Canonical Polyadic Decomposition via the generalized Schur decomposition

Eric Evert*†, Michiel Vandecappelle*†, *Student member, IEEE,* and Lieven De Lathauwer*†, *Fellow, IEEE*

***Abstract*—The canonical polyadic decomposition (CPD) is a fundamental tensor decomposition which expresses a tensor as a sum of rank one tensors. In stark contrast to the matrix case, with light assumptions, the CPD of a low rank tensor is (essentially) unique. The essential uniqueness of CPD makes this decomposition a powerful tool in many applications as it allows for extraction of component information from a signal of interest.**

One popular algorithm for algebraic computation of a CPD is the generalized eigenvalue decomposition (GEVD) which selects a matrix subpencil of a tensor, then computes the generalized eigenvectors of the pencil. In this article, we present a simplification of GEVD which improves the accuracy of the algorithm. Surprisingly, the generalized eigenvector computation in GEVD is in fact unnecessary and can be replaced by a QZ decomposition which factors a pair of matrices as a product of unitary and upper triangular matrices. Computing a QZ decomposition is a standard first step when computing generalized eigenvectors, so our algorithm can been seen as a direct simplification of GEVD.

***Index Terms*—Tensors, CPD, QZ, GEVD, Multilinear Algebra**

I. Introduction

Tensors, or multiindexed numerical arrays, are higher order generalizations of matrices and are natural structures for expressing data and signals which have inherent higher order structure. In this article we study the canonical polyadic decomposition (CPD) which expresses a tensor as a sum of rank one compoents. The CPD plays an important role in many applications due to the fact that, with mild assumptions, a low rank tensor has a unique CPD [1]–[3]. As such, one can recover underlying component information by computing a CPD of a low rank signal tensor [4]–[6]. The essential uniqueness of CPD has helped make tensors and tensor methods common place in machine learning and signal processing [4], [5].

A standard approach for computing a CPD of a low rank tensor is to first algebraically approximate the decomposition, then to refine the approximation with optimization routines. These algebraic approximations play an important role in optimization routines, as they are relatively inexpensive to compute and the strong initializations they provide can both improve final accuracy and reduce total computation time. Notably, computing a best low rank approximation of a noisy low rank signal tensor is nonconvex and NP-hard [7], so good algebraic initializations greatly aid in getting reliable solutions.

A common approach for algebraic CPD computation is to first compute the generalized eigenvectors of a matrix subpencil of a tensor [8]–[11]. One of the tensor's factors can then be obtained by computing the inverse transpose of the matrix of generalized eigenvectors for the pencil. This generalized eigenvalue decomposition (GEVD) has been examined by many authors, e.g., see [10]–[12].

We present a simplification of the GEVD algorithm which is more accurate than the original algorithm. As it turns out, the generalized eigenvector computation in GEVD is not needed. Instead, one need only compute a generalized Schur decomposition of a subpencil of the tensor. This decomposition is also called the $\mathbf{QZ}$ decomposition. For a generic low rank tensor, the computed $\mathbf{Q}$ and $\mathbf{Z}$ will simultaneously upper triangularize all frontal slices of the tensor. Our key observation is that, once one has upper triangularized all frontal slices of the tensor, one can simply read off one of the tensors factor matrices from the diagonal entries of the upper triangular slices. Borrowing the perspective of [13], upper triangularizing the frontal slices of a low rank tensor reveals its "joint generalized eigenvalues."

Computing a $\mathbf{QZ}$ decomposition is a standard first step in computing generalized eigenvectors [14], so our algorithm can be seen as a direct simplification of GEVD. Intuitively, we view CPD computation as a (joint) generalized eigenvalue computation rather than a (joint) generalized eigenvector computation. The increase in accuracy of in our algorithm is due to the fact that computing the $\mathbf{QZ}$ decomposition only relies on unitary matrices as opposed to the general invertible matrices needed for generalized eigenvectors. Furthermore, the $\mathbf{QZ}$ method eliminates an inverse computation needed by GEVD.

It is worth noting that ours is not the first algorithm for CPD computation based on simultaneous upper triangularization of the frontal slices of a tensor. For example, in [15] frontal slices are jointly upper triangularized by minimizing the Frobenius norm of the lower triangular portion of the slices. However, the method used in [15] to obtain factor matrices after simultaneous upper triangularization is more involved.

Our $\mathbf{QZ}$ based CPD algorithm is presented in Section II. In Section III we prove that our algorithm successfully computes the CPD of a generic low rank tensor. The article ends with Section IV where we illustrate the performance of our algorithm on direction-of-arrival retrieval and in a fluorescence data experiment using the amino acid data set from [16].

Notation and terminology. Let $\mathbb{K}$ denote either $\mathbb{R}$ or $\mathbb{C}$. We

Funding: Research supported by: (1) Flemish Government: Fonds de la Recherche Scientifique–FNRS and the Fonds Wetenschappelijk Onderzoek–Vlaanderen under EOS project no 30468160 (SeLMA) and Artificiële intelligentie (AI) Vlaanderen (3E190661); (2) KU Leuven Internal Funds C16/15/059 and ID-N project no 3E190402; (3) Leuven Institute for Artificial Intelligence (Leuven.ai).

*KU Leuven, Dept. of Electrical Engineering ESAT/STADIUS, Kasteelpark Arenberg 10, bus 2446, B-3001 Leuven, Belgium

† KU Leuven – Kulak, Group Science, Engineering and Technology, E. Sabbelaan 53, B-8500 Kortrijk, Belgium

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

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. For a matrix $\mathbf{A} \in \mathbb{K}^{I\times J}$, 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 $\mathbf{A}^{\mathrm{T}}$. We use $D_r(\mathbf{A})$ denote the diagonal matrix whose diagonal entries are given by the rth row of $\mathbf{A}$. We say a matrix is **generic** if it lies in a full measure set. Roughly speaking, a matrix is generic with probability equal to one.

