A RECURSIVE EIGENSPACE COMPUTATION FOR THE CANONICAL POLYADIC DECOMPOSITION*

ERIC EVERT†, MICHIEL VANDECAPPELLE†, AND LIEVEN DE LATHAUWER†

Abstract. The canonical polyadic decomposition (CPD) is a compact decomposition which expresses a tensor as a sum of its rank-1 components. A common step in the computation of a CPD is computing a generalized eigenvalue decomposition (GEVD) of the tensor. A GEVD provides an algebraic approximation of the CPD which can then be used as an initialization in optimization routines.

While in the noiseless setting GEVD exactly recovers the CPD, it has recently been shown that pencil based computations such as GEVD are unstable. In this article we present an algebraic method for approximation of a CPD which greatly improves on the stability of GEVD. Our method is still fundamentally pencil based; however, rather than using a single pencil and computing all of its generalized eigenvectors, we use many different pencils and in each pencil compute generalized eigenspaces corresponding to sufficiently well separated generalized eigenvalues. The resulting "generalized eigenspace decomposition" is significantly more robust to noise than the classical GEVD.

Stability of the generalized eigenspace decomposition is examined both empirically and theoretically. In particular, we provide probabilistic and deterministic bounds for stability of the algorithm.

Key words. tensors, canonical polyadic decomposition, generalized eigenvalue decomposition, multilinear algebra, multidimensional arrays

AMS subject classifications. Primary: 15A69, 15A22. Secondary: 15A42.

1. Introduction.

Tensors, or multiindexed arrays, play an important role in fields such as machine learning and signal processing. These higher-order generalizations of matrices allow for preservation of higher-order structure present in data and low rank decompositions of tensors allow for compression of data and recovery of underlying information [21, 3, 4]. One of the most popular decompositions for tensors is the canonical polyadic decomposition (CPD) which expresses a tensor as a sum of rank one tensors. An important feature of the CPD is that, with mild assumptions [8, 15, 20], the CPD of a low rank tensor is unique. Furthermore, for a low rank tensor with a unique CPD, the CPD can often be found algebraically. Such an algebraic solution can typically be obtained with limited computation time, hence is often used as an initialization for optimization based methods when the tensor is noisy.

One of the most popular algorithms for algebraic computation of a CPD is the generalized eigenvalue decomposition (GEVD) [9, 10, 19, 16]. The key idea behind the classical GEVD is that a factor matrix of a tensor may be obtained by computing the generalized eigenvectors of any subpencil of the tensor. However, it has recently been shown that pencil based algorithms such as GEVD are unstable [1]. That is, the condition number for computing a generalized eigenvalue decomposition of a subpencil

*Submitted to the editors May 27, 2021.

Funding: Research supported by: (1) Flemish Government: This work was supported by the Fonds de la Recherche Scientifique–FNRS and the Fonds Wetenschappelijk Onderzoek–Vlaanderen under EOS Project no 30468160 (SeLMA); (2) KU Leuven Internal Funds C16/15/059 and ID-N project no 3E190402. (3) This research received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme. (4) Work supported by Leuven Institute for Artificial Intelligence (Leuven.ai). (5) In addition, Michiel Vandecappelle was partially supported by an SB Grant (1S66617N) from the Research Foundation—Flanders (FWO).

†KU Leuven, Dept. of Electrical Engineering ESAT/STADIUS, Kasteelpark Arenberg 10, bus 2446, B-3001 Leuven, Belgium and KU Leuven – Kulak, Group Science, Engineering and Technology, E. Sabbelaan 53, B-8500 Kortrijk, Belgium

(Eric.Evert, Michiel.Vandecappelle, Lieven.DeLathauwer)@kuleuven.be.

can be arbitrarily larger than the condition number [27] for computing the CPD of the tensor. In this article we present an extension of the GEVD algorithm which significantly improves the stability of algebraic computation of the CPD. This increase in stability is obtained by using many subpencils of the tensor to compute the CPD and only performing stable computations with each subpencil.

The stability of computing generalized eigenvectors of a matrix pencil is heavily dependent on the separation between the generalized eigenvalues and the generalized eigenvectors of the pencil [24, 13, 18]. In the case that the generalized eigenvalues and eigenvectors are well separated, a GEVD may be stably computed. However, when either a pair of generalized eigenvalues or generalized eigenvectors are near parallel, computation of the generalized eigenvectors becomes unstable. As such, the GEVD algorithm performs well when there is a subpencil for the tensor in which all generalized eigenvalues are well separated; however, GEVD runs into challenges if one is unable to find a subpencil in which all generalized eigenvalues are well separated. It is not hard to show that this difficulty necessarily occurs when tensor rank becomes large.

We propose an alternative approach in which we exploit the fact that a tensor has many subpencils, and that a factor matrix of the tensor may be computed by combining information from many different subpencils. Using the language of the joint generalized eigenvectors (JGE vectors) [11], the primary goal of the GEVD algorithm is in fact to compute the JGE vectors of the tensor.

Let $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$. Suppose the tensor $\mathcal{T} \in \mathbb{K}^{R\times R\times K}$ is a rank R tensor with JGE vectors $\{\mathbf{x}_r\}_{r=1}^{R}$. Then for every subpencil of $\mathcal{T}$, the generalized eigenspaces are of the form $\operatorname{span}(\{\mathbf{x}_{r_1}, \ldots, \mathbf{x}_{r_k}\})$ for some collection of JGE vectors of $\mathcal{T}$. Intuitively it follows that one may compute the JGE vectors of $\mathcal{T}$ by computing generalized eigenspaces for many different subpencils of $\mathcal{T}$.

An important feature of this approach is that the stability of computing a particular generalized eigenspace does not depend on the separation of the generalized eigenvalues which correspond to that generalized eigenspace. Rather the stability depends on the separation of the set of generalized eigenvalues corresonding to the generalized eigenspace of interest from the other generalized eigenvalues of the pencil as well as the separation between the generalized eigenspace itself and the other generalized eigenvectors of the pencil [23]. Thus by only computing generalized eigenspaces which correspond to collections of generalized eigenvalues that are well separated from the other generalized eigenvalues of the pencil, one source of instability for the generalized eigenvalue decomposition is significantly reduced.

1.1. Notation and definitions. Let $\mathbb{K}$ denote either $\mathbb{R}$ or $\mathbb{C}$. We denote scalars, vectors, matrices, and tensors by lower case (a), bold lower case ($\mathbf{a}$), bold upper case ($\mathbf{A}$), and calligraphic script ($\mathcal{A}$), respectively. Additionally, fraktur font ($\mathfrak{A}$) is used to denote a subspace of $\mathbb{K}^I$. For a matrix $\mathbf{A} \in \mathbb{K}^{I\times I}$, we let $\mathbf{A}^{\mathrm{T}}$ denote the transpose of $\mathbf{A}$ while $\mathbf{A}^{\mathrm{H}}$ denotes the conjugate transpose of $\mathbf{A}$. If $\mathbf{A}$ is invertible, then we let $\mathbf{A}^{-\mathrm{T}}$ denote the inverse of $\mathbf{A}^{\mathrm{T}}$. A **tensor** is a multiindexed array with entries in $\mathbb{K}$. In this article we restrict our attention to tensors of order three. In other words, the tensors we consider are elements of $\mathbb{K}^{I_1\times I_2\times I_3}$.

Given nonzero vectors $\mathbf{a} \in \mathbb{K}^{I_1}, \mathbf{b} \in \mathbb{K}^{I_2}, \mathbf{c} \in \mathbb{K}^{I_3}$, let

$$\mathbf{a} \otimes \mathbf{b} \otimes \mathbf{c} \in \mathbb{K}^{I_1\times I_2\times I_3}$$

denote the $I_1 \times I_2 \times I_3$ tensor with i, j, k entry equal to $a_i b_j c_k$. A tensor of this form is called a **rank one tensor**. The minimal integer R such that $\mathcal{M} \in \mathbb{K}^{I_1\times I_2\times I_3}$ may

be written as

$$\mathcal{M} = \sum_{r=1}^{R} \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r$$

where the vectors $\mathbf{a}_r, \mathbf{b}_r, \mathbf{c}_r$ have entries in $\mathbb{K}$ is called the $\mathbb{K}$-rank of the tensor $\mathcal{M}$, and a decomposition of this form is called a canonical polyadic decomposition (CPD) of $\mathcal{M}$. Compactly we write $\mathcal{M} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$ where the matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ have $\mathbf{a}_r, \mathbf{b}_r$ and $\mathbf{c}_r$ as their rth columns, respectively. The matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are called the **factor matrices** for $\mathcal{M}$.

Given a tensor $\mathcal{M} \in \mathbb{K}^{I_1 \times I_2 \times I_3}$, let $\mathbf{M}_{[1;3,2]}$ denote an $I_1 \times I_2 I_3$ matrix with columns $\{\mathbf{m}(:, k, \ell)\}_{\forall k,\ell}$. Here, the columns of $\mathbf{M}_{[1;3,2]}$ are ordered so that the second index of $\mathbf{m}(:, k, \ell)$ increments faster than the third. The matrices $\mathbf{M}_{[2;3,1]}$ and $\mathbf{M}_{[3;2,1]}$ are defined similarly. A matrix of the form $\mathbf{M}_{[i;k,j]}$ is called a **mode-i unfolding** of $\mathcal{M}$.

Having defined the mode-i unfoldings of $\mathcal{M}$, we define

$$R_i(\mathcal{M}) \coloneqq \operatorname{rank}(\mathbf{M}_{[i;k,j]}) \qquad \text{for } i = 1, 2, 3. \tag{1.1}$$

The integer triple $(R_1(\mathcal{M}), R_2(\mathcal{M}), R_3(\mathcal{M}))$ is called the **multilinear rank** of $\mathcal{M}$.

For tensors $\mathcal{M}$ occurring in applications, $\mathcal{M}$ is often thought of as a noisy measurement of some signal tensor of interest. That is, one can decompose $\mathcal{M} = \mathcal{T}' + \mathcal{N}'$ where $\mathcal{T}'$ is the true signal tensor of interest and $\mathcal{N}'$ is measurement error. Most commonly the signal tensor $\mathcal{T}'$ has low multilinear rank. That is, one has $R_i(\mathcal{T}') \ll I_i$ for each i. In this case, under an appropriate unitary change of bases one has $t'_{i,j,k} = 0$ if $i > R_1(\mathcal{T}')$ or $j > R_2(\mathcal{T}')$ or $k > R_3(\mathcal{T}')$. In this choice of bases it is clear that $\mathcal{T}'$ may be compressed to a tensor of size $R_1(\mathcal{T}') \times R_2(\mathcal{T}') \times R_3(\mathcal{T}')$ without loss of information. Such a compression is called an **orthogonal compression** of $\mathcal{T}'$ to its multilinear rank, and a popular choice of orthogonal compression is the multilinear singular value decomposition. See [5] for details.

Compression of a noisy measurement $\mathcal{M} = \mathcal{T}' + \mathcal{N}'$ to the multilinear rank of $\mathcal{T}'$ is a common step in computation of a CPD which both approximates the (core of) the signal tensor $\mathcal{T}'$ and reduces computational complexity. However, such a compression has few implications for the methods discussed in this article. For this reason we often restrict to considering tensors $\mathcal{Y}$ with the property $R_i(\mathcal{Y}) = I_i$ for $i = 1, 2, 3$. Intuitively, the symbol $\mathcal{Y}$ is used to denote a tensor which can be thought of as a compression of $\mathcal{M} = \mathcal{T}' + \mathcal{N}'$ to the multilinear rank of $\mathcal{T}'$. In particular, we may write $\mathcal{Y} = \mathcal{T} + \mathcal{N}$ where $\mathcal{T}$ is an orthogonal compression $\mathcal{T}'$ to its multilinear rank and $\mathcal{N}$ is noise.

It is not difficult to show that the $\mathbb{K}$-rank of a tensor $\mathcal{Y}$ is bounded from below by each of the $R_i(\mathcal{Y})$. For most tensors the inequality is strict. However, for the signal portion $\mathcal{T}$ of a tensor $\mathcal{Y}$ occurring in applications, one often has (up to a permutation of indices) $\operatorname{rank}_{\mathbb{K}}(\mathcal{T}) = R_1(\mathcal{T}) = R_2(\mathcal{T}) \geq R_3(\mathcal{T})$. In light of this, the tensors $\mathcal{Y} = \mathcal{T} + \mathcal{N}$ we consider are taken to have size $R \times R \times K$ where $K \leq R$ and the signal portion $\mathcal{T}$ of $\mathcal{Y}$ is assumed to have $\mathbb{K}$-rank R. To summarize, throughout the article we let $\mathcal{M}$ denote an arbitrary tensor of size $I_1 \times I_2 \times I_3$, we let $\mathcal{Y}$ denote a full multilinear rank tensor of size $R \times R \times K$, and we let $\mathcal{T}$ denote a $R \times R \times K$ tensor with $\mathbb{K}$-rank equal to R and full multilinear rank.

Define the **mode-1 product** between a tensor $\mathcal{M} \in \mathbb{K}^{I_1 \times I_2 \times I_3}$ and a matrix $\mathbf{U} \in \mathbb{K}^{J_1 \times I_1}$ to be the tensor $\mathcal{M} \cdot_1 \mathbf{U} \in \mathbb{K}^{J_1 \times I_2 \times I_3}$ with mode-1 fibers given by

$$(\mathcal{M} \cdot_1 \mathbf{U})(:, k, \ell) = \mathbf{U} \cdot \mathbf{t}(:, k, \ell) \qquad \text{for all } k, \ell.$$

The mode-2 and mode-3 products are defined analogously. Say a tensor $\mathcal{Y} \in \mathbb{K}^{R\times R\times K}$ is **slice mix invertible** if there is a vector $\mathbf{v} \in \mathbb{R}^K$ such that the matrix $\mathcal{Y} \cdot_3 \mathbf{v}^{\mathrm{T}}$ is invertible. That is, $\mathcal{Y}$ is slice mix invertible if there exists a linear combination of the matrices $\mathbf{Y}(:,:,k)$ for $k = 1,\dots,K$ which is invertible. We call the matrices $\mathbf{Y}(:,:,k)$ the **frontal slices** of $\mathcal{Y}$ and use $\mathbf{Y}_k$ to denote $\mathbf{Y}(:,:,k)$ for $k = 1,\dots,K$.

