

A finite algorithm to compute rank-1 tensor approximations

Alex Pereira da Silva, Pierre Comon, André L. F. de Almeida

▶ **To cite this version:**

Alex Pereira da Silva, Pierre Comon, André L. F. de Almeida. A finite algorithm to compute rank-1 tensor approximations. IEEE Signal Processing Letters, 2016, 23 (7), pp.959-963. 10.1109/LSP.2016.2570862 . hal-01385538

HAL Id: hal-01385538
https://hal.science/hal-01385538

Submitted on 21 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Finite Algorithm to Compute Rank-1 Tensor Approximations

Alex P. da Silva, Pierre Comon, *Fellow, IEEE*, André L. F. de Almeida, *Senior Member, IEEE*

***Abstract*—We propose a non iterative algorithm, called SeROAP[a], to estimate a rank-1 approximation of a tensor in the real or complex field. Our algorithm is based on a sequence of singular value decompositions followed by a sequence of projections onto Kronecker vectors. For three-way tensors, we show that our algorithm is always at least as good as the state-of-the-art truncation algorithm, ST-HOSVD[b], in terms of approximate error. Thus, it gives a good starting point to iterative rank-1 tensor approximation algorithms. By means of computational experiments, it also turns out that for 4-th order tensors, SeROAP yields a better approximation with high probability when compared to the standard THOSVD[c] algorithm.**

***Index Terms*—rank-1 approximation; tensor; singular value decomposition.**

I. Introduction

In the last years, tensors have been playing an important role in many applications such as blind source separation [1], [2], [3], digital communications [4], [5], chemometrics [6], [7], neuroscience [8], sensor array processing [9] and data mining [10]. The attractiveness behind tensors lies in the uniqueness of their canonical polyadic (CP) decomposition under mild conditions [11], which is a powerful property not shared by standard matrix-based tools. There are several methods to compute the CP tensor decomposition. We will point out here some of the most used methods among many others. For the exact CP decomposition, [12] proposes a direct computation to decompose $2 \times n \times n$ tensors. In [13], a generalization of Sylvester's algorithm is described for decomposing symmetric tensors. In [14], one can use simultaneous matrix diagonalization by congruence, provided that the rank of the tensor is smaller than its greatest dimension. An approach based on eigenvectors of tensors is proposed in [15].

In practice, tensors are corrupted by noise so that we need to compute an approximate decomposition of low rank. Computing the exact CP decomposition is difficult [16], but finding a lower-rank approximation is even harder. In fact, this is an ill-posed problem in general [17]. Nevertheless, some useful algorithms have been conceived to solve locally the low-rank approximation problem. This kind of algorithms can be found in [18], [19], [20], [21], [22]. One of the most widely used is the alternating least squares (ALS) algorithm [6], which is an iterative method that consists in conditionally updating in an alternate way, the matrix factors composing the CP decomposition. Other gradient and Newton-based methods estimate the factor matrices all-at-once [18], [23].

A particular low rank approximation problem is the best rank-1 approximation. Let $\boldsymbol{\mathcal{T}} \in \mathbb{K}^{I_1 \times I_2 \times \ldots \times I_N}$ be a tensor with $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$. For vectors $\boldsymbol{a_i} \in \mathbb{K}^{I_i \times 1}$, $1 \leq i \leq N$, the best rank-1 tensor approximation problem can be stated as

$$p^{\min} = \min_{\boldsymbol{a}_i, 1 \leq i \leq N} \|\boldsymbol{\mathcal{T}} - \boldsymbol{a}_1 \otimes \boldsymbol{a}_2 \otimes \ldots \otimes \boldsymbol{a}_N\| \ , \qquad (1)$$

where $\|\cdot\|$ is the Frobenius norm and $\otimes$ is the outer product [17], [24], [25].

Contrary to best rank-r tensor approximations with $r > 1$, the best rank-1 tensor approximation of any tensor always exists in $\mathbb{R}$ or $\mathbb{C}$ since the set of rank-1 tensors is in a cone of Segre varieties [26], [27], which ensures Problem (1) is well-posed. The complexity class of the best rank-1 approximation problem is NP-hard.

An interest on rank-1 approximations lies, for instance, in deflation of symmetric and orthogonally decomposable tensors [28]. An application of best rank-1 approximation on multi-target tracking can be found in [29]. Recently, the authors in [30] apply Problem (1) to a class of low rank tensor optimization problems.