A **tensor** is a multiindexed array with entries in $\mathbb{K}$. The **order** of a tensor is the number of indices. Given a collection nonzero vectors $\mathbf{u}^{(1)} \in \mathbb{K}^{I_1}, \dots, \mathbf{u}^{(N)} \in \mathbb{K}^{I_N}$, let

$$\mathbf{u}^{(1)} \otimes \cdots \otimes \mathbf{u}^{(N)} \in \mathbb{K}^{I_1\times\cdots\times I_N}$$

denote the $I_1 \times \cdots \times I_N$ tensor with $i_1, i_2, \dots, i_N$ entry equal to $u^{(1)}_{i_1} u^{(2)}_{i_2} \cdots u^{(N)}_{i_N}$. A tensor of this form is called a **rank one tensor**. The minimal integer R such that

$$\mathcal{T} = \sum_{r=1}^{R} \mathbf{u}^{(1)}_r \otimes \cdots \otimes \mathbf{u}^{(N)}_r$$

where each $\mathbf{u}^{(n)}_r$ has entries in $\mathbb{K}$ is called the $\mathbb{K}$-rank of the tensor $\mathcal{T}$, and a decomposition of this form is called a **canonical polyadic decomposition** (CPD) of $\mathcal{T}$. Compactly we write $\mathcal{T} = [\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(N)}]\!]$. Here the matrix $\mathbf{U}^{(n)} \in \mathbb{K}^{I_n\times R}$ has $\mathbf{u}^{(n)}_r$ as its rth column. The matrix $\mathbf{U}^{(n)}$ is called a **factor matrix** of $\mathcal{T}$.

A **mode-ℓ fiber** of a tensor is a vector obtained by fixing all indices but the ℓth. Defined for a third order tensor, the **mode-ℓ unfolding** $\mathbf{T}_{[\ell;j,i]}$ of $\mathcal{T}$ is the matrix obtained by stacking all mode-ℓ fibers of $\mathcal{T}$ as columns of a matrix, where the mode i indices increment faster than the mode-j indices. The **ℓ-mode** product $\mathcal{T} \cdot_\ell \mathbf{A}$ between a matrix $\mathbf{A}$ and a tensor $\mathcal{T}$ is the tensor with mode-ℓ unfolding equal to $\mathbf{A}\mathbf{T}_{[\ell;j,i]}$. We also make use of more general unfoldings for tensors of order greater than three. E.g., if $\mathcal{T} \in \mathbb{K}^{I_1\times I_2\times I_3\times I_4}$, then $\mathbf{T}_{[4,3;2,1]}$ is an $I_1I_2 \times I_3I_4$ matrix with $I_1(i_2-1)+i_1, I_3(i_4-1)+i_3$ entry equal to t_{i_1,i_2,i_3,i_4}.

We often consider order three subtensors of a tensor. Given a tensor $\mathcal{T} \in \mathbb{K}^{I_1\times\cdots\times I_N}$ and an integer $3 \le n \le N$ define

$$\mathcal{T}[n] := \mathcal{T}(:,:,1,\dots,1,:,1,\dots,1)$$

where the third : occurs in the nth mode of $\mathcal{T}$. That is, $\mathcal{T}[n]$ is an order three subtensor of $\mathcal{T}$ formed by fixing all but the first, second, and nth. In the case that $\mathcal{T}$ has order 3, we call the matrices $\{\mathbf{t}(:,:,k)\}_{k=1}^{I_3}$ the **frontal slices** of $\mathcal{T}$.

II. CPD BY QZ

We now present the CPDQZ algorithm. We let $\mathcal{M} \in \mathbb{K}^{I_1\times\cdots\times I_N}$ denote a measured tensor of interest, and we assume that $\mathcal{M}$ has the form $\mathcal{M} = \mathcal{T} + \mathcal{N}$ where $\mathcal{T}$ is the signal portion of $\mathcal{M}$ and where $\mathcal{N}$ is noise. A standard assumption for generalized eigenvalue based algorithms such as GEVD and the generalized eigenspace decomposition (GESD), see [17], is that the tensor $\mathcal{T}$ has at least two factor matrices with full column rank. In particular, letting R denote the rank $\mathcal{T}$, one has $R \le \min\{I_1, I_2\}$ up to a permutation of indices. Thus, by computing a (truncated) orthogonal compression of $\mathcal{M}$ such as a multilinear singular value decomposition, see e.g. [18], we can restrict to the case where $\mathcal{M}$ has rank R and size $R \times R \times R_3 \times \cdots \times R_N$ with $R \ge R_n$ for all n.[1]

When compared to GEVD, our algorithm requires one additional mild assumption. Namely, we assume that the matrix $\mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)}$ has full column rank. Here $\odot$ denotes the Khatri-Rao product. In the upcoming CPDQZS variation of our algorithm, one instead must make the stronger assumption that there is an index $3 \le n \le N$ such that $R_n = R$ and $\mathbf{U}^{(n)}$ is invertible. These deterministic assumptions are satisfied for generic factor matrices provided the matrices in question all have at least as many rows as columns (before orthogonal compression).

The key observation behind CPDQZ is that one may compute all but two factor matrices of a tensor using a single $\mathbf{QZ}$ decomposition together with n-mode products. In particular, given a rank R tensor $\mathcal{T} \in \mathbb{K}^{R\times R\times R_3\times\cdots\times R_N}$ which satisfies our assumptions, to obtain factor matrices $\mathbf{U}^{(n)}$ for $n \ge 3$, one need only compute unitary matrices $\mathbf{Q}$ and $\mathbf{Z}$ such that

$$\mathbf{Q}\mathbf{T}[3](:,:,1)\mathbf{Z} \qquad \mathbf{Q}\mathbf{T}[3](:,:,2)\mathbf{Z}$$

is a $\mathbf{QZ}$ decomposition of the matrix pencil[2]

$$(\mathbf{T}[3](:,:,1), \mathbf{T}([3](:,:,2)),$$

i.e., such that these matrices are both upper triangular.[3]

Define $\mathcal{T}_{qz} := \mathcal{T} \cdot_1 \mathbf{Q} \cdot_2 \mathbf{Z}^{\mathrm{T}}$. It is then a matter of technical formula manipulation to show that

$$\mathbf{U}^{(n)}(r,:) = \mathbf{t}_{qz}[n](r,r,:)$$

for all $n = 3, \dots, N$ and all $r = 1, \dots, R$. See the supplementary materials for details.

It remains to compute $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$. To do this we first compute $\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)}$ by solving the overdetermined system

$$\mathbf{T}_{[N,N-1,\dots,3;2,1]} = (\mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)})(\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)})^{\mathrm{T}}.$$

The rth column of the matrix $\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)}$ is the Kronecker product of the rth column of $\mathbf{U}^{(2)}$ with that of $\mathbf{U}^{(1)}$ and can therefore be reshaped into a rank-1 matrix. It follows that the columns of these factors can be obtained by computing a rank-1 approximation of each reshaped column of $\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)}$.