Let $\mathcal{Y} \in \mathbb{K}^{R\times R\times K}$. An **implicit subpencil** of $\mathcal{Y}$ is a tensor of the form $\mathcal{Y} \cdot_3 \mathbf{V} \in \mathbb{K}^{R\times R\times 2}$ where $\mathbf{V} \in \mathbb{K}^{2\times K}$ has rank 2. More generally, call a tensor of the form $\mathcal{Y} \cdot_3 \mathbf{V} \in \mathbb{K}^{R\times R\times L}$ where $\mathbf{V} \in \mathbb{K}^{L\times K}$ has full row rank an **implicit subtensor** of $\mathcal{Y}$. For emphasis we note that implicit subpencils and implicit subtensors differ from the usual notions in that they can be revealed by modal multiplication in the third mode of the tensor. Throughout the article, implicit subtensors and subpencils of a given tensor are denoted using the calligraphic letter $\mathcal{S}$. In the reminder of the article we drop the distinction "implicit", as all subpencils and subtensors we consider are implicit.

1.2. Joint generalized eigenvalues and eigenvectors. Given a tensor $\mathcal{T} \in \mathbb{K}^{R\times R\times K}$, say a nonzero vector $\mathbf{z} \in \mathbb{K}^R$ is a **joint generalized eigenvector** (JGE vector) of $\mathcal{T}$ if there exists a $\boldsymbol{\lambda} \in \mathbb{K}^K$ and a nonzero vector $\mathbf{q} \in \mathbb{K}^R$ such that

$$\mathbf{T}_\ell \mathbf{z} = \lambda_\ell \mathbf{q} \tag{1.2}$$

for all $\ell = 1,\dots,K$. Note that assuming $\mathbf{q} \neq \mathbf{0}$ guarantees that $\lambda_i = 0$ if and only if $\mathbf{Y}_i\mathbf{x} = \mathbf{0}$. If $(\boldsymbol{\lambda}, \mathbf{z})$ is a solution to (1.2), then the one-dimensional subspace $\mathfrak{L} \coloneqq \operatorname{span}(\boldsymbol{\lambda}) \subset \mathbb{K}^K$ is a **joint generalized eigenvalue** of $\mathcal{Y}$ and $(\mathfrak{L}, \mathbf{z})$ is a **joint generalized eigenpair** of $\mathcal{Y}$.[1] [11, Theorem 1.1] shows that if $\mathcal{T} \in \mathbb{K}^{R\times R\times K}$ is slice mix invertible and has $\mathbb{K}$-rank R, then computation of a CPD for $\mathcal{T}$ is equivalent to computing the JGE pairs of $\mathcal{T}$. For more details about joint generalized eigenvalues and eigenvectors, see [11].

Given a subspace $\mathfrak{Z} \subset \mathbb{K}^R$ of dimension ℓ and a tensor $\mathcal{T} \in \mathbb{K}^{R\times R\times K}$, say $\mathfrak{Z}$ is a **joint generalized eigenspace** of $\mathcal{T}$ if there is a subspace $\mathfrak{Q} \subset \mathbb{K}^R$ of dimension less than or equal to ℓ such that

$$\mathbf{T}_k\mathfrak{Z} \subseteq \mathfrak{Q} \qquad \text{for all } k = 1,\dots,K.$$

In the case that $K = 2$ we simply call $\mathfrak{Z}$ a **generalized eigenspace**.[2] Given a JGE value $\mathfrak{L} \subset \mathbb{K}^K$ of a tensor $\mathcal{T} \in \mathbb{K}^{R\times R\times K}$, let $\mathfrak{Z}_\mathfrak{L} \subset \mathbb{K}^R$ be the subspace of joint generalized eigenvectors corresponding to $\mathfrak{L}$. One may easily show that $\mathfrak{Z}_\mathfrak{L}$ is a joint generalized eigenspace of $\mathcal{T}$.

1.2.1. Generalized eigenspaces of subtensors. The main goal of this article is to improve the stability of algebraic (pencil based) methods for computing a CPD. Key observations which allow us to accomplish this are that generalized subspaces may often be more stably computed than individual generalized eigenvectors and that the joint generalized eigenvectors of a tensor may be recovered by computing generalized eigenspaces for sufficiently many subtensors. The following proposition helps formalize this idea.

PROPOSITION 1.1. *Let $\mathcal{T} \in \mathbb{K}^{R\times R\times K}$ be a tensor of $\mathbb{K}$-rank R and let $\mathbf{V} \in \mathbb{K}^{L\times K}$ be a matrix with full row rank. Set $\mathcal{S} = \mathcal{T} \cdot_3 \mathbf{V}$, assume $\mathcal{S}$ is slice mix invertible, and*

[1] If K=2, then joint generalized eigenvectors and joint generalized eigenvalues are exactly equal to classical generalized eigenvalues and generalized eigenvectors (in the sense of [24]).

[2] Generalized eigenspaces are sometimes called deflating subspaces by other authors.

let $\mathfrak{L}_{\mathcal{S}}$ be a JGE value of $\mathcal{S}$. There exist JGE values $\mathfrak{L}_1, \ldots, \mathfrak{L}_J$ of $\mathcal{T}$ such that

$$\mathfrak{Z}_{\mathfrak{L}_{\mathcal{S}}} = \operatorname{span}(\{\mathfrak{Z}_{\mathfrak{L}_j}\}_{j=1}^{J}).$$

Proof. First note that assuming $\mathcal{S}$ is slice mix invertible implies $\mathcal{T}$ is slice mix invertible. Now, following the proof of [11, Proposition 4.1], one finds that each joint generalized eigenvalue of $\mathcal{S}$ has the form $\mathbf{V}\mathfrak{L}$ for some JGE value $\mathfrak{L}$ of $\mathcal{T}$. Given some JGE value $\mathfrak{L}_{\mathcal{S}}$ of $\mathcal{S}$, let $\mathfrak{L}_1, \ldots, \mathfrak{L}_J$ be those joint generalized eigenvalues of $\mathcal{T}$ such that $\mathbf{V}\mathfrak{L}_j = \mathfrak{L}_{\mathcal{S}}$. It is straightforward to check that

$$\operatorname{span}(\{\mathfrak{Z}_{\mathfrak{L}_j}\}_{j=1}^{J}) \subset \mathfrak{Z}_{\mathfrak{L}_{\mathcal{S}}}. \tag{1.3}$$

To prove the reverse containment, use the assumption that $\mathcal{S}$ is slice mix invertible and the fact that $\mathcal{T}$ has rank R to conclude $\mathcal{S}$ has rank R. Letting $\{\mathfrak{L}_{\mathcal{S},1}, \ldots, \mathfrak{L}_{\mathcal{S},\ell}\}$ be the set of JGE values of $\mathcal{S}$, it follows from [11, Theorem 1.1] and [11, Lemma 2.2] that the collection of spaces $\{\mathfrak{Z}_{\mathfrak{L}_{\mathcal{S},i}}\}_{i=1}^{\ell}$ spans $\mathbb{K}^R$ and also that these spaces pairwise have trivial intersection. Again using [11, Theorem 1.1] shows that the JGE vectors of $\mathcal{T}$ span $\mathbb{K}^R$. Thus, since each JGE vector of $\mathcal{T}$ lies in some $\mathfrak{Z}_{\mathfrak{L}_{\mathcal{S},i}}$, the containment in equation (1.3) must in fact be an equality. ∎

1.3. Guide to the reader. Section 2 explains the GESD algorithm for computation of a canonical polyadic decomposition. In Sections 3 and 4 we give bounds for stability of the GESD algorithm. In particular, Section 3 discusses a probabilistic bound for stability based on the Cramér-Rao induced bound for CPD, while Section 4 gives a deterministic bound for stability using perturbation theory for JGE values. Sections 5 and 6 present numerical experiments for the GESD algorithm. In Section 5 we compare the performance (in terms of timing and accuracy) of GESD to the classical GEVD algorithm. In Section 6 we compare the accuracy of the GESD algorithm to the bounds computed in Sections 3 and 4.

2. The GESD algorithm. We now give a formal description of the generalized eigenspace decomposition (GESD) for computation of a canonical polyadic decomposition. The main step of the algorithm is to use stably computable generalized eigenspaces to recursively split a given tensor into subtensors of reduced rank until the subtensors all have rank one (up to noise). The CPD of each (noisy) rank one tensor may be computed by a best rank one approximation, and the CPD for the full tensor is obtained by combining the decompositions of these rank one tensors.

Let $\mathcal{T}$ be an $R \times R \times K$ tensor of rank R with CPD $\mathcal{T} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$. Let $\mathbf{V} \in \mathbb{K}^{K\times 2}$ be a column-wise orthonormal matrix[3] and set $\mathcal{S} = \mathcal{T} \cdot_3 \mathbf{V}^{\mathrm{T}}$. The GESD algorithm begins by computing a collection of generalized eigenspaces $\mathfrak{Z}_1, \ldots, \mathfrak{Z}_L$ for the generalized eigenvalue problem

$$\mathbf{S}_i\mathbf{z} = \boldsymbol{\lambda}_i\mathbf{q} \text{ for } i = 1, 2 \tag{2.1}$$

such that $\operatorname{span}(\{\mathfrak{Z}_\ell\}_{\ell=1}^{L}) = \mathbb{K}^R$. In each step the GESD algorithm computes as many well separated eigenspaces as possible; however, to simplify exposition we will take $L = 2$.

Say the generalized eigenvalues of $\mathcal{S}$ are split into two distinct groups of size j_1 and j_2 where $j_1 + j_2 = R$. The corresponding eigenspaces may then for instance be

[3] Column-wise orthonormal matrices are used for numerical reasons.

computed with a **QZ** decomposition[4] of the matrix pencil $(\mathbf{S}_1, \mathbf{S}_2)$ [2]. That is we find unitary matrices $\mathbf{Q}, \mathbf{Z} \in \mathbb{K}^R$ such that

$$\mathbf{S}_1 = \mathbf{Q}\mathbf{X}_1\mathbf{Z}^{\mathrm{H}} \qquad \text{and} \qquad \mathbf{S}_2 = \mathbf{Q}\mathbf{X}_2\mathbf{Z}^{\mathrm{H}} \tag{2.2}$$

where $\mathbf{X}_1$ and $\mathbf{X}_2$ are (quasi[5]) upper triangular matrices with the classical generalized eigenvalues of the pencil on the diagonal.

If the decomposition is ordered so that the j_1 generalized eigenvalues in the first cluster appear first on the diagonal of $\mathbf{X}_1$ and $\mathbf{X}_2$, then the first j_1 columns of $\mathbf{Z}$ give us a matrix $\mathbf{Z}_1$ whose columns form a basis for the corresponding generalized eigenspace of $\mathcal{S}$. Similarly, we obtain a matrix $\mathbf{Z}_2$ whose columns form a basis for the eigenspace corresponding to the next j_2 generalized eigenvalues by reordering the terms in the **QZ** decomposition so that these generalized eigenvalues appear first on the diagonal of $\mathbf{X}_1$ and $\mathbf{X}_2$.

Using Proposition 1.1 together with [11, Theorem 1.1] shows that the matrices $\mathbf{Z}_1$ and $\mathbf{Z}_2$ that we have obtained are of the form $\mathbf{Z}_i = \mathbf{B}^{-\mathrm{T}}\mathbf{E}_i$ where the $\mathbf{E}_i$ have the property that the jth row of $\mathbf{E}_1$ is nonzero if and only if the jth row of $\mathbf{E}_2$ is a zero row, and vice versa. Define the tensors $\mathcal{T}^1$ and $\mathcal{T}^2$ by

$$\mathcal{T}^i = \mathcal{T} \cdot_2 \mathbf{Z}_i^{\mathrm{T}} = [\![\mathbf{A}, \mathbf{Z}_i^{\mathrm{T}}\mathbf{B}, \mathbf{C}]\!] = [\![\mathbf{A}, \mathbf{E}_i^{\mathrm{T}}, \mathbf{C}]\!].$$

and for $i = 1, 2$ set $R_i = \operatorname{rank}(\mathcal{T}^i)$. Since the $\mathbf{E}_i^{\mathrm{T}}$ have zero columns corresponding to the zero rows of $\mathbf{E}_i$, we have that $R_1 + R_2 = \operatorname{rank}(\mathcal{T})$. Moreover, suppose $\mathcal{T}^i$ has CPD $[\![\mathbf{A}_i, \check{\mathbf{E}}_i, \mathbf{C}_i]\!]$ for $i = 1, 2$. Here the matrix $\check{\mathbf{E}}_i \in \mathbb{K}^{R \times R_i}$ is the matrix obtained by removing the zero columns of $\mathbf{E}_i^{\mathrm{T}}$. Then up to permutation and scaling, one has $\mathbf{A} = (\mathbf{A}_1 \;\; \mathbf{A}_2)$. Similarly, the factor matrix $\mathbf{C}$ of $\mathcal{T}$ has the form $\mathbf{C} = (\mathbf{C}_1 \; \mathbf{C}_2)$ up to scaling and the same permutation as $\mathbf{A}$.

The above methodology may be recursively applied to each $\mathcal{T}^i$ to further reduce their ranks until one obtains a collection of tensors all having rank 1 (up to noise). The columns of the factor matrices $\mathbf{A}$ and $\mathbf{C}$ may then be approximated by computing rank 1 approximations for each of these (approximately) rank one tensors, see Section 2.1 below for details.