To compute the global minimum of Problem (1), we could resort to algebraic geometric tools such as those described in [31], [32]. However, these techniques are only efficiently applied to small-sized real tensors since they introduce a lot of variables due to relaxations, so that computational complexity and storage requirements become an issue.

Still in the algebraic geometry context, [33] describes an algorithm that computes best rank-1 approximations in a reasonable time, but the global minimum is attained only if some restrictive rank conditions are satisfied. Again, the drawback is the slow convergence for tensors with moderate dimensions.

Standard iterative algorithms such those described in [18] can be employed to tackle Problem (1). These algorithms are initialization-dependent, however, local convergence is ensured for the standard alternating least squares (ALS) algorithm [34].

In order to obtain a starting point for iterative algorithms, rank-1 tensor approximations close to a minimizer of Problem (1) can be delivered by finite algorithms, i.e., algorithms terminating within a finite number of steps. Among existing finite algorithms, we highlight here two algorithms based on

Alex P. da Silva and Pierre Comon are with GIPSA-Lab, 11 rue des Mathematiques, F-38402 Saint Martin d'Hères, cedex, France. André L. F. de Almeida is with the Department of Teleinformatics Engineering, Federal University of Ceará, CP 6005, Campus do Pici, 60455-760, Fortaleza, Ceará, Brazil. E-mails: {alex.pereira-da-silva@gipsa-lab.fr,pierre.comon}@gipsa-lab.fr, andre@gtel.ufc.br. This work has been funded by the European Research Council under the 7th Framework Program FP7/2007-2013 Grant Agreement no. 320594, and by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) under the program Ciências sem Fronteiras.

[a]Sequential Rank-One Approximation and Projection.

[b]Sequentially Truncated Higher-Order Singular Value Decomposition.

[c]Truncated Higher-Order Singular Value Decomposition.

the singular value decomposition (SVD). In [35], the authors extend the SVD concept to tensors, and a rank-1 approximation can be obtained by truncating the proposed higher-order singular value decomposition. It is called THOSVD throughout this paper. In [36], a new truncated strategy to higher order SVD, called ST-HOSVD, presents a computational complexity smaller than that of THOSVD. Moreover, in terms of approximate error, the authors show that ST-HOSVD is always at least as good as THOSVD for three-way real tensors when at least one of the unfolding matrices of the approximating tensor has a rank equal to 1. Therefore, ST-HOSVD has been an efficient alternative to compute a rank-1 three-way tensor approximation [36]. Other finite rank-1 algorithms are also described in [24].

We propose a finite algorithm, called SeROAP, that computes a rank-1 tensor approximation by means of a sequence of singular value decompositions of decreasing order tensors followed by a sequence of projections onto Kronecker vectors of increasing size. The goal of the proposed algorithm is to provide an approximate error at most equal to that obtained with ST-HOSVD (and consequently at most equal to that of THOSVD) for three-way tensors, with computational complexity smaller than that of THOSVD, and still competitive with that of ST-HOSVD, at least for tensors of small orders, regardless of their dimensions. Eventually, we report some experiments showing that, with high probability, SeROAP performs better than the THOSVD algorithm for 4-th order tensors.

II. Algorithm Description

Let π_i, $1 \leq i \leq N$, be the unfolding operator applied to a tensor $\boldsymbol{\mathcal{T}} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$ along the i-th mode. The unfolding operator is the procedure of reshaping the entries of a tensor into a matrix form [19]. Herein, $\boldsymbol{T}_{(i)} = \pi_i(\boldsymbol{\mathcal{T}})$ is a matrix representation of $\boldsymbol{\mathcal{T}}$ in the matrix space $\mathbb{K}^{I_i \times I_1 I_2 \cdots I_{i-1} I_{i+1} \cdots I_N}$. Let also π_i^{-1}, $1 \leq i \leq N$, be its inverse operator applied to a matrix form that recovers the original tensor $\boldsymbol{\mathcal{T}}$, that is $\boldsymbol{\mathcal{T}} = \pi_i^{-1}(\boldsymbol{T}_{(i)})$.

Before describing the SeROAP algorithm, we provide in the following a brief description of the rank-1 version of THOSVD and ST-HOSVD algorithms.