As an alternative, suppose $\mathbf{U}^{(n)} \in \mathbb{K}^{R\times R}$ has full rank for some $n \in \{3, \dots, N\}$. For ease of exposition we take $n = N$. Then the remaining factor matrices can be computed by first solving for $(\mathbf{U}^{(N-1)} \odot \cdots \odot \mathbf{U}^{(1)})$ in

$$\mathbf{T}_{[N-1,\dots,1;N]} = (\mathbf{U}^{(N-1)} \odot \cdots \odot \mathbf{U}^{(1)})(\mathbf{U}^{(N)})^{\mathrm{T}},$$

[1] We assume $R \ge R_n$ since tensor rank upper bounds multilinear rank.

[2] Linear combinations of matrices $\mathbf{T}(:,:,i_3,\dots,i_N)$ may be taken to form the matrix pencil used for $\mathbf{QZ}$ computation. We use a fixed pencil for the sake of exposition. A popular pencil choice which can improve numerics is the first two slices of the (truncated) core of the MLSVD of $\mathcal{T}$. However, this does not solve all issues, e.g., see [17, Section 4.4].

[3] If a matrix pencil $(\mathbf{M}_1, \mathbf{M}_2)$ has complex generalized eigenvalues and $\mathbf{Q}$ and $\mathbf{Z}$ have real entries, then there will be 2×2 blocks on the diagonal of each matrix $\mathbf{Q}\mathbf{M}_i\mathbf{Z}$. However, if $\mathbf{M}_1$ and $\mathbf{M}_2$ are formed from linear combinations of frontal slices of a real rank R tensor $\mathcal{T}$ which meets our assumptions, then all generalized eigenvalues of $(\mathbf{M}_1, \mathbf{M}_2)$ are real, and each $\mathbf{Q}\mathbf{M}_i\mathbf{Z}$ will be upper triangular, e.g., see [11]. Thus, the presence of 2×2 blocks on the diagonal may indicate that the low real rank model chosen is not appropriate or that the signal to noise ratio is not sufficiently high.

then computing rank-1 approximations of the appropriately reshaped columns of this matrix. We call the first approach CPDQZ and the second CPDQZS(ingle). For emphasis, CPDQZ and CPDQZS are the same for order three tensors.

For tensors or order greater than three, the difference between these methods is that the CPDQZS method uses rank-1 tensor approximations to compute most of its factors, while the CPDQZ relies more heavily on the initial $\mathbf{QZ}$ decomposition to obtain factors. Since a best rank-1 tensor approximation can often be accurately computed and since there is less opportunity for error accumulation before rank-1 tensor approximations are employed in CPDQZS, the CPDQZS method is expected to be more accurate than CPDQZ. This expectation is supported by our numerical experiments.

The growth rate of the cost of both GEVD and CPDQZ is $\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$. The growth rate of the cost of CPDQZS is $\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$. If $R_N = R$, then these growth rates coincide. However, the coefficients of the costs for the methods can be very different. The algorithms all share a step whose cost grows at $\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$. CPDQZS and GEVD each have additional steps with this growth rate, while CPDQZ does not. Thus, the coefficient of the cost of CPDQZ is lower than that of GEVD or CPDQZS. In practice, CPDQZ is observed to be much faster than CPDQZS and GEVD, see Section IV. See the supplementary materials for further discussion.

III. Algorithm derivation

We now derive the CPDQZ algorithm. To ease exposition, we temporarily assume $\mathcal{T} \in \mathbb{K}^{R\times R\times R}$ has order three. The first observation used in the derivation of the CPDQZ algorithm is that in the special case where $\mathcal{T} = [\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}]\!]$ has upper triangular frontal slices and meets our assumptions, then in an appropriate column ordering, the matrix $\mathbf{U}^{(1)}$ is upper triangular while the matrix $\mathbf{U}^{(2)}$ is lower triangular.

This fact follows quickly from the main observation behind GEVD. Namely, if $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are $R \times R$ invertible matrices, then the columns of the matrix $(\mathbf{U}^{(2)})^{-\mathrm{T}}$ are equal to the generalized eigenvectors of the matrix pencil

$$(\mathbf{T}(:,:,1), \mathbf{T}(:,:,2)).$$

In the case that $\mathbf{T}(:,:,1)$ and $\mathbf{T}(:,:,2)$ are upper triangular, a routine argument shows that the pencil's generalized eigenvectors can be ordered so that $(\mathbf{U}^{(2)})^{-\mathrm{T}}$ is upper triangular, hence $\mathbf{U}^{(2)}$ is lower triangular. Having shown that $\mathbf{U}^{(2)}$ is lower triangular, one may use the assumption that $\mathcal{T}$ has upper triangular frontal slices together with the formula

$$\mathbf{T}(:,:,r) = \mathbf{U}^{(1)} D_r(\mathbf{U}^{(3)})(\mathbf{U}^{(2)})^{\mathrm{T}} \qquad \text{for } r = 1, \ldots, R \quad (1)$$

to conclude not only that $\mathbf{U}^{(1)}$ is upper triangular, but also that, up to scaling, one has

$$\mathbf{U}^{(3)}(:,r) = \mathrm{diag}(\mathbf{T}(:,:,r)) \qquad \text{for } r = 1, \ldots, R.$$

From this point we need only show that if $\mathcal{T} = [\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}]\!] \in \mathbb{K}^{R\times R\times R}$ is an arbitrary tensor which meets our assumptions and if $\mathbf{Q}$ and $\mathbf{Z}$ are matrices which give a $\mathbf{QZ}$ decomposition of the matrix pencil

$$(\mathbf{QT}(:,:,1)\mathbf{Z}, \mathbf{QT}(:,:,2)\mathbf{Z}),$$

then the tensor $\mathcal{T} \cdot_1 \mathbf{Q} \cdot_2 \mathbf{Z}^{\mathrm{T}} = [\![\mathbf{QU}^{(1)}, \mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}, \mathbf{U}^{(3)}]\!]$ has upper triangular frontal slices. As we shall explain, this fact follows from repeating preceding argument.