Once we have computed the factor matrices $\mathbf{A}$ and $\mathbf{C}$, we may compute $\mathbf{B}$ by solving

$$\mathbf{T}_{[2;3,1]} = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^{\mathrm{T}}.$$

2.1. Rank one terms. Consider the case where $\mathbf{Z}_1 \in \mathbb{K}^{R \times 1}$ where $\mathbf{Z}_1$ spans an eigenspace of dimension one. In this case $\mathbf{Z}_1$ must have the form

$$\mathbf{Z}_1 = \beta \mathbf{B}^{-\mathrm{T}}\mathbf{e}_\ell$$

where $\mathbf{e}_\ell$ is a standard basis vector and $\beta \in \mathbb{K}$ is some constant. Thus we have

$$\mathcal{T} \cdot_2 \mathbf{Z}_1^{\mathrm{T}} = [\![\mathbf{A}, \beta\mathbf{e}_\ell, \mathbf{C}]\!] = [\![\mathbf{a}_\ell, \beta, \mathbf{c}_\ell]\!].$$

In this case we may compute $\mathbf{a}_\ell$ and $\mathbf{c}_\ell$ by computing a rank one approximation of the matrix $\mathcal{T} \cdot_2 \mathbf{Z}_1^{\mathrm{T}}$.

[4] The **QZ** decomposition is the generalization of the Schur decomposition to matrix pencils, e.g., see [24, Chapter VI, Theorem 1.9].

[5] When working over $\mathbb{R}$, if $\mathbf{S}$ has complex generalized eigenvectors, then since $\mathbf{Q}$ and $\mathbf{Z}$ have real entries, there will be 2×2 block matrices on the diagonal of $\mathbf{X}_1$ and $\mathbf{X}_2$. E.g., see [14, Theorem 1.2]

2.2. Choice of slices. We observe that the average performance of the GESD algorithm is improved by first considering a pencil formed by taking the first two frontal slices of the core of the MLSVD of the current subtensor on each splitting step. Intuitively, this is related to the fact that this matrix pencil explains more of the overall sum of squared entries than any other matrix pencil which can be formed from the frontal slices of the tensor.

We emphasize that this strategy is not always successful. Indeed, there are many settings where the generalized eigenvalues obtained by considering the first two slices of the core of an MLSVD are poorly separated. When this occurs it is necessary to consider other matrix pencils. See Section 5.4 for an example. In the case that a successful split is not obtained by considering slices of the MLSVD, we move on to considering random linear combinations of frontal slices.

Note that an MLSVD is always computed for each subtensor to address other potential issues (e.g., repeated factor matrix columns). Furthermore computing MLSVDs of the subtensors is cheap relative to other steps of the algorithm. Thus first considering slices of the MLSVDs has little impact on overall computational cost and on average significantly improves accuracy.

Pseudo code for the GESD algorithm is found in Algorithm 1 and Algorithm 2.

Algorithm 1: GESD recursive step

function `cpd_gesd` $(\mathcal{M}, R, \texttt{Random})$
Input : Tensor $\mathcal{M}$, rank R, boolean `Random`
Output: $[\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$
$K = \min(\text{size}(\mathcal{T}, 3), R)$
$[\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathcal{Y}] = \texttt{mlsvd}(\mathcal{M}, [R, R, K])$; % Compute MLSVD of $\mathcal{T}$/ Compress
if $R = 1$ **then**
  $\mathbf{A}_{\text{core}} = y_{111}$; $\mathbf{B}_{\text{core}} = 1$; $\mathbf{C}_{\text{core}} = 1$; % Trivial case
else
  $[\mathbf{Z}_1, \ldots, \mathbf{Z}_N] = \texttt{qz_split}(\mathcal{Y}, \texttt{Random})$; % Split JGE spaces into N separated subgroups (Algorithm 2)
  for $i = 1 : N$ **do**
    $\mathcal{Y}^i = \mathcal{Y} \cdot_2 \mathbf{Z}_i^{\mathrm{T}}$; % Reduce tensor to Nth subgroup
    $R_i = \texttt{size}(\mathbf{Z}_i, 2)$ % Specify rank for Nth subgroup
    $[\![\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i]\!] = \texttt{cpd_gesd}(\mathcal{Y}^i, R_i)$; % Recursion on smaller tensor and smaller rank
  end
  $\mathbf{A}_{\text{core}} = [\mathbf{A}_1, \ldots, \mathbf{A}_N]$; $\mathbf{C}_{\text{core}} = [\mathbf{C}_1, \ldots, \mathbf{C}_N]$; % Build $\mathbf{A}$ and $\mathbf{C}$ from subblocks
  $\mathbf{B}_{\text{core}} = \mathbf{Y}_{[2;3,1]}/(\mathbf{C} \odot \mathbf{A})^{\mathrm{T}}$; % Find $\mathbf{B}$ by solving linear system
end
$\mathbf{A} = \mathbf{V}_1\mathbf{A}_{\text{core}}$; $\mathbf{B} = \mathbf{V}_2\mathbf{B}_{\text{core}}$; $\mathbf{C} = \mathbf{V}_3\mathbf{C}_{\text{core}}$; % Undo the MLSVD / Expand

2.3. Handling of complex eigenvectors when working over $\mathbb{R}$. Our explanation of the GESD algorithm above focuses on the noiseless setting. An important aspect of this setting is that the joint generalized eigenvectors of a slice mix invertible real rank R tensor $\mathcal{T} \in \mathbb{R}^{R \times R \times K}$ are necessarily real valued [11], and, as a consequence, the subpencils of such a tensor have real valued generalized eigenvectors. In

Algorithm 2: Splitting eigenspaces (over $\mathbb{R}$)[6]

function `qz_split` $(\mathcal{Y}, \texttt{Random})$
Input : MLSVD core tensor $\mathcal{Y}$, boolean `Random`
Output: N matrices $\mathbf{Z}_n$ spanning JGE spaces
$R = \texttt{size}(\mathcal{Y}, 1)$
$N = 0$
while $N < 2$ **do**
  if *Random* **then**
    $\mathcal{S} = \mathcal{Y} \cdot_3 \texttt{qr}(\texttt{randn}(R, 2))^{\mathrm{T}}$; % Construct random pencil
  else
    Set indices k_1 and k_2 of next set of MLSVD slices
    $\mathcal{S} = \mathcal{Y}_{:,:,[k_1,k_2]}$; % Take MLSVD slices as pencil
  end
  $[\mathbf{X}_1, \mathbf{X}_2, \mathbf{Q}, \mathbf{Z}] = \texttt{qz}(\mathbf{S}_{::1}, \mathbf{S}_{::2})$;
    % Compute QZ decomposition of pencil
  Order eigenvalues $x_{2,ii}/x_{1,ii}$
  Compute chordal gaps between adjacent pairs $(x_{1,ii}, x_{2,ii})$
  Set N to be the number of chordal gaps above threshold
end
Split eigenvalues into N clusters according to gaps above threshold
for $n = 1 : N$ **do**
  Define binary vector $\mathbf{i}_n$ indicating the nth cluster.
  $[\mathbf{X}_{1,n}, \mathbf{X}_{2,n}, \mathbf{Q}_n, \mathbf{Z}_n] = \texttt{ordqz}(\mathbf{X}_1, \mathbf{X}_2, \mathbf{Q}, \mathbf{Z}, \mathbf{i}_n)$;
    % Modify QZ decomposition such that nth cluster is at top
  $\mathbf{Z}_n = (\mathbf{Z}_n)_{:,1:\texttt{sum}(\mathbf{i}_n)}$ % Extract JGE space of nth cluster
end

the noisy setting, however, it is possible that a subpencil of a perturbation $\mathcal{Y} = \mathcal{T} + \mathcal{N}$ has complex valued generalized eigenvectors.

For example, suppose $\mathcal{T} \in \mathbb{R}^{R \times R \times K}$ is a slice mix invertible tensor of real rank R, and suppose $\mathfrak{Z}$ is a JGE space of dimension 2. Let $\mathbf{Z} \in \mathbb{K}^{R \times 2}$ be a matrix whose columns form a basis for $\mathfrak{Z}$ and let $\mathcal{S}$ be an MLSVD of the tensor $\mathcal{Y} \cdot_2 \mathbf{Z}^{\mathrm{T}}$. Then $\mathcal{S}$ is a $2 \times 2 \times 2$ tensor. In the noiseless setting (i.e., if $\mathcal{Y} = \mathcal{T}$), if $\mathcal{S}$ is slice mix invertible, then $\mathcal{S}$ necessarily has a real rank 2 and both generalized eigenvectors of $\mathcal{S}$ are real valued. However, with sufficient noise, it is possible that $\mathcal{S}$ has real rank 3. This corresponds to the case where the generalized eigenvectors of $\mathcal{S}$ are complex valued.

If $\mathcal{S}$ has rank 3, then $\mathcal{S}$ does not have a best real rank 2 approximation [6], so $\mathcal{S}$ cannot be further decomposed and the true real rank R decomposition of the original tensor cannot be computed by the GESD algorithm at this point. However, the factor matrix $\mathbf{A}$ (resp. $\mathbf{C}$) in the CPD for the original tensor will have two columns which (approximately) lie in the mode 1 (resp. mode 3) subspace of the tensor $\mathcal{Y} \cdot_2 \mathbf{Z}^{\mathrm{T}}$.

At this point one may accept that these particular columns of $\mathbf{A}$ and $\mathbf{C}$ cannot be computed and replace them with bases for the two dimensional subspaces to which they belong. Alternatively, one could use the results from the GESD algorithm (with

[6]When working over $\mathbb{C}$ there is no natural way to sort the generalized eigenvalues. As a consequence one must compute the chordal distance between all pairs of generalized eigenvalues in order to cluster the generalized eigenvalues into well separated collections.

the bases for these two dimensional subspaces used as those two columns for $\mathbf{A}$ and $\mathbf{C}$) as an initialization in an optimization routine for computing the full CPD. While it is unlikely that these basis vectors are the true columns of $\mathbf{A}$ or $\mathbf{C}$, the optimization routine will at least be initialized with these columns in the correct two dimensional subspace which is a significant improvement over a random initialization.

3. A probabilistic bound for stability of GESD. We now examine the stability of the GESD algorithm. Our main goal in this section is to provide probabilistic bounds for the existence of a subpencil of the tensor of interest which has two distinct clusters of generalized eigenvalues which cannot have coalesced, a notion we now explain.

3.1. Coalescence of generalized eigenvalues. Suppose $\mathcal{T} \in \mathbb{K}^{R\times R\times 2}$ is a matrix pencil with distinct generalized eigenvalues. Then for sufficiently small perturbations $\mathcal{N}$, the generalized eigenvalues and generalized eigenvectors of $\mathcal{T}+\mathcal{N}$ are continuous in $\mathcal{N}$. However, if the perturbation $\mathcal{N}$ is large enough so that the matrix pencil $\mathcal{T}+\mathcal{N}$ has a repeated generalized eigenvalue, i.e., so that at least two generalized eigenvalues of $\mathcal{T}+\mathcal{N}$ have coalesced, then the path of corresponding generalized eigenvectors cannot be unambiguously continued, as after the generalized eigenvalues separate there is ambiguity in which generalized eigenvalue corresponds to which generalized eigenvector.

Although it is no longer possible to unambiguously continue the paths of generalized eigenvectors themselves after generalized eigenvalues have coalesced, the path of the corresponding generalized eigenspace is still continuous and unambiguous in $\mathcal{N}$ so long as the underlying generalized eigenvalues do not coalesce with other generalized eigenvalues. Thus guaranteeing the existence of two distinct clusters of generalized eigenvalues which cannot have coalesced guarantees that $\mathcal{T}+\mathcal{N}$ has at least two distinct generalized eigenspaces which are still continuous in the perturbation $\mathcal{N}$. These two generalized eigenspaces are still closely related to the original generalized eigenspaces of $\mathcal{T}$.

We note that the bounds presented in this section may be adapted not only to guarantee the existence of clusters of generalized eigenvalues which have not coalesced, but also to guarantee a minimum separation between the clusters of generalized eigenvalues. As previously mentioned, the separation between clusters of generalized eigenvalues heavily impacts the stability of computing corresponding generalized eigenspaces, so in this way our bounds have significant implications for the stability of a given step of the GESD algorithm.

3.2. A Cramér-Rao based bound for distinct eigenvalue clusters. Our basic strategy to obtain a probabilistic bound is to use the Cramér-Rao induced bound (CRIB) for the mean-square angular deviation of the vectors of a factor matrix as described in [25] to build confidence intervals for the generalized eigenvalues of a matrix pencil. These confidence intervals can be used to say that clusters of generalized eigenvalues have not coalesced with high probability, i.e., that the separation of the corresponding eigenspaces is well justified in GESD.

Before proceeding we remark that the CRIB is only guaranteed to be valid in a neighborhood around the tensor. Thus, while for small noise there is no issue, for large enough noise the CRIB may become unreliable. Despite this difficulty, the CRIB is still highly predictive of behaviour exhibited by the GESD algorithm in simulated data.

Now, let $\mathcal{T} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \in \mathbb{K}^{R\times R\times K}$ be a rank R tensor and let $\mathcal{N}(\sigma)$ be a random

Gaussian noise tensor whose independent identically distributed (i.i.d.) entries have variance σ^2. Assume that $\mathcal{T}+\mathcal{N}(\sigma)$ is slice mix invertible and has a best rank R approximation denoted by $\hat{\mathcal{T}}$ which has CPD $\hat{\mathcal{T}}=[\![\hat{\mathbf{A}},\hat{\mathbf{B}},\hat{\mathbf{C}}]\!]$. Let $\mathbf{c}_k$ and $\hat{\mathbf{c}}_k$ denote the kth columns of the factor matrices $\mathbf{C}$ and $\hat{\mathbf{C}}$, respectively. The CRIB for a column $\mathbf{c}_k$ of the factor matrix $\mathbf{C}$, denoted CRIB($\mathbf{c}_k$), computes a lower bound for the variance of the angle between $\mathbf{c}_k$ and $\hat{\mathbf{c}}_k$. For details on theory and computation of the CRIB we refer the reader to [25, 26]. For general discussion of Cramér-Rao bounds for tensors, also see [17].