A. Truncated HOSVD

A rank-1 tensor approximation is obtained with THOSVD by computing the left singular vectors of every of its matrix unfoldings [35]. Let $\boldsymbol{T}_{(i)} = \boldsymbol{U}_i \Sigma_i \boldsymbol{V}_i^{\mathsf{H}}$ be the SVD of $\boldsymbol{T}_{(i)}$. The principal left singular vector $\boldsymbol{u}_i$ of $\boldsymbol{U}_i$ is the i-th factor of the desired rank-1 approximation. Let λ be the contraction of $\boldsymbol{\mathcal{T}}$ onto the unit rank-1 tensor $\boldsymbol{u}_1 \otimes \boldsymbol{u}_2 \otimes \cdots \otimes \boldsymbol{u}_N$, that is, $\lambda = \langle \boldsymbol{\mathcal{T}}, \otimes_{i=1}^{N} \boldsymbol{u}_i \rangle$, where $\langle \cdot \rangle$ is the Euclidian scalar product. Thus, $\boldsymbol{\mathcal{X}} = \lambda \boldsymbol{u}_1 \otimes \boldsymbol{u}_2 \otimes \cdots \otimes \boldsymbol{u}_N$, is an estimated rank-1 approximation of $\boldsymbol{\mathcal{T}}$. By using Lanczos algorithm to compute the principal singular vector at k steps [37], THOSVD requires $\mathcal{O}\{2Nk \prod_{i=1}^{N} I_i\}$ operations to obtain all factors $\boldsymbol{u}_i$, where k is a user-defined parameter.

B. Sequentially truncated HOSVD

ST-HOSVD is an alternative strategy that can be used to compute a rank-1 approximation of a tensor [36]. The idea behind this algorithm in this case is to construct the rank-1 tensor approximation with the principal left singular vectors $\boldsymbol{u}_i$, $1 \leq i \leq N$, as in THOSVD, but computed from a sequence of tensors of smaller and smaller order, which in turn are constructed from the principal right singular vectors. Herein, we omit the effect of singular values, present in the description of ST-HOSVD in [36], because they do not affect the computation of the factors for rank-1 approximations.The approximate error $\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}\|$, for an estimated rank-1 tensor $\boldsymbol{\mathcal{X}}$, depends on the order in which the modes are processed.

In terms of complexity, ST-HOSVD requires $\mathcal{O}\{2k \sum_{j=1}^{N-1} \prod_{i=j}^{N} I_i\}$ flops to compute all factors, hence less than THOSVD.

C. Sequential rank one approximation and projection

The proposed algorithm, called SeROAP, is a competitive finite algorithm that computes a rank-1 tensor approximation. As ST-HOSVD, SeROAP also constructs a sequence of tensors of smaller and smaller order, thereby the approximate error depends on the ordering of modes in the process.

On the other hand, contrary to ST-HOSVD, SeROAP does not compute the factors at every iteration. Instead, the rank-1 approximation is directly computed after a projection process. The order-N version of SeROAP algorithm goes along the lines depicted in Algorithm 1 (see also [38] for a longer description). We assume $\boldsymbol{p} = [p_1 \ p_2 \ \ldots \ p_N]$, with $p_i \in \{1, 2, \ldots, N\}$ and $p_i \neq p_j$ for $i \neq j$, a vector that defines the order in which the modes are processed, and $\boxtimes$ is the Kronecker product. We also define $\mathrm{vec}(\cdot)$ as the vectorization operator that stacks the columns of a matrix into a long column vector, and the superscript symbols $*$, T and H that represent the conjugate, the transpose and the conjugate transpose operators, respectively.

input : $\boldsymbol{\mathcal{T}} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$: input data, $\boldsymbol{p}$: processing order
output: $\boldsymbol{\mathcal{X}} \in \mathbb{K}^{I_1 \times I_2 \times \cdots \times I_N}$: rank-1 approximation
$\boldsymbol{\mathcal{S}} \leftarrow \boldsymbol{\mathcal{T}}$
for $n \leftarrow p_1, p_2, \ldots, p_{N-1}$ **do**
  $\boldsymbol{S}_{(n)} \leftarrow \pi_n(\boldsymbol{\mathcal{S}})$
  $\boldsymbol{v}_n \leftarrow$ principal right singular vector of $\boldsymbol{S}_{(n)}$
  $\boldsymbol{\mathcal{S}} \leftarrow \pi_n^{-1}(\boldsymbol{v}_n^{\mathsf{T}})$
end
$\boldsymbol{u} \leftarrow$ principal left singular vector of $\boldsymbol{S}_{(p_{N-1})}$
$\boldsymbol{w} \leftarrow \boldsymbol{v}_{p_{N-1}}^{*} \boxtimes \boldsymbol{u}$
for $n \leftarrow p_{N-2}, p_{N-3}, \ldots, p_1$ **do**
  $\boldsymbol{X}_{(n)} \leftarrow \boldsymbol{S}_{(n)} \boldsymbol{w} \boldsymbol{w}^{\mathsf{H}}$
  $\boldsymbol{w} \leftarrow \mathrm{vec}(\boldsymbol{X}_{(n)})$
end
$\boldsymbol{\mathcal{X}} \leftarrow \pi_{p_1}^{-1}(\boldsymbol{X}_{(p_1)})$