Our assumptions guarantee that the pencil $(\mathbf{T}(:,:,1), \mathbf{T}(:,:,2))$ has real generalized eigenvalues [11], hence $\mathbf{QT}(:,:,1)\mathbf{Z}$ and $\mathbf{QT}(:,:,2)\mathbf{Z}$ are upper triangular by definition of the $\mathbf{QZ}$ factorization and the matrix of generalized eigenvectors of this pencil, i.e, the inverse transpose of the second factor matrix of the tensor $\mathbf{T}\cdot_1\mathbf{Q}\cdot_2\mathbf{Z}^{\mathrm{T}}$, can be taken to be upper triangular, e.g., see [14]. That is, $\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}$ can be taken to be lower triangular. Applying equation (1) to $\mathbf{T} \cdot_1 \mathbf{Q} \cdot_2 \mathbf{Z}^{\mathrm{T}}$ shows that $\mathbf{QU}^{(1)}$ is upper triangular and that all frontal slices of this tensor are upper triangular. Since $\mathcal{T}$ and $\mathcal{T} \cdot_1 \mathbf{Q} \cdot_2 \mathbf{Z}^{\mathrm{T}}$ have the same third factor matrix it then easily follows that the CPDQZ algorithm successfully recovers the CPD of a tensor which meets the assumptions stated at the beginning of Section II.

The extension of CPDQZ and CPDQZS to tensors of order greater than three follows a routine argument using the fact that an order N tensor with CPD $[\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}]\!]$ can be reshaped to an order three tensor with CPD $[\![\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)}]\!]$.

IV. Experiments

The proposed QZ methods make different trade-offs with respect to accuracy and speed. In this section, we compare both methods with the classical GEVD algorithm and the more recent GESD algorithm [19]. We use a machine with an AMD Ryzen 5 5600H CPU at 3,30GHz and 16GB of RAM using MATLAB R2021b and Tensorlab 3.0 [20].

In a first experiment, we generate fourth-order low-rank tensors $\mathcal{T}$ by sampling the entries of factor matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ of dimensions $80 \times R$ from the uniform distribution on $[0, 1]$ and we normalize all columns to unit length. The rank R is varied in the range $[4, 16]$ and Gaussian noise is added such that the SNR is $40\,\mathrm{dB}$. For the estimated factor matrices $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, $\hat{\mathbf{C}}$ and $\hat{\mathbf{D}}$, we show maximal relative factor matrix errors compared to the true factors, defined as

$$\max\left(\frac{\|\mathbf{A}-\hat{\mathbf{A}}\|_{\mathrm{F}}}{\|\mathbf{A}\|_{\mathrm{F}}}, \frac{\|\mathbf{B}-\hat{\mathbf{B}}\|_{\mathrm{F}}}{\|\mathbf{B}\|_{\mathrm{F}}}, \frac{\|\mathbf{C}-\hat{\mathbf{C}}\|_{\mathrm{F}}}{\|\mathbf{C}\|_{\mathrm{F}}}, \frac{\|\mathbf{D}-\hat{\mathbf{D}}\|_{\mathrm{F}}}{\|\mathbf{D}\|_{\mathrm{F}}}\right),$$

where columns of $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, $\hat{\mathbf{C}}$ and $\hat{\mathbf{D}}$ and have been optimally permuted and scaled to match the columns of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$, respectively. The results are shown in the top plots of Figure 1. CPDQZ is the fastest method, but is less accurate compared to CPDQZS and GEVD. CPDQZS is as fast as GEVD, but is more accurate over the whole range of ranks. In the bottom plots of Figure 1, the experiment is repeated, but now the SNR is varied between 10 and $60\,\mathrm{dB}$, with the rank $R = 10$ fixed. The same relative performance is seen for the four methods.

In a more applied experiment, the QZ approach is compared to GEVD in a direction-of-arrival (DOA) retrieval experiment for line-of-sight signals impinging on a uniform rectangular array (URA). The CPD can be applied to find the DOAs in this case [21]–[23]. The URA has $M \times M$ sensors, where $M = 20$, and collects $K = 20$ samples from $R = 8$ far-field sources. These have azimuths [15 20 25 30 35 40 50 55] and elevations

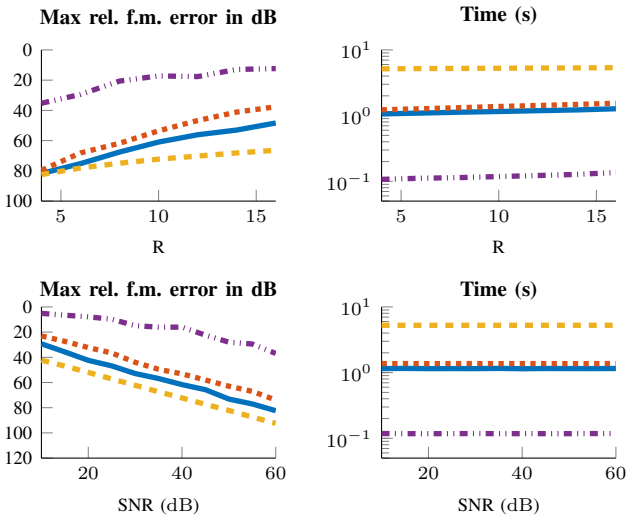


Fig. 1. The QZ methods make different trade-offs with respect to time and accuracy. Over 50 trials, CPDQZ ( ) is the fastest, but least accurate, while GESD ( ) is the most accurate and the slowest. CPDQZS ( ) and GEVD ( ) are equally fast, but the former is more accurate. The relative performance of the methods for the fourth-order tensors is consistent over varying ranks (with SNR 40 dB) and SNRs (with rank $R = 10$). In both cases, CPDQZS is on average about 8 dB more accurate than GEVD.

$[\,5\ 10\ 20\ 25\ 30\ 35\ 40\ 45\,]$ degrees, respectively. We consider omnidirectional sensors, evenly-spaced with inter-sensor spacing Δ. The azimuths $\mathbf{Z} \in \mathbb{C}^{M\times R}$ and elevations $\mathbf{L} \in \mathbb{C}^{M\times R}$ of the R sources that we collect at each sensor yield an observed tensor $\mathcal{T} \in \mathbb{C}^{M\times M\times K}$, where the kth frontal slice has a low rank decomposition $\mathbf{T}_k = \mathbf{A}^{(k)}\mathrm{diag}(\mathbf{s}^{(k)})(\mathbf{E}^{(k)})^{\mathrm{T}}$. The matrix $\mathbf{A}^{(k)} \in \mathbb{C}^{M\times R}$ has entries $a_{mr}^{(k)} = \exp((m-1)2\pi/\lambda \sin(z_{rk}\pi/180)\Delta i)$, while $\mathbf{E}^{(k)} \in \mathbb{C}^{M\times R}$ has entries $e_{mr}^{(k)} = \exp((m-1)2\pi/\lambda \sin(l_{rk}\pi/180)\Delta i)$. The vector $\mathbf{s}^{(k)} \in \mathbb{C}^{R}$ holds the sources and λ is the signal wavelength. By computing a rank-R CPD of a few frontal slices of $\mathcal{T}$, the azimuths and elevations of the sources can be recovered from the first two CPD factors. The tensor $\mathcal{T}$ is perturbed with Gaussian distributed noise in the range $[-20, 40]$dB and we compare the accuracy and speed of the CPDs obtained with GEVD and CPDQZS. In Figure 2, one can find the median error (left) and computation time (right) over 500 trials for the azimuth and elevation estimations of the eight sources. QZ is more accurate than GEVD, especially for the estimation of the source elevations, and is faster as well.

