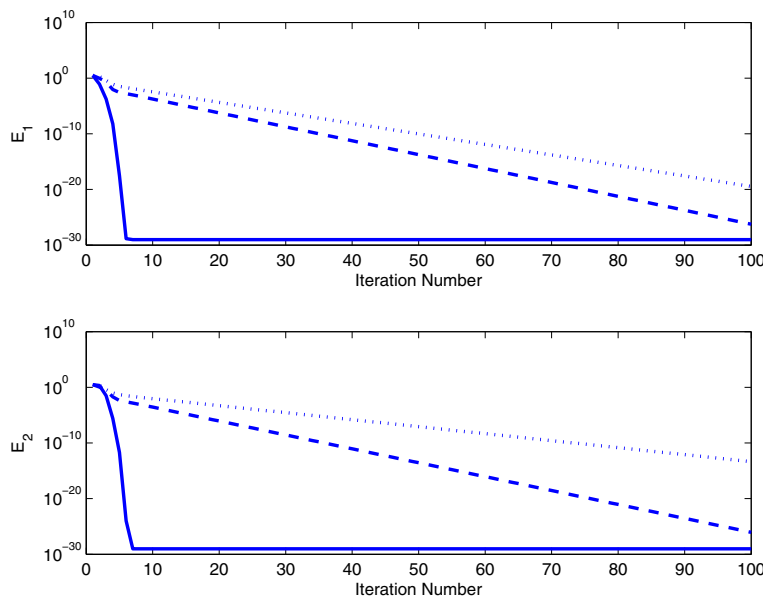
Figure 10 E_i **versus iteration number corresponding to smooth decomposition.** Dotted line: Cauchy points. Dashed line: Alternative minimization. Solid Line: proposed algorithm.
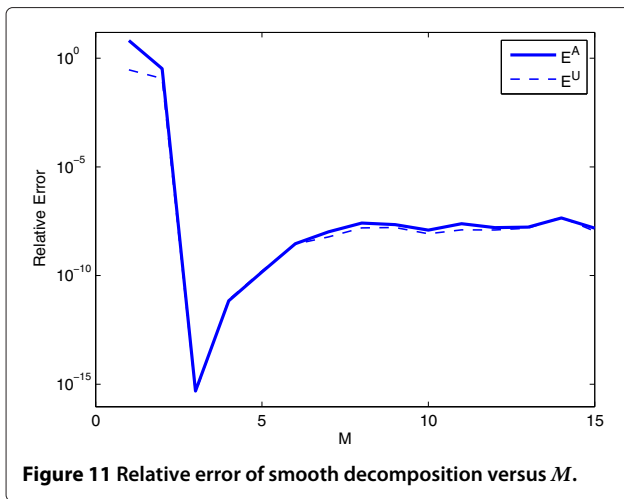
Figure 11 Relative error of smooth decomposition versus *M*.

majorization requires only to sort eigenvalues at each frequency sample in decreasing order. Most of conventional EVD algorithm sort eigenvalues in decreasing order, which we should only align eigenvector phases using 3. A plot of E_i versus iteration for $M = 20$ and spectrally majorized decomposition is depicted in Figure 12.

Due to an abrupt change in eigenvectors at the intersection frequency of eigenvalues, increasing the decomposition order leads to a slow decay of relative error. Figure 13 shows the relative error as a function of M.

To see the difference between smooth and spectrally majorized decomposition results, eigenvalues of spectrally majorized decomposition is shown in Figure 14, which is comparable with Figure 8 which corresponds to eigenvalues of smooth decomposition. Therefore, a low order polynomial is required using smooth decomposition and much higher polynomial order for spectrally majorized decomposition. Even with $M = 20$ the decomposition have relatively high error.