Note that if $\mathcal{T}$ is a tensor with two frontal slices, then the generalized eigenvalues of the pencil $(\mathbf{T}_1,\mathbf{T}_2)$ are given by the spans of the columns of the factor matrix $\mathbf{C}$. Therefore, we may directly apply the CRIB to a matrix pencil to get a lower bound on the variance of the angles of the pencil's generalized eigenvalues.

Let $\mathfrak{L}_1,\dots,\mathfrak{L}_R$ and $\hat{\mathfrak{L}}_1,\dots,\hat{\mathfrak{L}}_R$ be the generalized eigenvalues of the pencils $\mathcal{T}$ and $\mathcal{T}+\mathcal{N}(\sigma)\in\mathbb{K}^{R\times R\times 2}$, respectively[7]. Then based on the above discussion and assuming that the angular deviation between $\mathfrak{L}_k$ and $\hat{\mathfrak{L}}_k$ is normally distributed for each k, for a given variance σ^2 and probability p we may use the CRIB to determine the size of intervals $I_k(\sigma)$ around each $\mathfrak{L}_k$ such that one has $\hat{\mathfrak{L}}_k\in I_k(\sigma)$ for all k with probability bounded above by p. That is, for a given variance σ^2 we may determine a region around each generalized eigenvalue of $\mathcal{T}$ such that with high probability the corresponding generalized eigenvalue of the $\hat{\mathcal{T}}$ lies in that region.

We note that for each k, the size of $I_k(\sigma)$ is linear in σ. In particular, the square of the CRIB is obtained by taking the trace of a projection of the Cramér-Rao lower bound (CRLB) for $\mathbf{c}_k$. In general, the CRLB for a CPD is given by $\sigma^2\mathbf{J}^H\mathbf{J}$, where $\mathbf{J}^H\mathbf{J}$ is the Gramian of the Jacobian for computing a rank R CPD approximation to $\mathcal{T}+\mathcal{N}(\sigma)$, hence the square of the CRIB is quadratic in σ.

Let σ_* be the supremum over σ such that the collection of intervals $\{I_k(\sigma_*)\}$ may be split into two nonintersecting subcollections. That is, σ_* is the supremum over σ which satisfy that there are two distinct collections of indices J_1,J_2 such that $J_1\cup J_2=\{1,\dots,R\}$ and such that for all $j_1\in J_1$ and $j_2\in J_2$ one has $I_{j_1}(\sigma)\cap I_{j_2}(\sigma)=\emptyset$. Then σ_*^2 gives an approximation for the variance of the entries of $\mathcal{N}$ such that with probability p the generalized eigenvalues of the pencil $(\hat{\mathbf{T}}_1,\hat{\mathbf{T}}_2)$ may be divided into two distinct collections such that no generalized eigenvalue from either collection has coalesced with a generalized eigenvalue for the other. In other words, the generalized eigenspaces corresponding to these distinct collections cannot have coalesced.[8]

Having computed σ_* we may compute the expected value of the Frobenius norm[9]. of the noise tensor $\mathcal{N}(\sigma_*)$. Let $\mathbf{x}$ be a Gaussian white noise vector of length n whose entries have variance σ^2. Then one has

$$\mathbb{E}(\|\mathbf{x}\|_2)=\sqrt{2}\sigma\frac{\Gamma((n+1)/2)}{\Gamma(n/2)}\approx\sigma\sqrt{n}.$$

[7] If $\mathcal{T}+\mathcal{N}(\sigma)$ is slice mix invertible and has size $R\times R\times 2$, then assuming $\mathcal{T}+\mathcal{N}(\sigma)$ has a best rank R approximation is equivalent to assuming $\mathcal{T}+\mathcal{N}(\sigma)$ has rank R, hence $\mathcal{T}+\mathcal{N}(\sigma)=\hat{\mathcal{T}}$. See [22, Theorem 1].

[8] Since the CRIB is a lower bound, the probability that each $\hat{\mathfrak{L}}_k$ lies in the interval $I_k(\sigma_*)$ for each k may be less than p. However, the condition that each $\hat{\mathfrak{L}}_k$ lies in the interval $I_k(\sigma_*)$ is strictly stronger than the two collections of perturbed generalized eigenvalues being unable to coalesce. As such, our estimate has been impacted by both overestimation and underestimation.

[9] The upcoming deterministic bound for existence of two distinct groups of generalized eigenvalues which have not coalesced, see Corollary 4.4, computes an upper bound on the Frobenius norm of the noise tensor. Thus to compare these two bounds it is most appropriate to compare the expected value of the Frobenius norm of a mean zero error tensor with entry-wise variance σ_* to the deterministic bound of Corollary 4.4

Here, $\Gamma(\cdot)$ denotes the gamma function. Thus, if $\mathcal{T}$ has size $R \times R \times K$ then we find

$$\mathbb{E}(\|\mathcal{N}(\sigma_*)\|_{\mathrm{F}}) = \sqrt{2}\sigma_* \frac{\Gamma((R^2K+1)/2)}{\Gamma(R^2K/2)} \approx \sigma_* \sqrt{R^2K}. \tag{3.1}$$

In our upcoming numerical experiments we call the bound $\mathbb{E}(\|\mathcal{N}(\sigma_*)\|_{\mathrm{F}})$ as computed for a given p the **Cramér-Rao existence bound (CREB)**.

4. A deterministic bound for distinct eigenvalue clusters. In this section we give a deterministic bound for the existence of a subpencil of a tensor which has clusters of generalized eigenvalues that have not coalesced. When discussing this deterministic bound we restrict to the real setting. That is, in this section $\mathcal{T} \in \mathbb{R}^{R\times R\times K}$ is a slice mix invertible real tensor which has real rank R. See Section 4.3 for a brief discussion of the complex case.

We first introduce some notation. Define the **Grassmannian** $\mathrm{Gr}(1,\mathbb{K}^2)$ to be the collection of subspaces of $\mathbb{K}^2$ of dimension 1. Note that any element of $\mathrm{Gr}(1,\mathbb{R}^2)$ may be uniquely represented in the form $\mathrm{span}(\cos(\theta),\sin(\theta))$ for some $\theta \in [0,\pi)$. Thus given a subspace $\mathfrak{L} \in \mathrm{Gr}(1,\mathbb{R}^2)$ we define

$$\theta(\mathfrak{L}) \in [0,\pi) \qquad \text{by} \qquad \mathfrak{L} = \mathrm{span}\Big(\big(\cos(\theta(\mathfrak{L})),\sin(\theta(\mathfrak{L}))\big)\Big).$$

Additionally, for a nonzero vector $\boldsymbol{\lambda} \in \mathbb{R}^2$, define

$$\theta(\boldsymbol{\lambda}) = \theta(\mathrm{span}(\boldsymbol{\lambda})).$$

The metric we use for the Grassmannian is the chordal metric. Given two subspaces $\mathfrak{L}_1, \mathfrak{L}_2 \in \mathrm{Gr}(1,\mathbb{K}^2)$ define the **chordal metric** between $\mathfrak{L}_1$ and $\mathfrak{L}_2$, denoted $\chi(\mathfrak{L}_1,\mathfrak{L}_2)$ to be the sine of the (principal) angle between $\mathfrak{L}_1$ and $\mathfrak{L}_2$. It is straightforward to check that $\mathrm{Gr}(1,\mathbb{K}^2)$ is a path connected metric space under the chordal metric. We extend the definition of χ to $\mathbb{K}^2\backslash\{0\}$ by identifying nonzero vectors in $\mathbb{K}^2$ with their spans. That is, given $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2 \in \mathbb{K}^2\backslash\{0\}$ we have

$$\chi(\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2) := \chi(\mathrm{span}(\boldsymbol{\lambda}_1),\mathrm{span}(\boldsymbol{\lambda}_2)).$$

Our first step towards a bound for an $R \times R \times K$ tensor to have a subpencil in which generalized eigenvalues have not coalesced is to give a bound which guarantees that a matrix pencil (i.e., an $R \times R \times 2$ tensor) has generalized eigenvalues which have not coalesced.

THEOREM 4.1. *Let $\mathcal{T}$ and $\mathcal{Y}$ be tensors of size $R \times R \times 2$. Assume that $\mathcal{T}$ is slice mix invertible and has real rank R with CPD $\mathcal{T} = [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!]$, where the columns of $\mathbf{C}$ are normalized to have unit norm.*

Let $\{\mathrm{span}(\boldsymbol{\lambda}_1),\ldots,\mathrm{span}(\boldsymbol{\lambda}_r)\}$ be the generalized eigenvalues of $\mathcal{T}$ where the indices are ordered so that

$$0 \le \theta(\boldsymbol{\lambda}_r) \le \theta(\boldsymbol{\lambda}_{r+1}) < \pi \quad \text{for} \quad r = 1,\ldots,R-1.$$

Define $\boldsymbol{\lambda}_{R+1} = \boldsymbol{\lambda}_1$ and let δ be the Jth largest element of the set

$$\left\{ \frac{\chi(\boldsymbol{\lambda}_r,\boldsymbol{\lambda}_{r+1})}{2} \right\}_{r=1}^{R}.$$

Set $\epsilon_1 = \sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})\delta$ *and let* ϵ_2 *be the distance from* $\mathcal{T}$ *to the nearest pencil which is not slice mix invertible. If*

$$\|\mathbf{T}_{[1;3,2]} - \mathbf{Y}_{[1;3,2]}\|_2 < \min\{\epsilon_1, \epsilon_2\},$$

then the tensor $\mathcal{Y}$ *has at least* J *distinct (possibly complex valued) generalized eigenvalues.*

Furthermore, if one sets $\|\mathbf{T}_{[1;3,2]} - \mathbf{Y}_{[1;3,2]}\|_2 = \epsilon$*, then the generalized eigenvalues of* $\mathcal{Y}$ *may be divided into* J *nonempty sets* $\{\mathscr{L}^j\}_{j=1}^J$ *such that*

$$\min_{j\neq\ell} \chi(\mathscr{L}^j, \mathscr{L}^\ell) \geq 2\delta - \frac{2\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}. \tag{4.1}$$

The proof of Theorem 4.1 follows a simple idea. Using perturbation theoretic results for a matrix pencil, one may show that there are at least J distinct connected regions in $\mathrm{Gr}(1, \mathbb{C}^2)$ such that each region contains a generalized eigenvalue of $\mathcal{Y}$ and such that all generalized eigenvalues of $\mathcal{Y}$ are contained in the union of these disjoint regions. Roughly speaking, the "real parts" of these regions lie between each of the J largest chordal gaps between the ordered generalized eigenvalues of $\mathcal{T}$. To formalize this idea we introduce the notion of a Bauer–Fike ball and a Bauer–Fike region around a given generalized eigenvalue of $\mathcal{T}$.

Let the tensor $\mathcal{T} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$ and the generalized eigenvalues $\{\mathrm{span}(\boldsymbol{\lambda}_r)\}_{r=1}^R$ be as in the statement of Theorem 4.1. In particular, $\mathcal{T}$ is slice mix invertible with $\mathbb{R}$-rank R and the columns of $\mathbf{C}$ have unit norm. We define the (real or complex) **Bauer–Fike ball**, for the parameter ϵ around a given generalized eigenvalue $\mathrm{span}(\boldsymbol{\lambda}_r)$ of $\mathcal{T}$, denoted $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_r)$, by

$$\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_r) = \left\{ \mathrm{span}(\boldsymbol{\gamma}) \ \in \ \mathrm{Gr}(1, \mathbb{K}^2) : \boldsymbol{\gamma} \in \mathbb{K}^2/\{0\} \text{ and } \chi(\boldsymbol{\gamma}, \boldsymbol{\lambda}_r) < \frac{\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})} \right\}.$$

Using the notion of a Bauer–Fike ball[10], we define the **Bauer–Fike region** for the parameter ϵ around a generalized eigenvalue $\mathrm{span}(\boldsymbol{\lambda}_r)$, denoted $\mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_r)$ in the following manner. For each $N = 0, 1, 2, \ldots$ define

$$\mathrm{BF}_\epsilon^{\mathbb{K},N+1}(\boldsymbol{\lambda}_r) = \cup_{\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)\cap \mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell)\neq\emptyset} \mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell).$$

That is, $\mathrm{BF}_\epsilon^{\mathbb{K},N+1}(\boldsymbol{\lambda}_r)$ is the union over ℓ of Bauer–Fike balls $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell)$ such that $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell)$ has nontrivial intersection with $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$. The Bauer–Fike region $\mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_r)$ is defined by

$$\mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_r) = \cup_{N=1}^\infty \mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r).$$

It is not difficult to show that there is an integer N such that

$$\mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_r) = \mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r).$$

In particular one can take $N = R$. In words a Bauer–Fike region is a maximal union of Bauer–Fike balls such that the resulting set is path connected.

[10]For clarity we emphasize that a Bauer–Fike ball for the parameter ϵ around a generalized eigenvalue $\mathfrak{L}$ of $\mathcal{T}$ is simply the open neighborhood in $\mathrm{Gr}(1, \mathbb{K}^2)$ of radius $\frac{\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}$ centered at $\mathfrak{L}$. We introduce this notation to streamline the upcoming definition of Bauer–Fike regions, and to clearly illustrate the connection between these sets.

The complex Bauer–Fike regions for a given tensor $\mathcal{T}$ are useful in that they give us control over the locations of the generalized eigenvalues of a perturbation of $\mathcal{T}$. Assume ϵ is small enough so that $\|\mathbf{T}_{[1;3,2]} - \mathbf{Y}_{[1;3,2]}\|_2 < \epsilon$ implies that $\mathcal{Y}$ is slice mix invertible. Then following the argument in the proof of [11, Theorem 3.5] shows that the number of generalized eigenvalues of $\mathcal{Y}$ contained by each complex Bauer–Fike region for the parameter ϵ is equal to the number of overlapping complex Bauer–Fike balls which make up that complex Bauer–Fike region. That is, for a fixed index r, if there are exactly m indices ℓ (counting the case $r = \ell$) such that

$$\mathrm{BF}_\epsilon^{\mathbb{C}}(\boldsymbol{\lambda}_r) \cap \mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_\ell) \neq \emptyset,$$

and if $\|\mathbf{T}_{[1;3,2]} - \mathbf{Y}_{[1;3,2]}\|_2 < \epsilon$, then the complex Bauer–Fike region $\mathrm{BF}_\epsilon^{\mathbb{C}}(\boldsymbol{\lambda}_r)$ contains exactly m of the generalized eigenvalues of the tensor $\mathcal{Y}$ counting algebraic multiplicity.