Algorithm 1: SeROAP algorithm

The SeROAP algorithm can be described in two different phases: decreasing tensor order and projection. The decreasing

order phase (first *for* loop) is similar to ST-HOSVD algorithm, except that in SeROAP the factors of the rank-1 approximation are still not computed. Actually, the goal here is to store the unfoldings $\boldsymbol{S}_{(p_1)}, \boldsymbol{S}_{(p_2)}, \ldots, \boldsymbol{S}_{(p_{N-2})}$ to project their rows into Kronecker vectors that will be obtained in the projection phase. After the first loop, a Kronecker vector $\boldsymbol{w}$ is obtained from the last unfolding $\boldsymbol{S}_{(p_{N-1})}$ by computing its rank-1 approximation. For SeROAP, we do not need to take use of the last element p_N of the mode ordering.

In the projection phase (second *for* loop), at the first step, we project the rows of the matrix $\boldsymbol{S}_{(p_{N-2})}$ onto the Kronecker vector $\boldsymbol{w}$, by performing $\boldsymbol{X}_{(p_{N-2})} = \boldsymbol{S}_{(n)}\boldsymbol{w}\boldsymbol{w}^{\mathsf{H}}$. Notice that $\boldsymbol{X}_{(p_{N-2})}$ can be viewed as the unfolding of a rank-1 three-way tensor. Indeed,

$$\begin{aligned}\boldsymbol{X}_{(p_{N-2})} =& \boldsymbol{S}_{(p_{N-2})}\boldsymbol{w}\boldsymbol{w}^{\mathsf{H}} \iff \\ & \boldsymbol{\mathcal{X}}_{(p_{N-2})} = \boldsymbol{S}_{(p_{N-2})}\boldsymbol{w} \otimes \boldsymbol{u}^* \otimes \boldsymbol{v}_{p_{N-1}}.\end{aligned}$$

The vector $\boldsymbol{w}$ is updated using the vec operator applied to $\boldsymbol{X}_{(p_{N-2})}$. Since $\boldsymbol{X}_{(p_{N-2})}$ is the unfolding of a three-way tensor, $\mathrm{vec}(\boldsymbol{X}_{(p_{N-2})})$ is a Kronecker product of three vectors: $\mathrm{vec}(\boldsymbol{X}_{(p_{N-2})}) = \boldsymbol{v}_{p_{N-1}} \boxtimes \boldsymbol{u}^* \boxtimes \boldsymbol{S}_{(p_{N-2})}\boldsymbol{w}$. Thus, at the next iteration, the unfolding $\boldsymbol{S}_{(p_{N-3})}$ is projected now onto a *three-way* Kronecker vector. This projection generates a matrix $\boldsymbol{X}_{(p_{N-3})}$ that is the unfolding of a 4-th order tensor. By continuing the process, we note that at the end of the projection phase, the vector $\boldsymbol{w}$ is a Kronecker product of $N-1$ vectors, so that the matrix $\boldsymbol{X}_{(p_1)}$ is actually the unfolding of an order-N rank-1 tensor.

The complexity of SeROAP is $\mathcal{O}\{2k\sum_{j=1}^{N-1}\prod_{i=j}^{N} I_i\}$ for the decreasing order phase, and $\mathcal{O}\{2\sum_{j=1}^{N-1}\prod_{i=j}^{N} I_i\}$ for the projection phase. Moreover, all matrices $\boldsymbol{S}_{(p_1)}, \boldsymbol{S}_{(p_2)}, \ldots, \boldsymbol{S}_{(p_{N-2})}$ must be stored before the second phase, which gives a total of $\sum_{j=1}^{N-2}\prod_{i=j}^{N} I_i$ push operations. Notice that if N is not large, the complexity of SeROAP is smaller than that of THOSVD, and not much larger than that of ST-HOSVD. For instance, for $k = 4$, $N = 3$ and $I_1 = I_2 = I_3 = 100$, we have approximately $24, 8.08$, and 10.1 million flops for THOSVD, ST-HOSVD and SeROAP algorithms, respectively. For SeROAP, we need 10^6 floating points in memory to store the unfolding matrix $\boldsymbol{S}_{(p_1)}$.