Lastly, we use the amino acid dataset from [16]. This dataset holds the measured emission and excitation spectra of five amino acid mixtures in a tensor $\mathcal{T}$ of size $5 \times 201 \times 61$. The first mode corresponds to the five mixtures and the second and third modes refer to the emission and excitation spectra, respectively. In the left plot of Figure 3, the theoretical emission spectra are shown for the three amino acids. Following [16], $\mathcal{T}$ admits a rank-3 CPD with each component corresponding to one amino acid. We add Gaussian noise with SNRs between -20 and $20\,$dB and use GEVD, GESD and CPDQZS to compute a CPD of the perturbed $\mathcal{T}$. We compare the accuracy of the estimated concentrations of the amino acids in each mixture, provided by the mode-1 factor matrix of the CPD.

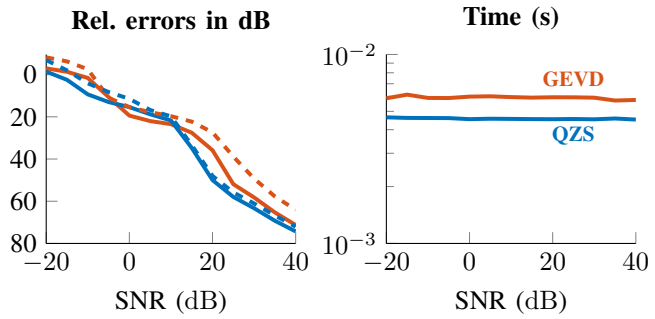


Fig. 2. CPDQZS estimates the elevations of the sources more accurately than GEVD outside of a small SNR window. In addition, CPDQZS has a lower computational cost. Left: mean relative errors over the eight sources of the azimuths ( ) and elevations ( ) that are estimated for CPDQZS and the azimuths ( ) and elevations ( ) that are estimated for GEVD. Right: computation time for both methods. Medians over 500 trials are shown.

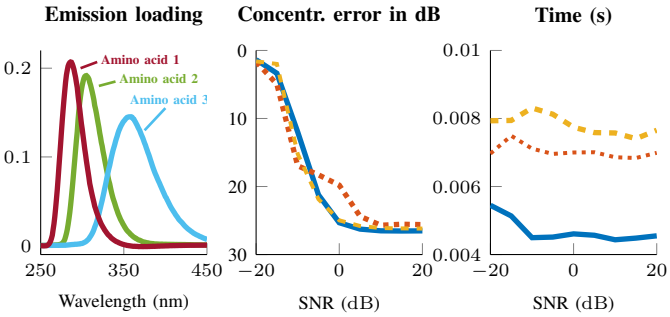


Fig. 3. The accuracy of CPDQZS ( ) for the estimation of the amino acid concentrations is as least as good as GEVD ( ) and GESD ( ), while requiring a lower computation time. The theoretical emission loading for the three amino acids in function of the wavelength is shown in the left plot. In the center plot, the mode-1 factor matrix error is shown, which corresponds to the estimated concentrations of the amino acids in the five mixtures. The right plot shows the computation times. All results are medians over 100 trials.

The relative mode-1 factor matrix errors are shown in the center plot of Figure 3, while the computation times are shown in the right plot. CPDQZS is about as accurate as GEVD and GESD over the range of SNRs and is also notably faster.

V. Conclusion

We presented the novel CPDQZ and CPDQZS algorithms for algebraic CPD computation which can be viewed as direct simplifications of the popular GEVD algorithm. These algorithms replace the generalized eigenvector computation of GEVD with a **QZ** decomposition which is used to upper triangularize the frontal slices of a tensor. One factor matrix is then obtained by reading the diagonal entries of the upper triangular slices. We showed in experiments that, in the case of CPDQZS, this simplification results in an increase in accuracy when compared to GEVD. For tensors of order four or more[4], CPDQZ is observed to be less accurate but faster than GEVD.

In future work we will investigate a deflation style algorithm in the spirit of GESD where the deflation step is based on a **QZ** computation rather than generalized eigenspace computations.

[4]Recall that CPDQZ and CPDQZS are identical for tensors of order three.

REFERENCES

[1] I. Domanov and L. De Lathauwer, "On the uniqueness of the canonical polyadic decomposition of third-order tensors — Part I: Basic results and uniqueness of one factor matrix," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 3, pp. 855–875, 2013.

[2] J. B. Kruskal, "Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics," *Linear Algebra Appl.*, vol. 18, pp. 95–138, 1977.

[3] N. D. Sidiropoulos and R. Bro, "On the uniqueness of multilinear decomposition of N-way arrays," *J. Chemometrics*, vol. 14, no. 3, pp. 229–239, 2000.

[4] N. 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.*, vol. 65, no. 13, pp. 3551–3582, 2017.

[5] A. Cichocki, D. P. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. F. Caiafa, and A.-H. Phan, "Tensor decompositions for signal processing applications: From two-way to multiway component analysis," *IEEE Signal Process. Mag.*, vol. 32, no. 2, pp. 145–163, 2015.

[6] P. Comon and C. Jutten, *Handbook of blind source separation: Independent component analysis and applications*. Academic press, 2009.

[7] C. J. Hillar and L.-H. Lim, "Most tensor problems are NP-hard," *J. ACM*, vol. 60, pp. 45:1–45:39, 2013.

[8] I. Domanov and L. De Lathauwer, "Canonical polyadic decomposition of third-order tensors: Reduction to generalized eigenvalue decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 2, pp. 636–660, 2014.

[9] ——, "Canonical polyadic decomposition of third-order tensors: Relaxed uniqueness conditions and algebraic algorithm," *Linear Algebra and its Applications*, vol. 513, pp. 342–375, 2017.