Our goal in the proof of Theorem 4.1 will be to show that $\mathcal{T}$ has at least J distinct complex Bauer–Fike regions for the parameter ϵ. Following the above discussion shows that if $\mathcal{T}$ and $\mathcal{Y}$ satisfy the assumptions of Theorem 4.1, then $\mathcal{Y}$ has at least J distinct clusters of generalized eigenvalues which cannot have coalesced in a perturbation from $\mathcal{T}$ to $\mathcal{Y}$. To show that $\mathcal{T}$ has J distinct Bauer–Fike regions, we will make use of the fact that Bauer–Fike regions are connected subsets of $\mathrm{Gr}(1, \mathbb{K}^2)$.

LEMMA 4.2. *Let $\mathcal{T} \in \mathbb{R}^{R\times R\times 2}$ be a slice mix invertible $\mathbb{R}$-rank R tensor with generalized eigenvalues $\mathfrak{L} = \{\boldsymbol{\lambda}_r\}_{r=1}^R$. Then for each r, the Bauer–Fike region $\mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_r) \subset \mathrm{Gr}(1, \mathbb{K}^2)$ is path connected in the topology on $\mathrm{Gr}(1, \mathbb{K}^2)$ given by the chordal metric.*

Proof. We give the proof by induction on N in the definition of a Bauer–Fike region. First observe that $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_r)$ is the open neighborhood in $\mathrm{Gr}(1, \mathbb{K}^2)$ of chordal radius $\frac{\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}$ centered at $\mathrm{span}(\boldsymbol{\lambda}_r)$. It follows that $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_r)$ is path connected.

Now fix N and assume that $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$ is path connected. By definition we have that $\mathrm{BF}_\epsilon^{\mathbb{K},N+1}(\boldsymbol{\lambda}_r)$ is the union of $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$ with those $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell)$ which have nontrivial intersection with $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$. We know that $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell)$ is path connected for each ℓ, and by assumption $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$ is path connected, thus the union of $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$ with any $\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_\ell)$ having nontrivial intersection with $\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r)$ is path connected. It follows that $\mathrm{BF}_\epsilon^{\mathbb{K},N+1}(\boldsymbol{\lambda}_r)$ is path connected. Using the fact that there is an integer N such that

$$\mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_r) = \mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_r)$$

completes the proof. ∎

Another technical part of our proof will be to restrict from considering complex Bauer–Fike regions to considering real Bauer–Fike regions. We emphasize that the generalized eigenvalues of $\mathcal{Y}$ can only be guaranteed to lie in the union of the complex Bauer–Fike regions of $\mathcal{T}$, thus it is necessary to use complex Bauer–Fike regions. However, complex Bauer–Fike regions are more difficult to work with due to a lack of any notion of ordering in $\mathrm{Gr}(1, \mathbb{C}^2)$, hence our desire to reduce to working over $\mathbb{R}$. The following proposition makes this reduction possible.

PROPOSITION 4.3. *Let $\mathcal{T} \in \mathbb{R}^{R\times R\times 2}$ be a slice mix invertible tensor having real rank R. Let $\{\mathrm{span}(\boldsymbol{\lambda}_r)\}_{r=1}^R$ be the set of generalized eigenvalues of $\mathcal{T}$ with $\boldsymbol{\lambda}_r \in \mathbb{R}^2$ for each $r = 1, \ldots, R$. Then for any r_1, r_2 one has*

$$\mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_2}) = \emptyset \quad \text{if and only if} \quad \mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_2}) = \emptyset. \tag{4.2}$$

Additionally,

$$\mathrm{BF}_\epsilon^{\mathbb{C}}(\boldsymbol{\lambda}_{r_1}) = \mathrm{BF}_\epsilon^{\mathbb{C}}(\boldsymbol{\lambda}_{r_2}) \quad \text{if and only if} \quad \mathrm{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_{r_1}) = \mathrm{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_{r_2})$$

It follows that the number of distinct complex Bauer–Fike regions for $\mathcal{T}$ is exactly equal to the number of distinct real Bauer–Fike regions for $\mathcal{T}$.

Proof. Let $\mathcal{T}$ and $\boldsymbol{\lambda}_{r_1}$ and $\boldsymbol{\lambda}_{r_2}$ be as in the statement of the lemma. It is clear that $\mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_2}) = \emptyset$ implies $\mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_2}) = \emptyset$. On the other hand, using the fact that a vector $\boldsymbol{\zeta}$ which minimizes the chordal distance to a set of real vectors $\{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2\}$ may be obtained by taking $\boldsymbol{\zeta}$ to be a real vector that bisects the angle between $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$, it is straightforward to show that if $\mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_2}) \neq \emptyset$, then $\mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_2}) \neq \emptyset$. Thus we have $\mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_2}) = \emptyset$ if and only if $\mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_1}) \cap \mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_2}) = \emptyset$.

Using equation (4.2) together with the above, one may show by induction that for all $N \in \mathbb{N}$ and all $r_1, r_2 \in \{1, \ldots, R\}$ one has

$$\mathrm{BF}_\epsilon^{\mathbb{C},0}(\boldsymbol{\lambda}_{r_2}) \subset \mathrm{BF}_\epsilon^{\mathbb{C},N}(\boldsymbol{\lambda}_{r_1}) \quad \text{if and only if} \quad \mathrm{BF}_\epsilon^{\mathbb{R},0}(\boldsymbol{\lambda}_{r_2}) \subset \mathrm{BF}_\epsilon^{\mathbb{R},N}(\boldsymbol{\lambda}_{r_1}). \tag{4.3}$$

Furthermore, it is straightforward to check that

$$\mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_{r_1}) = \mathrm{BF}_\epsilon^{\mathbb{K}}(\boldsymbol{\lambda}_{r_2})$$

if and only if there is an integer N such that

$$\mathrm{BF}_\epsilon^{\mathbb{K},0}(\boldsymbol{\lambda}_{r_2}) \subset \mathrm{BF}_\epsilon^{\mathbb{K},N}(\boldsymbol{\lambda}_{r_1}).$$

Combining this fact with equation (4.3) completes the proof. ∎

4.1. Proof of Theorem 4.1. We have now laid the necessary ground work to give the proof of Theorem 4.1. The proof is technical, but, as previously mentioned, the main idea behind the proof is simple. Before giving technical details, we give an overview of the role played by Bauer–Fike regions.

As previously discussed, each distinct Bauer–Fike region for $\mathcal{T}$ contains a generalized eigenvalue of $\mathcal{Y}$ and all the generalized eigenvalues of $\mathcal{Y}$ are contained in some Bauer–Fike region of $\mathcal{T}$. By using this fact together with Proposition 4.3 to show that $\mathcal{Y}$ has at least J distinct generalized eigenvalues, it is sufficient to show that $\mathcal{T}$ has J distinct real Bauer–Fike regions.

To show that $\mathcal{T}$ has at least J distinct real Bauer–Fike regions, we divide $\mathrm{Gr}(1, \mathbb{R}^2)$ into J distinct regions such that each region fully contains a real Bauer–Fike region of $\mathcal{T}$. The desired division of $\mathrm{Gr}(1, \mathbb{R}^2)$ into distinct regions is accomplished by using subspaces of $\mathbb{R}^2$ to cut in half each of the J largest gaps between the ordered generalized eigenvalues of $\mathcal{T}$.

Our choice of parameter for our Bauer–Fike balls will guarantee that these cuts cannot be contained in any Bauer–Fike ball of $\mathcal{T}$, hence, these cuts cannot be contained in any Bauer–Fike region. The fact that a real Bauer–Fike region lying between two adjacent cuts is distinct from a real Bauer–Fike region lying between another pair of adjacent cuts follows from the path connectedness of the real Bauer–Fike regions. Details follow below.

Proof. Let $\mathcal{T} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$ and J and ϵ be as in the statement of Theorem 4.1 and assume $J \geq 2$. We first show that $\mathcal{T}$ has J distinct complex Bauer–Fike regions. Using Proposition 4.3, it is sufficient to show that $\mathcal{T}$ has J distinct real Bauer–Fike regions.

Now, from the ordering of our eigenvalues, there must exist J distinct indices $r_1, \ldots, r_J \in \{1, \ldots, R\}$ such that $r_j < r_{j+1}$ for each j and such that for each $r_j \neq R$ there is a unit vector $\boldsymbol{\eta}_j \in \mathbb{R}^2$ satisfying

$$\theta(\boldsymbol{\lambda}_{r_j}) < \theta(\boldsymbol{\eta}_j) < \theta(\boldsymbol{\lambda}_{r_j+1})$$

and

$$\chi(\boldsymbol{\eta}_j, \boldsymbol{\lambda}_{r_j}) \geq \delta \qquad \text{and} \qquad \chi(\boldsymbol{\eta}_j, \boldsymbol{\lambda}_{r_j+1}) \geq \delta.$$

Using $\epsilon_1 = \sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})\delta$ gives

$$\chi(\boldsymbol{\eta}_\ell, \boldsymbol{\lambda}_{r_\ell}) > \frac{\epsilon_1}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})} \qquad \text{and} \qquad \chi(\boldsymbol{\eta}_\ell, \boldsymbol{\lambda}_{r_\ell+1}) > \frac{\epsilon_1}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}.$$

If the case $r_J = R$ occurs, then we may instead choose a unit vector $\boldsymbol{\eta}_J$ such that

$$\chi(\boldsymbol{\eta}_J, \boldsymbol{\lambda}_R) > \frac{\epsilon_1}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})} \qquad \text{and} \qquad \chi(\boldsymbol{\eta}_J, \boldsymbol{\lambda}_1) > \frac{\epsilon_1}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}$$

and such that either

$$\theta(\boldsymbol{\lambda}_R) < \theta(\boldsymbol{\eta}_J) \qquad \text{or} \qquad \theta(\boldsymbol{\eta}_J) < \theta(\boldsymbol{\lambda}_1).$$

To simplify exposition, we assume that $\theta(\boldsymbol{\lambda}_R) < \theta(\boldsymbol{\eta}_J)$.

Note that the $\operatorname{span}(\boldsymbol{\eta}_j)$ divide $\operatorname{Gr}(1, \mathbb{R}^2)$ into J distinct open connected components. In particular, there exist J disjoint open subsets $\mathscr{G}_1, \ldots, \mathscr{G}_J$ of $\operatorname{Gr}(1, \mathbb{R}^2)$ such that

$$\cup_{j=1}^{J} \mathscr{G}_j = \operatorname{Gr}(1, \mathbb{R}^2) \backslash \left(\cup_{j=1}^{J} \operatorname{span}(\boldsymbol{\eta}_j) \right).$$

Using the ordering of our generalized eigenvalues together with the definition of the $\boldsymbol{\eta}_j$, a straightforward check shows that $\operatorname{span}(\boldsymbol{\eta}_j)$ cannot be contained in a real Bauer–Fike region $\operatorname{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_r)$ for any $r = 1, \ldots, R$. In particular we have

$$\cup_{r=1}^{R} \operatorname{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_r) \subset \left(\cup_{j=1}^{J} \mathscr{G}_j \right) \tag{4.4}$$

Recalling that each Bauer–Fike region is connected and that $\mathscr{G}_j$ are open and disjoint sets, equation (4.4) then implies that if there are indices r and j such that

$$\operatorname{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_r) \cap \mathscr{G}_j \neq \emptyset \qquad \text{then} \qquad \operatorname{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_r) \subseteq \mathscr{G}_j. \tag{4.5}$$

By construction we have

$$\mathscr{G}_j \cap \left(\cup_{r=1}^{R} \operatorname{BF}_\epsilon^{\mathbb{R}}(\boldsymbol{\lambda}_r) \right) \neq \emptyset \qquad \text{for each } j = 1, \ldots, J.$$

Combining this with equation (4.5) shows that each $\mathscr{G}_j$ contains some real Bauer–Fike region of $\mathcal{T}$. In particular, $\mathcal{T}$ has at least J distinct real Bauer–Fike regions, hence $\mathcal{T}$ has at least J distinct complex Bauer–Fike regions.

Now let $\mathcal{Y} \in \mathbb{R}^{R \times R \times 2}$ be a tensor which satisfies

$$\|\mathbf{T}_{[1;3,2]} - \mathbf{Y}_{[1;3,2]}\|_2 \leq \epsilon < \min\{\epsilon_1, \epsilon_2\}.$$

Following the argument used to prove [11, Theorem 3.5] shows that for each $j = 1, \ldots, J$ the region $\operatorname{BF}_\epsilon^{\mathbb{C}}(\boldsymbol{\lambda}_{r_j})$ must contain at least one generalized eigenvalue of $\mathcal{Y}$ and that all the generalized eigenvalues of $\mathcal{Y}$ are contained in some Bauer–Fike region. Therefore, since $\mathcal{T}$ has at least J distinct Bauer–Fike regions, the tensor $\mathcal{Y}$ has at least J distinct generalized eigenvalues.

It remains to construct the nonempty sets $\{\mathscr{L}^j\}_{j=1}^{J}$ of generalized eigenvalues of $\mathcal{Y}$ which satisfy equation (4.1). To this end, let $\{\operatorname{span}(\tilde{\boldsymbol{\lambda}}_r)\}_{r=1}^{R}$ be the set of generalized eigenvalues of $\mathcal{Y}$ and for each $j = 1, \ldots, J$ define $\mathscr{L}^j$ by

$$\mathscr{L}^j = \left\{ \operatorname{span}(\tilde{\boldsymbol{\lambda}}_\ell) \mid \tilde{\boldsymbol{\lambda}}_\ell \in \{\tilde{\boldsymbol{\lambda}}_r\}_{r=1}^{R} \cap \mathscr{G}_j \right\}.$$

In words, $\mathscr{L}^j$ is the set of generalized eigenvalues of $\mathcal{Y}$ which are contained in $\mathscr{G}_j$.