III. Theoretical Analysis

For three-way tensors, in SeROAP we only need to compute the principal singular triplet of two matrices: one to reduce the order of the tensor, and the other to construct the Kronecker vector. Yet, only a single projection is performed to obtain the rank-1 approximation. In this case, we present in the following a theoretical result showing that the rank-1 approximation computed with SeROAP is *always* at least as good as that delivered by ST-HOSVD, for the same ordering of modes.

Theorem 1. *Let $\boldsymbol{\mathcal{T}} \in \mathbb{K}^{I_1 \times I_2 \times I_3}$ be a three-way tensor. $\mathbb{K} = \mathbb{C}$ or $\mathbb{R}$. Let also $\boldsymbol{\mathcal{X}}^{ST}$ and $\boldsymbol{\mathcal{X}}^{Se}$ be the rank-1 approximations delivered by ST-HOSVD and SeROAP algorithms, respectively. Then the approximate error*

$$\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{Se}\| \leq \|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{ST}\|$$

holds for any mode ordering defined by $\boldsymbol{p}$.

Proof. Let $\boldsymbol{p} = [p_1\ p_2\ p_3]$, with $p_i \in \{1, 2, 3\}$ and $p_i \neq p_j$ for $i \neq j$. Let also $\boldsymbol{T}_{(p_1)}$, $\boldsymbol{X}^{\mathrm{ST}}_{(p_1)}$ and $\boldsymbol{X}^{\mathrm{Se}}_{(p_1)}$ be the unfolding matrix along mode p_1 of tensors $\boldsymbol{\mathcal{T}}$, $\boldsymbol{\mathcal{X}}^{\mathrm{ST}}$ and $\boldsymbol{\mathcal{X}}^{\mathrm{Se}}$, respectively. Denoting by $\boldsymbol{u}_{p_1}, \boldsymbol{u}_{p_2}$ and $\boldsymbol{u}_{p_3}$ the factors obtained with ST-HOSVD, the approximate error can be written as

$$\|\boldsymbol{T}_{(p_1)} - \boldsymbol{X}^{\mathrm{ST}}_{(p_1)}\|^2 = \|\boldsymbol{T}_{(p_1)} - \lambda \boldsymbol{u}_{p_1}(\boldsymbol{u}_{p_3} \boxtimes \boldsymbol{u}_{p_2})^{\mathsf{T}}\|^2,$$

where λ is a positive scalar number for unit factors $\boldsymbol{u}_n, n \in \{p_1, p_2, p_3\}$. Hence,

$$\begin{aligned}\|\boldsymbol{T}_{(p_1)} - & \lambda \boldsymbol{u}_{p_1}(\boldsymbol{u}_{p_3} \boxtimes \boldsymbol{u}_{p_2})^{\mathsf{T}}\|^2 = \\ & \|\boldsymbol{\mathcal{T}}\|^2 - \lambda \mathrm{trace}\{\boldsymbol{T}^{\mathsf{H}}_{(p_1)}\boldsymbol{u}_{p_1}(\boldsymbol{u}_{p_3} \boxtimes \boldsymbol{u}_{p_2})^{\mathsf{T}}\} - \\ & - \lambda^* \mathrm{trace}\{\boldsymbol{T}_{(p_1)}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2})\boldsymbol{u}^{\mathsf{T}}_{p_1}\} + \lambda^*\lambda.\end{aligned}$$

The scalar λ is actually the contraction of $\boldsymbol{\mathcal{T}}$ onto the unit rank-1 tensor $\boldsymbol{u}_{p_1} \otimes \boldsymbol{u}_{p_2} \otimes \boldsymbol{u}_{p_3}$, so that it can be written as

$$\begin{aligned}\lambda = \mathrm{trace}\{\boldsymbol{T}^{\mathsf{H}}_{(p_1)}\boldsymbol{u}_{p_1}(\boldsymbol{u}_{p_3} \boxtimes \boldsymbol{u}_{p_2})^{\mathsf{T}}\} = \\ = \boldsymbol{u}^{\mathsf{H}}_{p_1}\boldsymbol{T}_{(p_1)}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2}).\end{aligned} \tag{2}$$