[10] E. Sanchez and B. Kowalski, "Tensorial resolution: A direct trilinear decomposition," *J. Chemometrics*, vol. 4, no. 1, pp. 29–45, 1990.

[11] S. E. Leurgans, R. T. Ross, and R. B. Abel, "A decomposition for three-way arrays," *SIAM J. Matrix Anal. Appl.*, vol. 14, no. 4, pp. 1064–1083, 1993.

[12] N. M. Faber, L. M. C. Buydens, and G. Kateman, "Generalized rank annihilation method. I: Derivation of eigenvalue problems," *J. Chemometrics*, vol. 8, no. 2, pp. 147–154, 1994.

[13] E. Evert and L. De Lathauwer, "Guarantees for existence of a best canonical polyadic approximation of a noisy low-rank tensor," *Technical Report 19–71, ESAT-STADIUS, KU Leuven, Leuven, Belgium*, 2021, (To appear in SIAM J. Matrix Anal. Appl.).

[14] G. Golub and C. V. Loan, *Matrix Computations*. The Johns Hopkins University Press, 1996.

[15] L. De Lathauwer, B. De Moor, and J. Vandewalle, "Computation of the canonical decomposition by means of a simultaneous generalized Schur decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 26, no. 2, pp. 295–327, 2004.

[16] R. Bro, "Multi-way analysis in the food industry-models, algorithms, and applications," in *MRI, EPG and EMA," Proc ICSLP 2000*. Citeseer, 1998.

[17] E. Evert, M. Vandecappelle, and L. De Lathauwer, "A recursive eigenspace computation for the canonical polyadic decompositions," ESAT-STADIUS, KU Leuven, Leuven, Belgium, Tech. Rep. 20-80, 2020, (To appear in SIAM J. Matrix Anal. Appl.).

[18] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2000.

[19] E. Evert, M. Vandecappelle, and L. De Lathauwer, "A recursive eigenspace computation for the canonical polyadic decomposition," *Technical Report 20-80, ESAT-STADIUS, KU Leuven, Leuven, Belgium*, 2020.

[20] N. Vervliet, O. Debals, L. Sorber, M. Van Barel, and L. De Lathauwer, "Tensorlab 3.0," 3 2016. [Online]. Available: http://www.tensorlab.net/

[21] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, "Parallel factor analysis in sensor array processing," *IEEE transactions on Signal Processing*, vol. 48, no. 8, pp. 2377–2388, 2000.

[22] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[23] M. Haardt, F. Roemer, and G. Del Galdo, "Higher-order SVD-based subspace estimation to improve the parameter estimation accuracy in multidimensional harmonic retrieval problems," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3198–3213, 7 2008.

Canonical Polyadic Decomposition via the generalized Schur decomposition Supp. Mats.

Eric Evert*†, Michiel Vandecappelle*†, *Student member, IEEE,* and Lieven De Lathauwer*†, *Fellow, IEEE*

These supplementary materials contain further details on the derivation of the CPDQZ algorithm and on the computational complexities of GEVD, CPDQZ, and CPDQZS.

I. Details on the derivation of CPDQZ algorithm.

We present two propositions which clarify the details of the derivation of the CPDQZ algorithm. As in the article, given a tensor $\mathcal{T} \in \mathbb{K}^{R\times R\times R\times R_3\times\cdots\times R_N}$ and an integer $3 \leq n \leq N$, define

$$\mathcal{T}[n] := \mathcal{T}(:,:,1,\ldots,1,:,1,\ldots,1)$$

where the third : occurs in the nth mode of $\mathcal{T}$. The first result shows that generically the first two factor matrices of a low rank tensor having upper triangular frontal slices can be taken to be triangular.

Proposition I.1. *Let $\mathcal{T} = \llbracket \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)} \rrbracket \in \mathbb{K}^{R\times R\times R_3\times\cdots\times R_N}$ be a rank R tensor where the matrices $\mathbf{U}^{(3)}, \ldots, \mathbf{U}^{(N)}$ are generic. Assume the matrix $\mathbf{T}[3](:,:,r)$ is upper triangular for each $r = 1,\ldots,R$, and assume $\mathcal{T}[3]$ has full multilinear rank. Then, up to column scaling, one may order the columns of the factor matrices so that*

$$\mathbf{U}^{(3)}(:,r) = \mathit{diag}(\mathbf{T}[3](:,:,r))$$

for each $r = 1,\ldots,R$. Furthermore, in this column ordering $\mathbf{U}^{(1)}$ is an upper triangular matrix, while $\mathbf{U}^{(2)}$ is a lower triangular matrix.

Proof. By assumption, $\mathbf{U}^{(r)}$ is generic for all $r = 4,\ldots,N$, so by appropriately scaling the columns of these factor matrices we have

$$\mathcal{T}[3] = \llbracket \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)} \rrbracket.$$

Therefore, we can without loss of generality restrict to the case where $\mathcal{T}$ is an order 3 tensor.

Now, by assumption $\mathcal{T} \in \mathbb{K}^{R\times R\times R}$ has rank R and full multilinear rank, thus we can use [1, Theorem 1.1] to conclude that the (spans of the) columns of $\mathbf{U}^{(3)}$ are equal to the joint generalized eigenvalues of $\mathcal{T}$, and the columns of $(\mathbf{U}^{(2)})^{-\mathrm{T}}$ are equal to the joint generalized eigenvectors of $\mathcal{T}$. Furthermore, using [1, Proposition 2.3] together with [1, Theorem 1.1] shows that the set joint of generalized eigenvalues of $\mathcal{T}$ is equal to

$$\{\mathrm{span}(\mathbf{t}(r,r,:))\}_{r=1}^{R}.$$

By assumption $\mathbf{U}^{(3)}$ is generic, from which it follows that the joint generalized eigenvectors of $\mathcal{T}$ are distinct. Using this together with the assumption that $\mathbf{T}(:,:,i)$ is upper triangular for each $i = 1,\ldots,R$, one may show that the joint generalized eigenvector corresponding to $\mathbf{t}(r,r,:)$ must lie in $\mathrm{span}(\{\mathbf{e}_1,\ldots,\mathbf{e}_r\})$ and that $\langle \mathbf{t}(r,r,:), \mathbf{e}_r\rangle \neq 0$. In particular, ordering the columns of $\mathbf{U}^{(3)}$ so that $\mathbf{U}^{(3)}(:,r) = \mathbf{t}(r,r,:)$, one has that $(\mathbf{U}^{(2)})^{-\mathrm{T}}$ is an upper triangular matrix, hence $\mathbf{U}^{(2)}$ is lower triangular. Having shown $\mathbf{U}^{(2)}$ is lower triangular, it is straightforward to show that $\mathbf{U}^{(1)}$ is upper triangular. □

Our second result shows that, to upper triangularize the frontal slices of the $\mathcal{T}[n]$, it is sufficient to upper triangularize the frontal slices of $\mathcal{T}[3]$.