Now suppose $\tilde{\boldsymbol{\lambda}}_m$ and $\tilde{\boldsymbol{\lambda}}_n$ are generalized eigenvalues of $\mathcal{Y}$ such that $\tilde{\boldsymbol{\lambda}}_m \in \mathscr{G}_i$ and $\tilde{\boldsymbol{\lambda}}_n \in \mathscr{G}_\ell$ for some indices $i \neq \ell$. Then there must exist generalized eigenvalues $\boldsymbol{\lambda}_m$ and $\boldsymbol{\lambda}_n$ of $\mathcal{T}$ such that $\boldsymbol{\lambda}_m \in \mathscr{G}_i$ and $\boldsymbol{\lambda}_n \in \mathscr{G}_\ell$ and

$$\chi(\boldsymbol{\lambda}_m, \tilde{\boldsymbol{\lambda}}_m) \leq \frac{\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})} \qquad \text{and} \qquad \chi(\boldsymbol{\lambda}_n, \tilde{\boldsymbol{\lambda}}_n) \leq \frac{\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}.$$

Furthermore, from the definition of the $\mathscr{G}_j$ we must have

$$\chi(\boldsymbol{\lambda}_m, \boldsymbol{\lambda}_n) \geq 2\delta.$$

Using the triangle inequality shows

$$\chi(\tilde{\boldsymbol{\lambda}}_m, \tilde{\boldsymbol{\lambda}}_n) \geq \chi(\boldsymbol{\lambda}_m, \boldsymbol{\lambda}_n) - \chi(\boldsymbol{\lambda}_m, \tilde{\boldsymbol{\lambda}}_m) - \chi(\boldsymbol{\lambda}_n, \tilde{\boldsymbol{\lambda}}_n) \geq 2\delta - \frac{2\epsilon}{\sigma_{\min}(\mathbf{A})\sigma_{\min}(\mathbf{B})}$$

from which inequality (4.1) follows. ∎

4.2. Distinct eigenvalue clusters for subpencils of a tensor. We now use Theorem 4.1 to give the radius of a Frobenius norm ball around a given tensor in which every tensor is guaranteed to have a subpencil with at least J distinct generalized eigenvalues.

COROLLARY 4.4. *Let $\mathcal{T} \in \mathbb{R}^{R\times R\times K}$ be a slice mix invertible tensor of real rank R. Let $\mathbf{U} \in \mathbb{R}^{K\times K}$ be any orthogonal matrix. Set $\mathcal{S} = \mathcal{T} \cdot_3 \mathbf{U}$. For each $k = 1, \ldots, \lfloor K/2 \rfloor$, let $\epsilon_i \geq 0$ be the bound obtained from Theorem 4.1 which guarantees that matrix pencils in a ball of Frobenius radius ϵ_k around the subpencil*

$$(\mathbf{S}_{2k-1}, \mathbf{S}_{2k})$$

have at least J distinct generalized eigenvalues. Here we take $\epsilon_k = 0$ if the corresponding subpencil is not slice mix invertible. Set $\epsilon = ||(\epsilon_1, \ldots, \epsilon_{\lfloor K/2 \rfloor})||_2$. Then every tensor in a ball of Frobenius radius ϵ around $\mathcal{T}$ has a subpencil which has at least J distinct generalized eigenvalues.

Proof. Let $\mathcal{Y} \in \mathbb{R}^{R\times R\times K}$ be a tensor such that $\|\mathcal{T} - \mathcal{Y}\|_{\mathrm{F}} < \epsilon$ and set $\mathcal{P} = \mathcal{Y} \cdot_3 \mathbf{U}$. Then we have $\|\mathcal{S} - \mathcal{P}\|_{\mathrm{F}} < \epsilon$. It follows that there must be some index $k \in \{1, \ldots, \lfloor K/2 \rfloor\}$ such that

$$\|(\mathbf{S}_{2k-1}, \mathbf{S}_{2k}) - (\mathbf{P}_{2k-1}, \mathbf{P}_{2k})\|_2 \leq \|(\mathbf{S}_{2k-1}, \mathbf{S}_{2k}) - (\mathbf{P}_{2k-1}, \mathbf{P}_{2k})\|_{\mathrm{F}} < \epsilon_k,$$

hence Theorem 4.1 may be applied to the subpencil $(\mathbf{P}_{2k-1}, \mathbf{P}_{2k})$ of $\mathcal{P}$. ∎

Theorem 4.1 and Corollary 4.4 are similar to results appearing in [11]; however, the results appearing here are more general and their proofs require additional consideration.

4.3. Deterministic bounds over the complex field. We now briefly consider the complex setting. In this setting one may still use Bauer–Fike regions to determine where the generalized eigenvalues of a perturbation may lie. However, when working with a matrix pencil which has complex generalized eigenvalues, there is no natural way to order the generalized eigenvalues, hence one cannot simply consider the Jth largest element of the set

$$\left\{ \frac{\chi(\boldsymbol{\lambda}_r, \boldsymbol{\lambda}_{r+1})}{2} \right\}_{r=1}^{R}.$$

as in Theorem 4.1.

As an alternative, one could use Bauer–Fike regions together with a binary search to determine the parameter ϵ for Bauer–Fike balls which guarantees the existence of J distinct generalized eigenvalue clusters in each subpencil of a given tensor. The resulting bounds for each subpencil may then be used as the ϵ_i in Corollary 4.4.

5. GESD vs. GEVD accuracy and timing. We now compare both the accuracy and timing of GESD to GEVD where the GEVD solution is computed using the first two slices of the MLSVD of the tensor[11]. To this end, we generate low-rank tensors and add noise with varying signal to noise ratios (SNRs). We then compute the CPD with both methods and compare the computation time and factor matrix error (`cpderr`) of the results. For a low-rank tensor generated from the factor matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, the factor matrix error of its estimated CPD $\left[\!\left[\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}\right]\!\right]$ is defined as $\max\left(\frac{\|\mathbf{A}-\hat{\mathbf{A}}\|_{\mathrm{F}}}{\|\mathbf{A}\|_{\mathrm{F}}}, \frac{\|\mathbf{B}-\hat{\mathbf{B}}\|_{\mathrm{F}}}{\|\mathbf{B}\|_{\mathrm{F}}}, \frac{\|\mathbf{C}-\hat{\mathbf{C}}\|_{\mathrm{F}}}{\|\mathbf{C}\|_{\mathrm{F}}}\right)$, where columns of $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ have been optimally permuted and scaled to match the columns of $\mathbf{A}$,$\mathbf{B}$ and $\mathbf{C}$, respectively. The SNR of the tensor $\mathcal{T}$ and the noise $\mathcal{N}$ in dB is defined as $20\log_{10}\left(\frac{\|\mathcal{T}\|_{\mathrm{F}}}{\|\mathcal{N}\|_{\mathrm{F}}}\right)$. For each SNR, the median error and computation time are computed over 50 trials unless differently specified. The computation time is the time required to obtain the factor matrices $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$ and $\hat{\mathbf{C}}$ starting from the noisy tensor $\mathcal{T}+\mathcal{N}$. All experiments in this section and Section 6 are performed on a machine with an Intel Core i7-6820HQ CPU at 2.70GHz and 16GB of RAM using MATLAB R2020b and Tensorlab 3.0 [29].

We start with two easier problems where the tensor dimensions are considerably larger than the tensor rank. As a more challenging case, we then move on to tensors with factor matrices that have correlated columns. Next, we examine the accuracy and speed of the GESD method compared to a state-of-the art optimization method. In addition we provide supplementary materials which contain numerical experiments that demonstrate the efficacy of the proposed method in accurately extracting spectral-like components.

We note that as a tuning parameter in GESD one may adjust the angle between generalized eigenvalues which is sufficiently large for splitting between the corresponding generalized eigenspaces. To a point, the accuracy of GESD may be improved by increasing the requested angle; however, increasing this angle can decrease the speed of GESD. One could also choose to restrict the number of generalized eigenvalue clusters which GESD is allowed to separate in at each step. The optimal splitting strategy in terms of both timing and accuracy is a topic of further investigation. Furthermore, one could additionally consider the separation between computed generalized eigenspaces as a parameter for deciding if a particular split is acceptable.[12]

5.1. Tensor dimension exceeds tensor rank. The following experiments resemble typical problems in signal processing, where the tensor dimensions are often orders of magnitude larger than the tensor rank. For the first experiment, the factor matrix entries of a $100 \times 100 \times 100$-tensor of rank 10 are independently sampled from

[11] Similar to GESD, the average performance of GEVD benefits by using the first two slices of the MLSVD. However, this strategy is not always successful, see Section 5.4.

[12] An in-depth discussion of the sensitivity of generalized eigenspace computations to the separation between the corresponding generalized eigenspaces is given in [7]. While the bounds discussed therein take into account both eigenvalue and eigenspace conditioning, they are expensive to compute and lead to major increases in runtime if used in GESD. This is of course undesirable if the GESD algorithm is used to quickly obtain a good initialization for an optimization algorithm. [12] further explores this topic in the context of joint generalized eigenspaces.

the standard normal distribution. The results are shown in Figure 1. In this setting, the GESD algorithm is marginally more accurate than GEVD, but the computational cost of GESD is about double that of GEVD. While GESD is more computationally expensive, the first step in this setting is to compress the starting tensor. The cost of this compression dominates the other parts of the computation, thus using GESD over GEVD leads to only a small increase in the total run time of the algorithm.

For the second experiment, the factor matrices are independently sampled from the uniform distribution on $[0, 1]$ instead of the standard normal distribution. This setting is more challenging than the first since the columns of the factor matrices are less well separated. In Figure 2, we can see the accuracies and computation times of both methods. Compared to the case with normally distributed factor matrix entries, a higher SNR is needed to obtain a reasonable solution, but from an SNR of 0 dB upwards, the error of the GESD method is more than an order of magnitude smaller than the error for the GEVD method. That is, using GESD leads to a gain of over 20 dB in accuracy. As in the previous experiment, the total cost of compression together with GESD is only slightly higher than compression with GEVD since compression is the most computationally expensive step.

5.2. Factor matrices with correlated columns. We now consider the more challenging setting where the tensor of interest has rank equal to dimensions. That is, the tensors we consider in this section have rank R and size $R \times R \times R$. This case is significantly more difficult than the $I_1 \times I_2 \times I_3$ rank R case with $I_1, I_2, I_3 \gg R$ which generally occurs in practice and is discussed in Section 5.1. Recall that the first step in both GESD and GEVD is to compute a MLSVD of the tensor of interest. In the $I_1, I_2, I_3 \gg R$ case, computing this MLSVD and reducing to an $R \times R \times R$ tensor typically significantly increases the SNR of the problem thereby reducing the difficulty of the problem. In addition, the compression generally takes much more time than computing the CPD of the small core tensor using GEVD or GESD.

In Figure 3 we consider the particularly difficult case where the columns of the factor matrices are highly correlated. More concretely, we consider the case where the first column of $\mathbf{A}$ is sampled from the uniform distribution on $[0, 1]$ as before and each successive column of $\mathbf{A}$ is randomly sampled so that it forms a 10° angle with $\mathbf{a}_1$. The other factor matrices are sampled analogously. Results for this case are shown in Figure 3.

For the problems in Figure 3, we see that GESD is significantly more accurate than GEVD. In fact, GESD is able to compute meaningful solutions starting at about an order of magnitude lower SNR than GEVD. This demonstrates that the handling of pencil specific eigenvalue clusters is better in GESD than in GEVD due to the fact that GESD is able to compute an eigenspace for the cluster, but delays computation of eigenvectors to a pencil in which the previously clustered eigenvalues are better separated. As before, the GESD algorithm takes about three times as long as the GEVD algorithm.

In Figure 4, we investigate the speed and accuracy of both algorithms as a function of tensor rank. Here we run GESD and GEVD on noiseless tensors of rank R and size $R \times R \times R$ where R ranges from 50 to 350 and where the factor matrices are independently sampled from the uniform distribution on $[0, 1]$. For all values of R we see that GESD is roughly two orders of magnitude more accurate than GEVD, but that GESD takes about three times as long to run as GEVD.

5.3. Comparison with optimization. As the GEVD algorithm can quickly compute a reasonably good fit for the CPD, its solution is often used as an initializa-

tion for an optimization algorithm. By providing a good initialization, the number of iterations of the more accurate yet more computationally expensive optimization algorithm can ideally be lowered significantly. In the next experiment, we compare the error and computation time of a nonlinear least-squares (NLS) algorithm [28] for three different initialization strategies: starting from the GEVD solution, the GESD solution or a random initial point. The optimization algorithm is run until either the relative cost function change or the relative factor matrix change drops under 10^{-8}. The time to compute the GEVD or GESD solutions is included in the runtime of the optimization algorithm.

In Figure 5, median values are shown over 20 trials for a $10 \times 10 \times 10$-tensor of rank 10 with factor matrices independently sampled from the uniform distribution on $[0, 1]$. We can see that the NLS algorithm generally converges to a good solution for all three initialization strategies. The GEVD solution is notably less accurate than either a GESD or optimization based solution. In the right plot of Figure 5, it can be seen that the GESD method is about an order of magnitude faster than the optimization algorithm. Also, the computation time of the optimization algorithm is significantly reduced by starting from the GEVD or GESD initialization, even when their computation time is included. For all SNRs, computation of a GESD or GEVD initialization followed by optimization is equally fast.