Plugging (3) into equation (2), we obtain, after simplifications,

$$\|\boldsymbol{T}_{(p_1)} - \boldsymbol{X}^{\mathrm{ST}}_{(p_1)}\|^2 = \|\boldsymbol{\mathcal{T}}\|^2 - |\lambda|^2.$$

Note that $\boldsymbol{u}^{\mathsf{H}}_{p_1}\boldsymbol{T}_{(p_1)} = \|\boldsymbol{T}_{(p_1)}\|_2\, \boldsymbol{v}^{\mathsf{H}}_{p_1}$ for the principal singular triplet $(\boldsymbol{u}_{p_1}, \boldsymbol{v}_{p_1}, \|\boldsymbol{T}_{(p_1)}\|_2)$ of $\boldsymbol{T}_{(p_1)}$, where $\|\cdot\|_2$ stands for the spectral norm. Hence

$$\begin{aligned}\lambda = \boldsymbol{u}^{\mathsf{H}}_{p_1}\boldsymbol{T}_{(p_1)}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2}) = \|\boldsymbol{T}_{(p_1)}\|_2\, \boldsymbol{v}^{\mathsf{H}}_{p_1}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2}) \\ \implies |\lambda|^2 = \|\boldsymbol{T}_{(p_1)}\|_2^2\, |\boldsymbol{v}^{\mathsf{H}}_{p_1}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2})|^2,\end{aligned}$$

which leads to the following approximate error

$$\|\boldsymbol{T}_{(p_1)} - \boldsymbol{X}^{\mathrm{ST}}_{(p_1)}\|^2 = \|\boldsymbol{\mathcal{T}}\|^2 - \|\boldsymbol{T}_{(p_1)}\|_2^2\, |\boldsymbol{v}^{\mathsf{H}}_{p_1}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2})|^2.$$

On the other hand for SeROAP, we have $\boldsymbol{X}^{\mathrm{Se}}_{(p_1)} = \boldsymbol{T}_{(p_1)}\boldsymbol{w}\boldsymbol{w}^{\mathsf{H}}$, for $\boldsymbol{w} = \boldsymbol{v}^*_{p_2} \boxtimes \boldsymbol{u}$. Thus,

$$\begin{aligned}\|\boldsymbol{T}_{(p_1)} - \boldsymbol{X}^{\mathrm{Se}}_{(p_1)}\|^2 &= \|\boldsymbol{T}_{(p_1)} - \boldsymbol{T}_{(p_1)}\boldsymbol{w}\boldsymbol{w}^{\mathsf{H}}\|^2 \\ &= \|\boldsymbol{\mathcal{T}}\|^2 + \|\boldsymbol{T}_{(p_1)}\boldsymbol{w}\|^2\, \|\boldsymbol{w}\|^2 - 2\|\boldsymbol{T}_{(p_1)}\boldsymbol{w}\|^2 \\ &= \|\boldsymbol{\mathcal{T}}\|^2 - \boldsymbol{w}^{\mathsf{H}}\boldsymbol{T}^{\mathsf{H}}_{(p_1)}\boldsymbol{T}_{(p_1)}\boldsymbol{w}.\end{aligned} \tag{3}$$

The vectors $\boldsymbol{u}$ and $\boldsymbol{v}_{p_2}$ are the principal left and right singular vectors of the unfolding $\boldsymbol{S}_{(p_2)}$, as described in Algorithm 1. The eigenvalue decomposition of $\boldsymbol{T}^{\mathsf{H}}_{(p_1)}\boldsymbol{T}_{(p_1)}$ can be expressed by

$$\boldsymbol{T}^{\mathsf{H}}_{(p_1)}\boldsymbol{T}_{(p_1)} = \|\boldsymbol{T}_{(p_1)}\|_2^2 \boldsymbol{v}_{p_1}\boldsymbol{v}^{\mathsf{H}}_{p_1} + \boldsymbol{V}, \tag{4}$$

where $\boldsymbol{V}$ is a semidefinite positive matrix. Plugging (4) into (3)

$$\begin{aligned}\|\boldsymbol{T}_{(p_1)} - \boldsymbol{X}^{\mathrm{Se}}_{(p_1)}\|^2 &= \|\boldsymbol{\mathcal{T}}\|^2 - \|\boldsymbol{T}_{(p_1)}\|_2^2 \boldsymbol{w}^{\mathsf{H}}\boldsymbol{v}_{p_1}\boldsymbol{v}^{\mathsf{H}}_{p_1}\boldsymbol{w} - c \\ &= \|\boldsymbol{\mathcal{T}}\|^2 - \|\boldsymbol{T}_{(p_1)}\|_2^2 |\boldsymbol{v}^{\mathsf{H}}_{p_1}\boldsymbol{w}|^2 - c,\end{aligned}$$

with $c = \boldsymbol{w}^{\mathsf{H}}\boldsymbol{V}\boldsymbol{w} \geq 0$.