7 Conclusion

An algorithm for polynomial EVD and SVD based on DFT formulation has been presented. One of the advantages of the DFT formulation is that it enables us to control properties of decomposition. Among these properties, we introduce how to setup the decomposition to achieve spectrally majorization and frequency smoothness. We have shown, if singular values (eigenvalues) intersect at some frequency, then simultaneous achievement of spectral majorization and smooth decomposition is not possible. In this situation, setting up the decomposition to possess spectral majorization requires considerably higher order polynomial decomposition and more computational complexity. Highest order polynomial coefficients of singular vectors (eigenvectors) are utilized as square error to obtain a compact decomposition based on phase alignment of frequency samples. The algorithm has the flexibility to compute a decomposition with approximately positive singular values, and
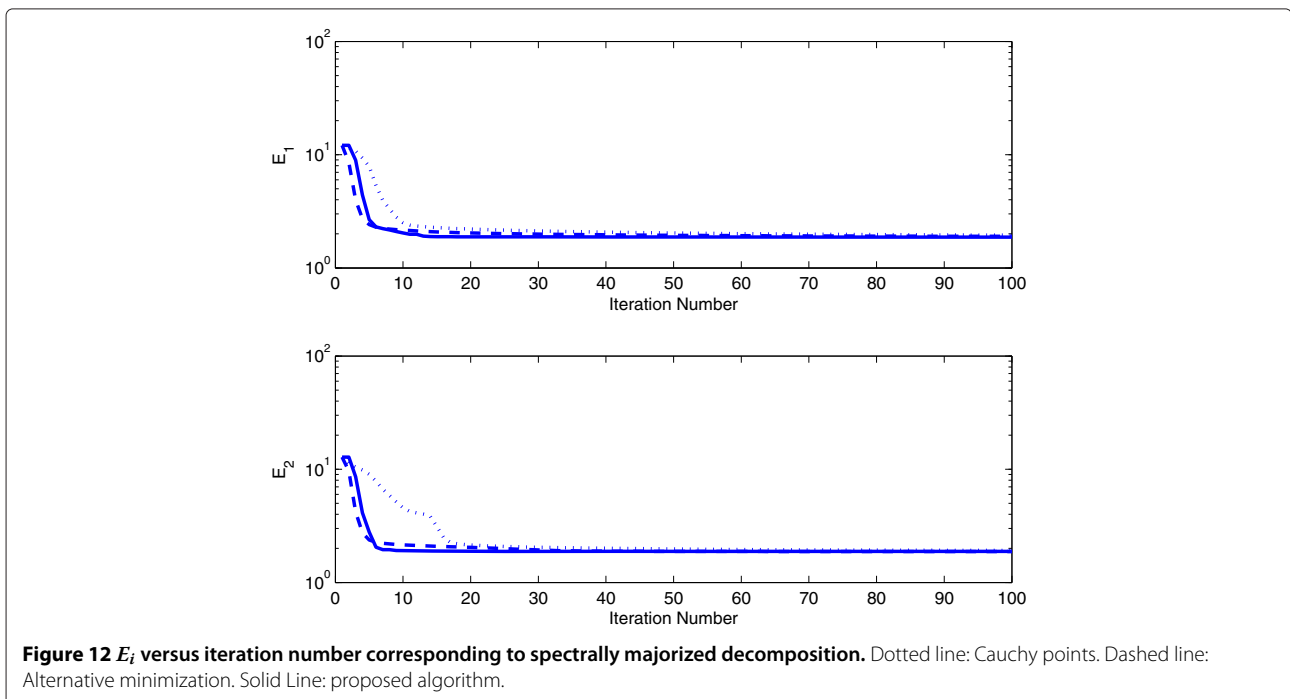

Figure 12 E_i versus iteration number corresponding to spectrally majorized decomposition. Dotted line: Cauchy points. Dashed line: Alternative minimization. Solid Line: proposed algorithm.
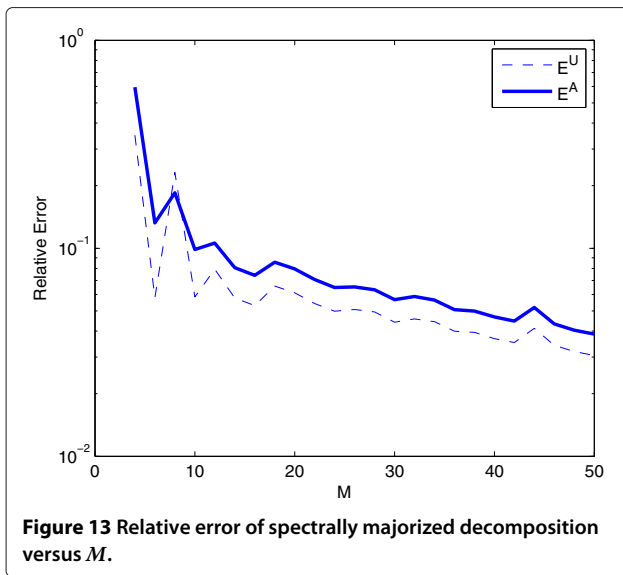
Figure 13 Relative error of spectrally majorized decomposition versus *M*.

a more relaxed decomposition with complex singular values. A solution for this nonlinear quadratic problem is proposed via Newton's method. Since we apply an approximate Hessian matrix to assist the Newton optimization, a fast convergence is achieved. The algorithm capability to control the order of polynomial elements of decomposed matrices and to select properties of decomposition, make the proposed method as a good choice for filterbank and MIMO precoding applications. Finally, performance of the proposed algorithm under different conditions is demonstrated via simulations. Simulation results reveal superior decomposition accuracy in contrast with coefficient domain algorithms due to relaxation of paraunitarity.
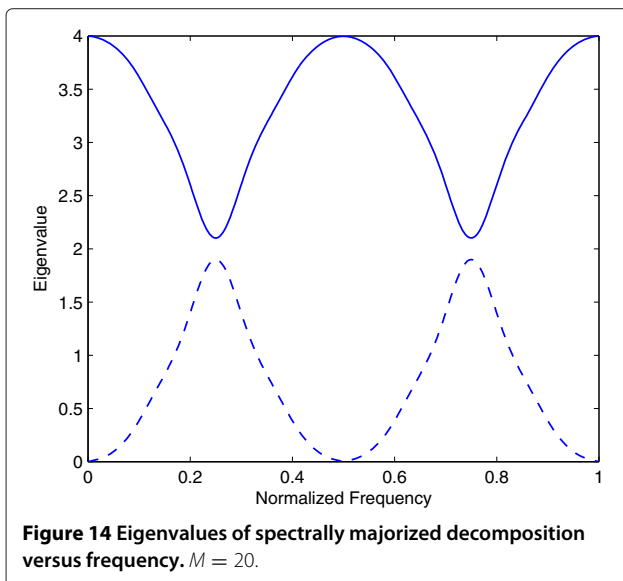

Figure 14 Eigenvalues of spectrally majorized decomposition versus frequency. $M = 20$.