In Figure 6, median values are shown over 20 trials for a $10 \times 10 \times 10$-tensor of rank 10 where the columns of the factor matrices are sampled with a fixed 10° angle between them. For this more challenging problem, we can see from the left plot of Figure 6 that decent algebraic initializations are only achieved at higher SNR values compared to the uncorrelated case. Additionally we see that the SNR required for GESD to compute reasonably accurate initializations is significantly lower than that for GEVD. In this experiment, random initialization of the optimization algorithm does not lead to a meaningful solution at all, which again highlights the importance of algebraic initialization. As before, optimization with a GESD initialization and optimization with a GEVD initialization typically take the same amount of time.

These experiments illustrate that using GESD as an algebraic initialization compared to GEVD does not cause any loss in terms of time or accuracy. Furthermore see we that GESD based initializations are significantly more accurate than GEVD based initializations. Thus using GESD over GEVD to initialize gives more reliable initializations without impacting total computation time.

5.4. GESD and GEVD vs. noiseless problems. We now illustrate an additional advantage of GESD compared to GEVD. Namely there is no fixed strategy for pencil selection in GEVD which is able to solve all problems meeting the assumptions of the algorithm even in the noiseless setting. As an example, let $\mathcal{T}$ be the (real) rank 3 tensor with frontal slices

$$\mathbf{T}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \mathbf{T}_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/2 \end{pmatrix} \quad \mathbf{T}_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/5 \\ 0 & 1/5 & 0 \end{pmatrix}.$$

As previously discussed, a popular strategy for pencil selection in GEVD is to use the first two slices of a core for the MLSVD of $\mathcal{T}$. The tensor $\mathcal{S}$ with frontal slices

$$\mathbf{S}_1 \approx -\begin{pmatrix} 1.39 & 0 & 0 \\ 0 & 1.11 & 0 \\ 0 & 0 & 1.11 \end{pmatrix} \quad \mathbf{S}_2 \approx \begin{pmatrix} -0.252 & 0 & 0 \\ 0 & 0.157 & 0 \\ 0 & 0 & 0.157 \end{pmatrix} \quad \mathbf{S}_3 \approx -\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1/5 \\ 0 & 1/5 & 0 \end{pmatrix}$$

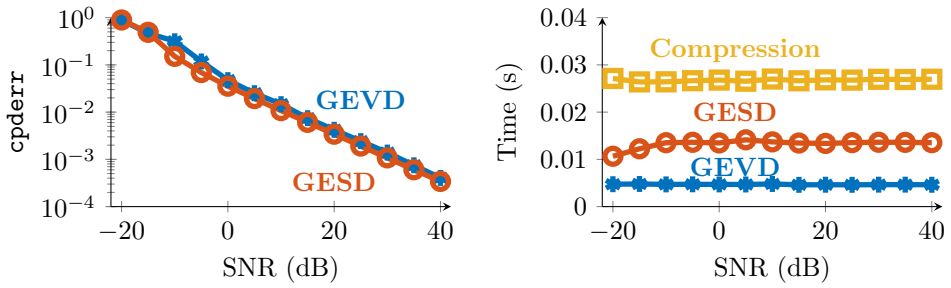


FIG. 1. *For normally distributed factor matrices, the GESD and GEVD methods have similar performance with respect to the factor matrix accuracy. Median (over* 50 *trials)* `cpderr` *(left) and computation time (right) for rank* 10 *tensors of dimensions* $100 \times 100 \times 100$ *with factor matrices sampled from the standard normal distribution. The runtime of GESD on the core tensor is about three times that of the GEVD on the core; however overall runtime is dominated by the compression step.*

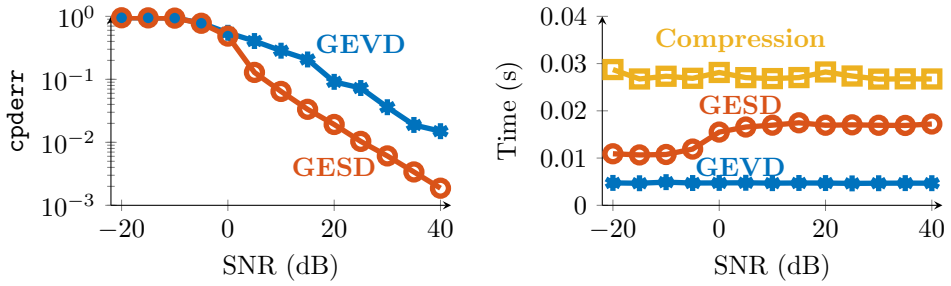


FIG. 2. *For uniformly distributed factor matrices, the GESD method obtains a higher accuracy for the estimated factor matrices than the GEVD method. Median (over* 50 *trials)* `cpderr` *(left) and computation time (right) for rank* 10 *tensors of dimensions* $100 \times 100 \times 100$ *with factor matrices sampled from the uniform distribution on* $[0, 1]$.

is a core for the MLSVD of $\mathcal{T}$. If one uses the first two slices of $\mathcal{S}$ to compute the CPD of $\mathcal{T}$ via GEVD, then one must compute the eigenvectors of the matrix

$$\mathbf{S}_1^{-1}\mathbf{S}_2 \approx -\begin{pmatrix} 0.181 & 0 & 0 \\ 0 & 0.143 & 0 \\ 0 & 0 & 0.143 \end{pmatrix}.$$

Since $\operatorname{span}\{\mathbf{e}_2, \mathbf{e}_3\}$ is an eigenspace of dimension 2 for this matrix corresponding to the eigenvalue -0.143, there is an ambiguity in which eigenvectors should be used. Tensorlab, for example, assumes $\mathbf{e}_2$ and $\mathbf{e}_3$ are the correct eigenvectors to take in this subspace and proceeds from there. Instead considering the matrix pencil $(\mathbf{S}_1, \mathbf{S}_3)$ shows that $\mathbf{e}_2 + \mathbf{e}_3$ and $\mathbf{e}_2 - \mathbf{e}_3$ are the correct eigenvectors to take from the subspace $\operatorname{span}\{\mathbf{e}_2, \mathbf{e}_3\}$.

To make matters worse for GEVD, if one proceeds under the assumption that $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ are the correct eigenvectors to take, then the GEVD based solution one arrives at is a local optima for the optimization based approach for computing a CPD. Thus using this GEVD based initialization followed by optimization cannot

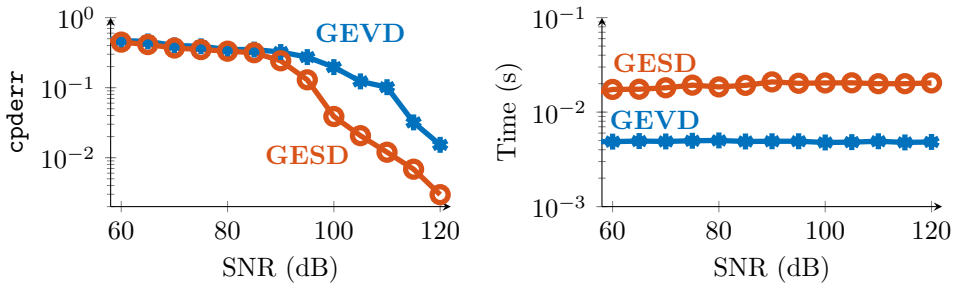


FIG. 3. *For factor matrices with correlated columns, the GESD method performs significantly better than the GEVD method with respect to the factor accuracy. Median (over* 50 *trials)* `cpderr` *(left) and computation time (right) for rank* 10 *tensors of dimensions* $10 \times 10 \times 10$ *with factor matrices with correlated columns* (10° *angles). Because of the difficulty of the problem only very high SNR values lead to a decent solution. The computation time of the GESD method is about three times that of the GEVD method.*

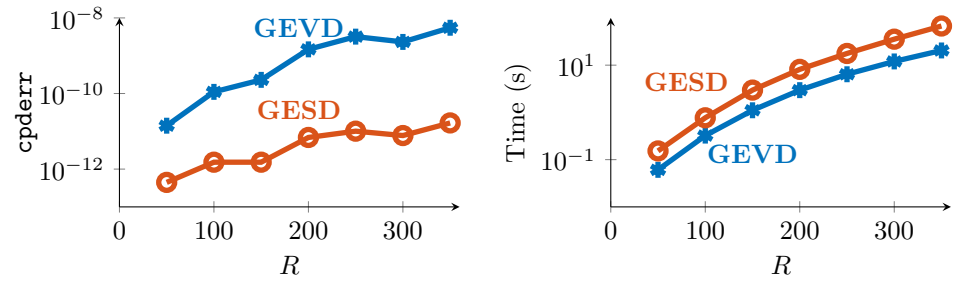


FIG. 4. *The relative time difference between the GESD and GEVD methods remains about the same as tensor rank increases. Median (over* 20 *trials)* `cpderr` *and computation time for exact rank* R *tensors of dimensions* $R \times R \times R$ *with factor matrices sampled from the uniform distribution on* $[0, 1]$*. The GESD method obtains factors with a higher accuracy than the GEVD method for all tensor dimensions.*

yield the true rank 3 decomposition of $\mathcal{T}$.[13]

GESD, on the other hand, is easily able to compute the decomposition of this tensor. Although GESD still uses the pencil $(\mathbf{S}_1, \mathbf{S}_2)$ on its first iteration, GESD recognizes that the eigenspace $\text{span}\{\mathbf{e}_2, \mathbf{e}_3\}$ cannot be decomposed with this pencil and delays the decomposition to a later pencil.

6. Stability of GESD vs. the CREB. In this section we compare the CREB and deterministic bound computed in Sections 3 and 4 for the existence of two distinct eigenvalue clusters against the accuracy of the first step of our GESD algorithm. We note that the CREB and deterministic bound computed from the original tensor only provide information about the first step of our algorithm. For this reason, we omit a direct comparison of these bounds to the accuracy of the GESD algorithm.

Let $\mathcal{T} \in \mathbb{R}^{R \times R \times K}$ be a real rank R tensor with CPD $[\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$, and let $\mathcal{N}$ be a noise tensor. As previously discussed, given $\mathcal{T} + \mathcal{N}$ as an input, the goal of the first step of the GESD algorithm is to robustly decompose $\mathbb{R}^R$ into (at least) two disjoint JGE spaces of $\mathcal{T}$.

[13]The GEVD based decomposition has approximately 0.516 factor matrix error when compared to the correct decomposition.

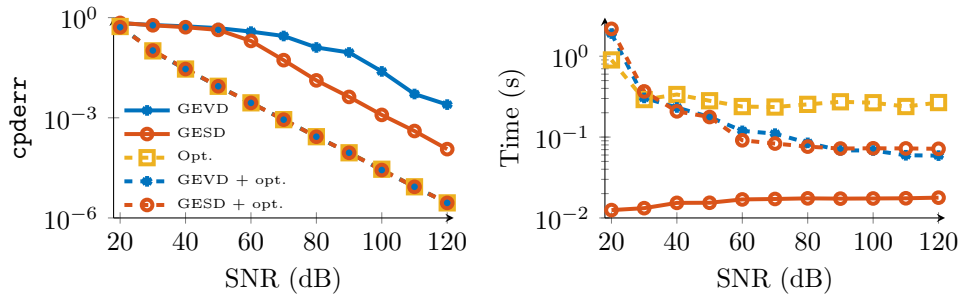


FIG. 5. *Initializing an optimization algorithm with the GESD or GEVD solution is faster than starting from a random initialization. Median (over* 20 *trials)* `cpderr` *(left) and computation time (right) for rank* 10 *tensors of dimensions* $10 \times 10 \times 10$ *with factor matrices sampled from the uniform distribution on* $[0, 1]$*. All initializations lead to the same solution. Optimization after initializing with the more accurate GESD solution is equally fast as optimization after initializing with the GEVD solution.*

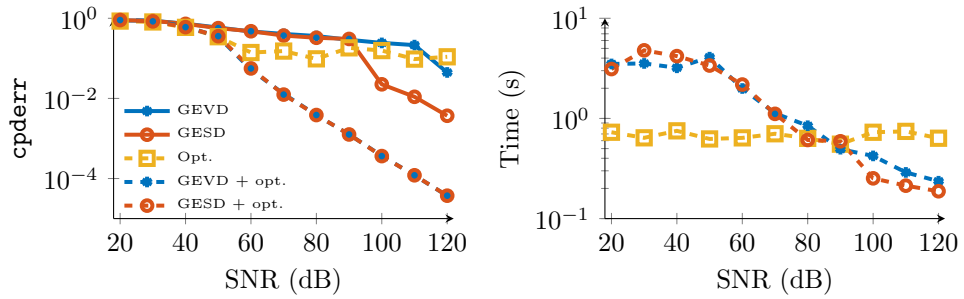


FIG. 6. *For factor matrices with correlated columns, the GESD solution is more accurate than the GEVD solution. An optimization algorithm initialized with the GESD solution is faster than when initialized with the GEVD solution. Median (over* 20 *trials)* `cpderr` *(left) and computation time (right) for rank* 10 *tensors of dimensions* $10 \times 10 \times 10$ *with correlated factor matrix columns* (10° *angles). Initializing the optimization algorithm randomly does not lead to a good solution.*

Suppose $\mathbf{Z}_1$ and $\mathbf{Z}_2$ are matrices such that the range of each $\mathbf{Z}_i$ is a JGE space of $\mathcal{T}$ and such that the matrix $(\mathbf{Z}_1\ \mathbf{Z}_2)$ is invertible. Recall that the $\mathbf{Z}_i$ have the form $\mathbf{Z}_i = \mathbf{B}^{-\mathrm{T}}\mathbf{E}_i$ where $\mathbf{E}_1$ and $\mathbf{E}_2$ are matrices which for all $j = 1, \ldots, R$ satisfy that the jth row of $\mathbf{E}_1$ is nonzero if and only if the jth row of $\mathbf{E}_2$ is a zero row. Using this fact we may evaluate the robustness of the first step of our algorithm in two ways.