To complete the proof of the theorem, we just need to show that $|\boldsymbol{v}^{\mathsf{H}}_{p_1}\boldsymbol{w}|^2 \geq |\boldsymbol{v}^{\mathsf{H}}_{p_1}(\boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2})|^2$, or equivalently that

$$|\langle \boldsymbol{w}, \boldsymbol{v}_{p_1}\rangle| \geq |\langle \boldsymbol{u}^*_{p_3} \boxtimes \boldsymbol{u}^*_{p_2}, \boldsymbol{v}_{p_1}\rangle|.$$

This is true, because $\boldsymbol{w}$ is by construction (cf. Algorithm 1) the vector closest to $\boldsymbol{v}_{p_1}$ among all vectors of the form $\boldsymbol{a} \boxtimes \boldsymbol{b}$ where $\boldsymbol{a}$ and $\boldsymbol{b}$ have unit norm. □

Corollary 2. *Let $\boldsymbol{\mathcal{X}}^{TH}$ be the rank-1 approximation delivered by THOSVD algorithm. Then the approximate error*

$$\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{Se}\| \leq \|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{TH}\|$$

holds for any mode ordering defined by $\boldsymbol{p}$.

Proof. The proof follows directly from Theorem 1 and Theorem 7.2 in [36], in which the approximate error computed with ST-HOSVD is at most equal to that obtained with THOSVD for three-way tensors, when at least one of the multilinear ranks of the approximating tensor is equal to 1. □

IV. Numerical experiments

In order to corroborate the result of Theorem 1, we have performed experiments for four three-way scenarios: $3 \times 4 \times 5$, $3 \times 4 \times 20$, $3 \times 20 \times 20$ and $20 \times 20 \times 20$ tensors. With these scenarios, we have intended to take into account cases with equal dimensions or not. For each of them, a sample of 300 complex tensors with real and imaginary parts uniformly distributed in $[-1, 1]$ was generated thereby ensuring a coherent and meaningful comparison. We have chosen the ordering $\boldsymbol{p} = [3\ 1\ 2]$ for the modes.

In order to characterize the comparison between SeROAP and ST-HOSVD algorithms, we introduce the metric

$$\Delta\phi_s = 1 - \frac{\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{\mathrm{Se}}\|}{\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{\mathrm{ST}}\|},$$

which measures how much SeROAP outperforms ST-HOSVD in terms of approximate error. Notice that $0 \leq \Delta\phi_s < 1$. The higher is $\Delta\phi_s$, the better is the performance of SeROAP over ST-HOSVD. Figure 1 evinces this for any fixed permutation vector $\boldsymbol{p}$. Notice that the higher the dimensions of the tensors, the smaller is the variance of $\Delta\phi_s$. This can be explained by the fact that the degree of freedom to construct rank-1 tensors in some tensor space is large for higher dimensions, which can result in a vast number of rank-1 tensors with nearby approximate errors. Thus, the approximations delivered by SeROAP and ST-HOSVD yield close approximate errors.

V. Discussion on higher-order tensors

A better performance in terms of approximate error of SeROAP over ST-HOSVD and THOSVD is not ensured anymore for higher-order tensors. Actually, a smaller approximate error obtained with ST-HOSVD over THOSVD is not ensured either [36]. Nevertheless, it turns out by numerical experiments that SeROAP delivers a better rank-1 approximation than THOSVD with high probability for 4-th order tensors. For $N \geq 5$, the successive projections of SeROAP introduces substantial errors in the rank-1 tensor approximation, so that the comparison is not recommended in this case.