Proposition I.2. *Let $\mathcal{T} = \llbracket \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)} \rrbracket \in (\mathbb{K}^R)^{\otimes N}$ be a generic tensor of rank R, and let $\mathbf{Q}$ and $\mathbf{Z}$ be unitary matrices such that*

$$\mathbf{Q}\mathbf{T}(:,:,1,1,\ldots,1)\mathbf{Z} \qquad \text{and} \qquad \mathbf{Q}\mathbf{T}(:,:,2,1,\ldots,1)\mathbf{Z}$$

is a $\mathbf{QZ}$ decomposition of the pencil $(\mathbf{T}(:,:,1,1,\ldots,1), \mathbf{T}(:,:,2,1,\ldots,1))$. Define $\mathcal{T}_{qz} := \mathcal{T} \cdot_1 \mathbf{Q} \cdot_2 \mathbf{Z}^{\mathrm{T}}$. Then $\mathcal{T}_{qz}[n]$ has upper triangular frontal slices for all $n = 3,\ldots,N$ and there exists a permutation matrix Π such that $\mathbf{Q}\mathbf{U}^{(1)}\Pi$ is an upper triangular matrix while $\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}\Pi$ is a lower triangular matrix.

Funding: Research supported by: (1) Flemish Government: Fonds de la Recherche Scientifique–FNRS and the Fonds Wetenschappelijk Onderzoek–Vlaanderen under EOS project no 30468160 (SeLMA) and Artificiële intelligentie (AI) Vlaanderen (3E190661); (2) KU Leuven Internal Funds C16/15/059 and ID-N project no 3E190402; (3) Leuven Institute for Artificial Intelligence (Leuven.ai).

*KU Leuven, Dept. of Electrical Engineering ESAT/STADIUS, Kasteelpark Arenberg 10, bus 2446, B-3001 Leuven, Belgium

† KU Leuven – Kulak, Group Science, Engineering and Technology, E. Sabbelaan 53, B-8500 Kortrijk, Belgium

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

In addition, there exist diagonal scaling matrices $\mathbf{D}_3, \dots, \mathbf{D}_N \in \mathbb{K}^{R\times R}$ *such that for all* $n = 3, \dots, N$ *one has*

$$(\mathbf{U}^{(n)}\Pi\mathbf{D}_n)(:, r) = \mathcal{T}_{qz}[n](r, r, :).$$

In words, in the column ordering which makes $\mathbf{QU}^{(1)}$ *and* $\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}$ *triangular, the columns of the* $\mathbf{U}^{(n)}$ *are given by the fibers on the diagonals of the* $\mathcal{T}_{qz}[n]$.

Proof. First observe that $\mathcal{T}_{qz}$ has CPD $\mathcal{T}_{qz} = [\![\mathbf{QU}^{(1)}, \mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \dots, \mathbf{U}^{(N)}]\!]$. Furthermore, since the $\mathbf{U}^{(n)}$ are generic, for each $n \geq 3$ there exists a diagonal matrix $\mathbf{D}_n \in \mathbb{K}^{R\times R}$ such that

$$\mathcal{T}_{qz}[n] = [\![\mathbf{QU}^{(1)}, \mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}, \mathbf{U}^{(n)}\mathbf{D}_n]\!].$$

In particular, the CPD of $\mathcal{T}_{qz}[n]$ is obtained by taking the first two factor matrices of $\mathcal{T}_{qz}$ together with a column scaling of $\mathbf{U}^{(n)}$. Using the identity

$$\mathcal{T}_{qz}[n](:, :, r) = \mathbf{QU}^{(1)}D_r(\mathbf{U}^{(n)}\mathbf{D}_n)(\mathbf{U}^{(2)})^{\mathrm{T}}\mathbf{Z},$$

to show $\mathcal{T}_{qz}[n]$ has upper triangular frontal slices, it is sufficient to show that $\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}$ is lower triangular and $\mathbf{QU}^{(1)}$ is upper triangular (perhaps allowing for a common permutation of the columns of these matrices).

Now, since $\mathcal{T}$ has generic factor matrices, the matrix pencil

$$(\mathbf{QT}(:, :, 1, 1, \dots, 1)\mathbf{Z}, \mathbf{QT}(:, :, 2, 1, \dots, 1)\mathbf{Z})$$

has R distinct generalized eigenvalues, and all the generalized eigenvalues of the pencil have geometric multiplicity equal to 1. Let $\{\mathbf{x}_1, \dots, \mathbf{x}_R\}$ be the set of generalized eigenvectors of the pencil. Then since $\mathbf{QT}(:, :, 1, 1, \dots, 1)\mathbf{Z}$ and $\mathbf{QT}(:, :, 2, 1, \dots, 1)\mathbf{Z}$ are both upper triangular, up to a permutation of indices the $\mathbf{x}_r$ have the property

$$\mathbf{x}_r \in \operatorname{span}(\{\mathbf{e}_1, \dots, \mathbf{e}_r\}) \qquad \text{and} \qquad \langle \mathbf{x}_r, \mathbf{e}_r\rangle \neq 0 \tag{1}$$

for each $r = 1, \dots, R$. That is, if one sets $\mathbf{X} := [\mathbf{x}_1 \dots \mathbf{x}_R] \in \mathbb{K}^{R\times R}$, then $\mathbf{X}$ is an invertible upper triangular matrix.

Furthermore since $(\mathbf{QT}(:, :, 1, 1, \dots, 1)\mathbf{Z}, \mathbf{QT}(:, :, 2, 1, \dots, 1)\mathbf{Z})$ has R distinct generalized eigenvalues, we may use [2, Proposition 1.1] to conclude that the joint generalized eigenvectors of $\mathcal{T}_{qz}[3]$ are equal to the generalized eigenvectors of $(\mathbf{QT}(:, :, 1, 1, \dots, 1)\mathbf{Z}, \mathbf{QT}(:, :, 2, 1, \dots, 1)\mathbf{Z})$. That is, the set of joint generalized eigenvectors of $\mathcal{T}_{qz}[3]$ is equal $\{\mathbf{x}_1, \dots, \mathbf{x}_R\}$.