Let $\tilde{\mathbf{Z}}_1$ and $\tilde{\mathbf{Z}}_2$ be the matrices computed by the first step of our algorithm. Then one should have that the ranges of

$$\mathbf{B}^{\mathrm{T}}\tilde{\mathbf{Z}}_1 = \tilde{\mathbf{E}}_1 \qquad \text{and} \qquad \mathbf{B}^{\mathrm{T}}\tilde{\mathbf{Z}}_2 = \tilde{\mathbf{E}}_2.$$

are (approximately) orthogonal. Additionally, there should exist disjoint index sets $J_1, J_2 \subset \{1, \ldots, R\}$ such that $J_1 \cup J_2 = \{1, \ldots, R\}$ and such that

$$\text{ran } \tilde{\mathbf{Z}}_1 \approx \text{span}(\{\mathbf{b_j}\}_{\mathbf{j}\in\mathbf{J_1}}) \qquad \text{and} \qquad \text{ran } \tilde{\mathbf{Z}}_\mathbf{2} \approx \text{span}(\{\mathbf{b_j}\}_{\mathbf{j}\in\mathbf{J_2}})$$

where the $\mathbf{b_j}$ are columns of the matrix $(\mathbf{B}^{-\mathrm{T}})$. That is, the principal (largest) angle between the range of $\tilde{\mathbf{Z}}_i$ and $\text{span}(\{\mathbf{b_j}\}_{\mathbf{j}\in\mathbf{J_i}})$ should be approximately zero.

6.1. Figure for GESD stability. In Figure 7, we plot the median (over 50 trials) of the aforementioned angles as a function of the SNR of $\mathcal{T}$ to $\mathcal{N}$ where $\mathcal{N}$

is randomly generated Gaussian noise. That is, for many choices of the SNR s, we randomly generate noise tensors $\mathcal{N}$ such that the SNR of $\mathcal{T}$ to $\mathcal{N}$ is equal to s. In each trial we compute subspaces $\tilde{\mathbf{Z}}_1$ and $\tilde{\mathbf{Z}}_2$ for the tensor $\mathcal{T}+\mathcal{N}$ using the GESD algorithm. We check accuracy by plotting the median over the trials of the angles described above. As before, $\mathcal{T}$ has been generated by independently sampling all entries of the factor matrices $\mathbf{A}, \mathbf{B}$, and $\mathbf{C}$ uniformly on $[0, 1]$ and forming the associated low-rank tensor.

The first image of Figure 7 shows the $R \times R \times R$ rank R case with $R = 4$, while the case $R = 8$ is displayed in the second image. In addition to plotting the aforementioned angles, the CREB and deterministic bound for existence of two distinct eigenvalue clusters are plotted. The CREB and deterministic bound are computed by taking the largest bound found using either five or fifty[14] different unitary matrices and the subpencil which gives the best CREB is the subpencil used to compute generalized eigenspaces in the GESD algorithm. The value of p used to compute the CREB is $p = 0.9$.

We find that instability in GESD is reasonably well predicted by the bounds. For SNR above the deterministic bound, the first step of GESD is highly accurate. Small errors start to appear when SNR decreases below the deterministic bound and significant errors start to occur within a few dB of the CREB. For all SNR values, the inaccuracy in the orthogonality of the $\tilde{\mathbf{E}}_i$ and the inaccuracy in the angle between the $\tilde{\mathbf{Z}}_i$ and the columns of $\mathbf{B}^{-\mathrm{T}}$ are observed to have similar magnitude.

7. Conclusion. We have presented an algebraic generalized eigenspace based algorithm for CPD computation which uses multiple subpencils of a given tensor to compute a decomposition. A key idea in the GESD algorithm is that one may combine information from many subpencils of a tensor to compute its CPD, thus allowing one to only perform trustworthy computations in each pencil. The performance of GESD has been compared to that of the popular GEVD algorithm. We find that in all situations, GESD is at least as accurate as GEVD, and that GESD is typically significantly more accurate than GEVD. We see that GESD particularly shines in difficult problems where the factor matrices of a tensor are highly correlated.

While the GESD algorithm typically takes about three times as long to run on a core tensor as GEVD, in practical settings which involve compression steps and optimization steps, a CPD computation using GESD for algebraic initialization of optimization is at least as fast as a CPD computation using GEVD for algebraic initialization. We observe that GESD based initializations are often significantly more accurate than GEVD based initializations, and can thereby lead to more reliable optimization. We also illustrate that no single fixed pencil selection strategy for GEVD can solve all given noiseless CPD problems even when the initial tensor meets the assumptions of the GEVD algorithm. Since GESD intelligently determines which computations can be reliably performed with a given subpencil, GESD does not suffer from this issue.

In addition to experimental analysis, we provide a theoretical analysis of the GESD algorithm. Here we compute deterministic and probabilistic bounds for the existence of a subpencil which has two distinct clusters of generalized eigenvalues which have not coalesced. These bounds may be interpreted as theoretical guarantees that a next step in the GESD algorithm is able to compute generalized eigenspaces which are meaningfully related to the generalized eigenspaces of the original noiseless tensor.

[14]The number of subpencils tried is in fact equal to $5\lfloor R/2 \rfloor$ or $50\lfloor R/2 \rfloor$, as for each unitary the CREB and deterministic bound are computed using $\lfloor R/2 \rfloor$ different subpencils.

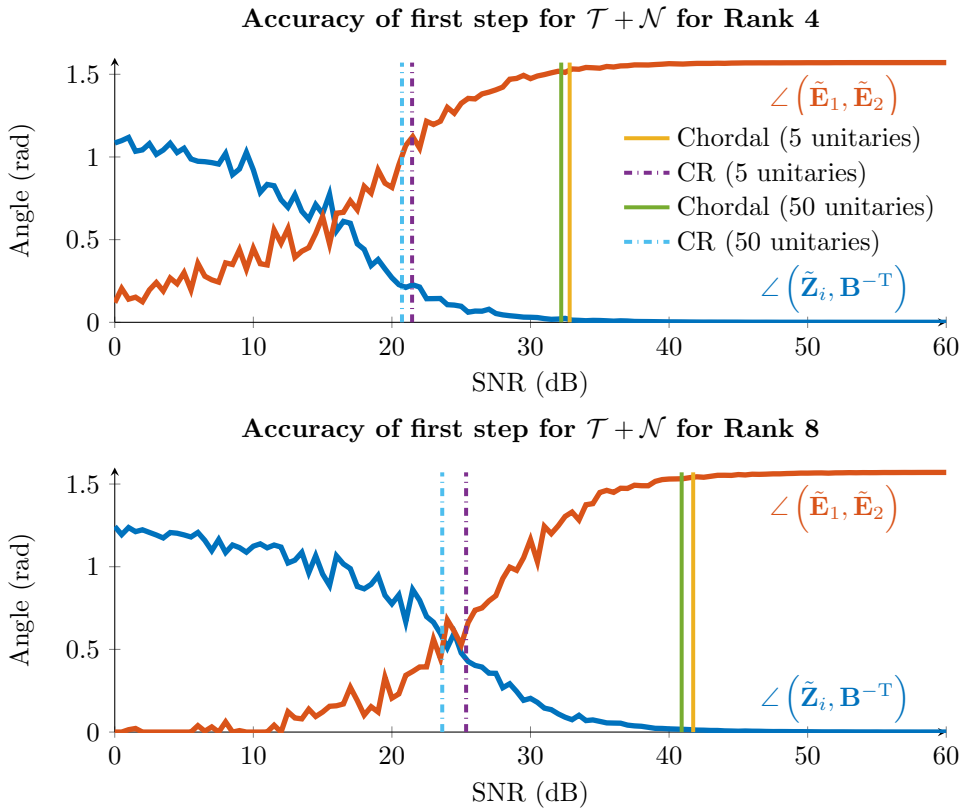


FIG. 7. *The angle between* $\tilde{\mathbf{E}}_1$ *and* $\tilde{\mathbf{E}}_2$ *stays close to* $\pi/2$ *until the chordal bound is crossed, while the angle between* $\tilde{\mathbf{Z}}_i$ *and columns of* $\mathbf{B}^{-T}$ *stays close to zero until the chordal bound is crossed. Trying more unitary matrices shifts the bounds sightly to the left. Mean (over* 50 *trials) angle between* $\tilde{\mathbf{E}}_1$ *and* $\tilde{\mathbf{E}}_2$ *and between* $\tilde{\mathbf{Z}}_i$ *and columns of* $\mathbf{B}^{-T}$ *for a rank-*4 *tensor of dimensions* $4\times4\times4$ *(top) and for a rank-*8 *tensor of dimensions* $8\times8\times8$ *(bottom), picking the best unitary matrix out of either* 5 *or* 50 *trials. The SNR is varied from* $120dB$ *to* $0dB$.

Currently we assume the existence of reasonably sized eigenvalue gaps which may be used to compute at least two distinct generalized eigenspaces. While this assumption is reasonable for many tensors of low rank, this assumption may fail to hold for particularly ill-conditioned tensors or when rank becomes large. Indeed, for generic tensors, the expected generalized eigenvalue gap becomes arbitrarily small as tensor rank and dimensions increase. Thus for tensors of sufficiently high rank, one should not expect to find a suitable gap. Resolving this is a direction for further research.

Acknowledgements. The authors thank Alexander Wagner, Ph.D for discussions which helped improve exposition in the article.

REFERENCES

[1] C. BELTRÁN, P. BREIDING, AND N. VANNIEUWENHOVEN, *Pencil-based algorithms for tensor rank decomposition are not stable*, SIAM J. Matrix Anal. Appl., 40 (2019).

[2] D. CAMPS, *Pole Swapping Methods for the Eigenvalue Problem—Rational QR Algorithms*, PhD thesis, KU Leuven, 2019.

[3] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and H. A. PHAN, *Tensor decompositions for signal processing applications: From two-way to multiway component analysis*, IEEE Signal Process. Mag., 32 (2015), pp. 145–163.
[4] P. Comon and C. Jutten, *Handbook of Blind Source Separation: Independent Component Analysis and Applications*, Academic Press, 2010.
[5] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278.
[6] V. de Silva and L. H. Lim, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM J. Matrix Anal. & Appl., 30 (2008), pp. 1084–1127.
[7] J. W. Demmel and B. Kågström, *Computing stable eigendecompositions of matrix pencils*, Linear Algebra and its Applications, 88-89 (1987), pp. 139 – 186.
[8] I. Domanov and L. De Lathauwer, *On the uniqueness of the canonical polyadic decomposition of third-order tensors—Part II: Uniqueness of the overall decomposition*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 876–903.
[9] I. Domanov and L. De Lathauwer, *Canonical polyadic decomposition of third-order tensors: Reduction to generalized eigenvalue decomposition*, SIAM J Matrix Anal Appl., 34 (2014), pp. 636–660.
[10] I. Domanov and L. De Lathauwer, *Canonical polyadic decomposition of third-order tensors: relaxed uniqueness conditions and algebraic algorithm*, Linear Algebra Appl, 513 (2017), pp. 342–375.
[11] E. Evert and L. De Lathauwer, *Guarantees for the existence of best low rank tensor approximations*, Tech. Report 19-71, ESAT-STADIUS, KU Leuven, Leuven, Belgium, 2019.
[12] E. Evert and L. De Lathauwer, *Perturbation theory for joint generalized eigenspaces*, Tech. Report 20-82, ESAT-STADIUS, KU Leuven, Leuven, Belgium, 2021.
[13] G. H. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, 4th ed., 2013.
[14] D. Kressner, *Numerical Methods for General and Structured Eigenvalue Problems*, Springer-Verlag Berlin Heidelberg, 2005.
[15] J. Kruskal, *Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics*, Linear Algebra Appl., 18 (1977), pp. 95–138.
[16] S. E. Leurgans, R. T. Ross, and R. B. Abel, *A decomposition for three-way arrays*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1064–1083.
[17] X. Liu and N. Sidiropoulos, *Cramér–Rao lower bounds for low-rank decomposition of multidimensional arrays*, IEEE Trans. Signal Process., 49 (2001), pp. 2074–2086.
[18] Y. Nakatsukasa, *Algorithms and Perturbation Theory for Matrix Eigenvalue Problems and the Singular Value Decomposition*, PhD thesis, University of California at Davis, 2011.
[19] E. Sanchez and B. R. Kowalski, *Tensorial resolution: A direct trilinear decomposition*, J. Chemom., 4 (1990), pp. 29–45.
[20] N. Sidiropoulos and R. Bro, *On the uniqueness of multilinear decomposition of n-way arrays*, Journal of Chemometrics, 14 (2000), pp. 229–239.
[21] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, *Tensor decomposition for signal processing and machine learning*, IEEE Trans. Signal Process., 65 (2017), pp. 3551–3582.
[22] A. Stegeman, *Degeneracy in Candecomp/Parafac explained for $p \times p \times 2$ arrays of rank $p+1$ or higher*, Psychometrika, 71 (2006), pp. 483–501.
[23] G. W. Stewart, *Error and perturbation bounds for subspaces associated with certain eigenvalue problems*, SIAM Review, 15 (1973), pp. 727–764.
[24] G. W. Stewart and J. Sun, *Matrix Perturbation Theory*, Academic Press, 1990.
[25] P. Tichavskỳ and Z. Koldovskỳ, *Weight adjusted tensor method for blind seperation of underdetermined mixtures of nonstationary sources*, IEEE Trans. on Signal Process., 59 (2011), pp. 1037–1047.
[26] P. Tichavskỳ, A.-H. Phan, and Z. Koldovskỳ, *Cramér–Rao-induced bounds for CANDECOMP/PARAFAC tensor decomposition*, IEEE Trans. Signal Process., 61 (2013), pp. 1986–1997.
[27] N. Vannieuwenhoven, *Condition numbers for the tensor rank decomposition*, Linear Algebra Appl., 535 (2017), pp. 35–86.
[28] N. Vervliet and L. De Lathauwer, *Numerical optimization based algorithms for data fusion*, in Data Fusion Methodology and Applications, M. Cocchi, ed., vol. 31, Elsevier, 2019, pp. 1–41.
[29] N. Vervliet, O. Debals, L. Sorber, M. Van Barel, and L. De Lathauwer, *Tensorlab 3.0*, Mar. 2016, https://www.tensorlab.net. Available online.