Author details
[1] Department of Electrical Engineering, Amirkabir University of Technology, P.O. Box 15914, Tehran, Iran. [2] Electrical Engineering and Computer Science, University of Wisconsin-Milwaukee, P.O. Box 784, Milwaukee, WI 53201–0784, USA.

References
1. T Kailath, *Linear Systems*. (Prentice Hall, Englewood Cliffs, NJ, 1980)
2. J Tugnait, B. Huang, Multistep linear predictors-based blind identification and equalization of multiple-input multiple-output channels. IEEE Trans. Signal Process. **48**(1), 569–571 (2000)
3. R Fischer, Sorted spectral factorization of matrix polynomials in MIMO communications. IEEE Trans. Commun. **53**(6), 945–951 (2005)
4. H Zamiri-Jafarian, M Rajabzadeh, in *IEEE Vehicular Technology Conference, VTC Spring 2008*. A polynomial matrix SVD approach for time domain broadband beamforming in MIMO-OFDM systems, (2008), pp. 802–806
5. R Brandt, Polynomial matrix decompositions: evaluation of algorithms with an application to wideband MIMO communications (2010). Master of Science in Engineering Physics Thesis, Uppsala University, Uppsala, Sweden,
6. D Palomar, M Lagunas, A Pascual, A Neira, in *International Symposium on Signal Processing and Its Applications, ISSPA*. Practical implementation of jointly designed transmit-receive space-time IIR filters, (2001), pp. 521–524
7. R Lambert, A Bell, in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*. Blind separation of multiple speakers in a multipath environment, (1997), pp. 423–426
8. S Redif, J McWhirter, P Baxter, T Cooper, in *OCEANS 2006*. Robust broadband adaptive beamforming via polynomial eigenvalues, (2006), pp. 1–6
9. P Vaidyanathan, *Multirate Systems and, Filterbanks*, 4th edn. (Prentice Hall, Englewood Cliffs, 1993)
10. J Foster, J McWhirter, J Chamber, in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*. A novel algorithm for calculating the QR decomposition for a polynomial matrix, (2009), pp. 3177–3180
11. J Foster, J Mcwhirter, M Davies, J Chambers, An algorithm for calculating the QR and singular value decompositions of polynomial matrices. IEEE Trans. Signal Process. **58**(3), 1263–1274 (2010)
12. D Cescato, H Bolcskei, QR decomposition of Laurent polynomial matrices sampled on the unit circle. IEEE Trans. Inf. Theory. **56**(9), 4754–4761 (2010)
13. J Mcwhirter, P Baxter, T Cooper, S Redif, An EVD algorithm for para-Hermitian polynomial matrices. IEEE Trans. Signal Process. **55**(5), 2158–2169 (2007)
14. A Tkacenko, in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*. Approximate eigenvalue decomposition of para-Hermitian systems through successive FIR paraunitary transformations ((Dallas, Texas, USA, 2010), pp. 4074–4077
15. R Lambert, Multichannel blind deconvolution: FIR matrix algebra and separation of multipath mixtures (1996). PhD Thesis, University of Southern California, Los Angeles
16. P Vaidyanathan, Theory of optimal orthonormal subband coders. IEEE Trans. Signal Process. **46**(4), 1528–1543 (1998)
17. A Tkacenko, P Vaidyanathan, On the Spectral Factor Ambiguity of FIR Energy Compaction Filter Banks. IEEE Trans. Signal Process. **54**(1), 146–160 (2006)
18. R Brandt, M Bengtsson, in *International Symposium on Personal, Indoor, and Mobile Radio Communication*. Wideband MIMO channel diagonalization in the time domain, (2011), pp. 1958–1962
19. L Dieci, T Eirola, On smooth decomposition of matrices. SIAM J. Matrix Anal. Appl. **20**(3), 800–819 (1999)

20. S Redif, J McWhirter, S Weiss, Design of FIR paraunitary filter banks for subband coding using a polynomial eigenvalue decomposition. IEEE Trans. Signal Process. **59**(11), 5253–5264 (2011)

21. A Oppenheim, R Schafer, J Buck, *Discrete-Time Signal Processing*, 2nd edn. (Prentice Hall, Englewood Cliffs, 1999)

22. J Nocedal, S Wright, *Numerical Optimization*. (Springer, New York, 1999)