Using [1, Theorem 1.1] shows that, up to permutation and scaling, $(\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)})^{-\mathrm{T}} = \mathbf{X}$. Therefore, up to a permutation of columns, the matrix $\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}$ is lower triangular. It is then straightforward to show that $\mathbf{QU}^{(1)}$ is upper triangular in the column ordering in which $\mathbf{Z}^{\mathrm{T}}\mathbf{U}^{(2)}$ is lower triangular.

The second part of the proof follows immediately from the first part of the proof together with the argument used to prove Proposition I.1. □

II. Computational Complexity

We now examine in further detail the computational complexities of GEVD, CPDQZ, and CPDQZS. In particular, tables I, II, and III give the asymtotic computational cost of each of these methods. We observe that the costs of both GEVD and CPDQZ grow at a rate of $\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$. Meanwhile, the cost of CPDQZS grows at a rate of $\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$. Recall, however, that as an additional assumption CPDQZS requires that one factor matrix $\mathbf{U}^{(n)}$ for $n \in \{3, \dots, N\}$ has full column rank. In our derivation we assumed that it is the matrix $\mathbf{U}^{(N)}$ which satisfies this assumption, hence $R_N = R$. Thus, for a tensor which meets the assumptions of the CPDQZS algorithm, the growth rate of the cost of each of these methods is the same.

While the growth rate of the computational costs for the methods all coincide under this assumption, the coefficients for the costs are notably different. All three methods share one (approximately) common step whose cost grows at a rate of $\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$. This is the only step of CPDQZ whose cost grows at this rate. Meanwhile CPDQZS has a second step whose cost grows at this rate, and GEVD has two additional steps whose costs grow at this rate. As such, the coefficient for the computational cost of CPDQZ is lower than that for CPDQZS or GEVD. On synthetic data we observe that CPDQZ is significantly faster than CPDQZS or GEVD and that CPDQZS is slightly faster than GEVD, see Section IV of the article. We also note that coefficient of the cost for the $\mathbf{QZ}$ step of CPDQZS is less than that for the generalized eigenvalue decomposition of GEVD [3].

References

[1] E. Evert and L. De Lathauwer, "Guarantees for existence of a best canonical polyadic approximation of a noisy low-rank tensor," *Technical Report 19–71, ESAT-STADIUS, KU Leuven, Leuven, Belgium*, 2021, (To appear in SIAM J. Matrix Anal. Appl.).

[2] E. Evert, M. Vandecappelle, and L. De Lathauwer, "Canonical polyadic decomposition via QZ," ESAT-STADIUS, KU Leuven, Leuven, Belgium, Tech. Rep. 21-124, 2021.

[3] G. Golub and C. V. Loan, *Matrix Computations*. The Johns Hopkins University Press, 1996.

Complexity for GEVD	
Compute matrix $\mathbf{V}$ of generalized eigenvectors of a pair of $R \times R$ slices of $\mathcal{T}[N]$	$\mathcal{O}(R^3)$
Obtain $\mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)} \odot \mathbf{U}^{(1)}$ through the product $\mathbf{T}_{[N,\ldots,3,1;2]}\mathbf{V}^{\mathrm{T}} = \mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)} \odot \mathbf{U}^{(1)}$	$\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$
Compute matrices $\mathbf{U}^{(n)}$ for $n \neq 2$ by computing rank one approximations of columns of $\mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)} \odot \mathbf{U}^{(1)}$	$\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$
Solve for $\mathbf{U}^{(2)}$ in $\mathbf{T}_{[N,\ldots,3,1;2]} = (\mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)} \odot \mathbf{U}^{(1)})(\mathbf{U}^{(2)})^{\mathrm{T}}$	$\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$
Total	$\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$

TABLE I
COMPUTATIONAL COMPLEXITY OF EACH STEP OF THE GEVD ALGORITHM

Complexity for CPDQZ	
Compute $\mathbf{QZ}$ decomposition of a pair of $R \times R$ slices of $\mathcal{T}[n]$ to obtain unitaries $\mathbf{Q}_n$ and $\mathbf{Z}_n$ for $n = 3, \ldots, N$	$\mathcal{O}(NR^3)$
Modal product of $\mathcal{T}[n]$ with $\mathbf{Q}_n$ and $\mathbf{Z}_n$ to obtain $\mathbf{U}^{(n)}$ for $n = 3, \ldots, N$	$\mathcal{O}(NR^4)$
Solve for $(\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)})$ in $\mathbf{T}_{[N,N-1,\ldots,3;2,1]} = (\mathbf{U}^{(N)} \odot \cdots \odot \mathbf{U}^{(3)})(\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)})^{\mathrm{T}}$	$\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$
Compute $\mathbf{U}^{(2)}$ and $\mathbf{U}^{(1)}$ by computing rank one approximations of columns of $(\mathbf{U}^{(2)} \odot \mathbf{U}^{(1)})$	$\mathcal{O}(R^4)$
Total	$\mathcal{O}(R^3(\Pi_{i=3}^{N} R_i))$

TABLE II
COMPUTATIONAL COMPLEXITY OF EACH STEP OF THE CPDQZ ALGORITHM

Complexity for CPDQZS	
Compute $\mathbf{QZ}$ decomposition of a pair of $R \times R$ slices of $\mathcal{T}[N]$ to obtain unitaries $\mathbf{Q}$ and $\mathbf{Z}$	$\mathcal{O}(R^3)$
Modal product of $\mathcal{T}[N]$ with $\mathbf{Q}$ and $\mathbf{Z}$ to obtain $\mathbf{U}^{(N)}$	$\mathcal{O}(R^4)$
Solve for $\mathbf{U}^{(N-1)} \odot \cdots \odot \mathbf{U}^{(1)}$ in $\mathbf{T}_{[N-1,\ldots,1;N]} = (\mathbf{U}^{(N-1)} \odot \cdots \odot \mathbf{U}^{(1)})(\mathbf{U}^{(N)})^{\mathrm{T}}$	$\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$
Compute remaining matrices $\mathbf{U}^{(n)}$ by computing rank one approximations of columns of $\mathbf{U}^{(N-1)} \odot \cdots \odot \mathbf{U}^{(1)}$	$\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$
Total	$\mathcal{O}(R^4(\Pi_{i=3}^{N-1} R_i))$

TABLE III
COMPUTATIONAL COMPLEXITY OF EACH STEP OF THE CPDQZS ALGORITHM