We have evaluated the performance of the algorithms under four scenarios for 4-th order tensors: $5 \times 10 \times 15 \times 20$, $5 \times 10 \times 20 \times 20$, $5 \times 20 \times 20 \times 20$ and $20 \times 20 \times 20 \times 20$ tensors. We assume a sample of complex tensors with entries distributed as before. We also define the metric

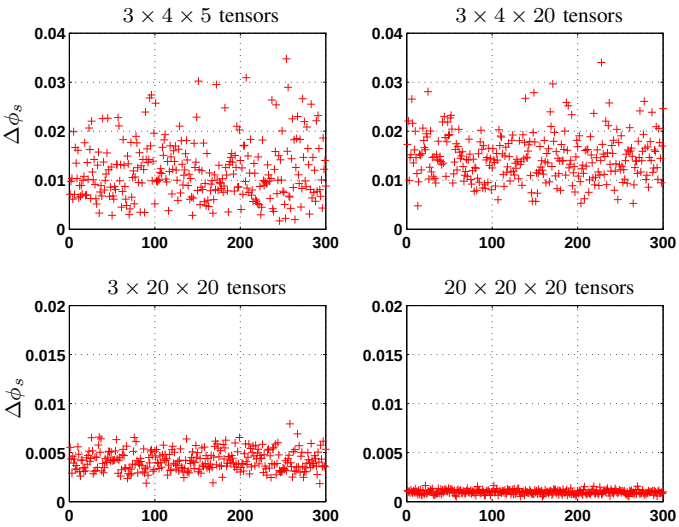


Fig. 1. SeROAP vs ST-HOSVD - $\boldsymbol{p} = [3\ 1\ 2]$.

$$\Delta\phi_t = 1 - \frac{\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{\mathrm{Se}}\|}{\|\boldsymbol{\mathcal{T}} - \boldsymbol{\mathcal{X}}^{\mathrm{TH}}\|}$$

to compare SeROAP with THOSVD. We have chosen the order $\boldsymbol{p} = [1\ 2\ 3\ 4]$ for the modes. Table I illustrates the performance of the algorithms in terms of the percentage of $\Delta\phi_t > 0$ for a sample of 10^5 tensors. In all scenarios SeROAP is statistically better than the THOSVD algorithm with a probability larger than 80%. Thus, our proposed algorithm can be a reasonable choice for computing a rank-1 approximation for 4-th order tensors.

TABLE I
Percentage of tensors in which SeROAP outperforms THOSVD.

Scenario of 4-th order tensors	% $\Delta\phi_t > 0$
$5 \times 10 \times 15 \times 20$	89.143 %
$5 \times 10 \times 20 \times 20$	88.435 %
$5 \times 20 \times 20 \times 20$	82.677 %
$20 \times 20 \times 20 \times 20$	92.874 %

VI. Conclusions

The proposed SeROAP algorithm has shown to be a simple solution to the rank-1 tensor approximation problem. For three-way tensors, SeROAP is *always* the most accurate algorithm among the competing ones, and hence the best choice for initializing iterative algorithms to estimate the best rank-1 approximation. The proof is delineated in Theorem 1. For small orders, we saw that its complexity can be smaller than that of THSOVD, and still competitive with respect to that of ST-HOSVD. According to our experiments, SeROAP is generally a good option for 4-th order tensors. Indeed, we have seen that it delivers better rank-1 approximations than THOSVD, for most tensors drawn randomly according to an absolutely continuous distribution.

REFERENCES

[1] E. Moreau, "A generalization of joint-diagonalization criteria for source separation," *Signal Processing, IEEE Transactions on*, vol. 49, no. 3, pp. 530–541, 2001.

[2] P. Comon and C. Jutten, Eds., *Handbook of Blind Source Separation, Independent Component Analysis and Applications*, Academic Press, Oxford UK, Burlington USA, 2010, ISBN: 978-0-12-374726-6, hal-00460653.

[3] X. Luciani and L. Albera, "Joint eigenvalue decomposition of non-defective matrices based on the lu factorization with application to ica," *Signal Processing, IEEE Transactions on*, vol. 63, no. 17, pp. 4594–4608, 2015.

[4] C. De Luigi and E. Moreau, "Optimal joint diagonalization of complex symmetric third-order tensors. application to separation of non circular signals," in *Independent Component Analysis and Signal Separation*, pp. 25–32. Springer, 2007.

[5] A. L. F. de Almeida, G. Favier, and J. C. M. Mota, "Parafac-based unified tensor modeling for wireless communication systems with application to blind multiuser equalization," *Signal Processing*, vol. 87, no. 2, pp. 337–351, Feb. 2007.

[6] A. Smilde, R. Bro, and P. Geladi, *Multi-way analysis: applications in the chemical sciences*, John Wiley & Sons, 2005.

[7] X. Luciani and L. Albera, "Canonical polyadic decomposition based on joint eigenvalue decomposition," *Chemometrics and Intelligent Laboratory Systems*, vol. 132, pp. 152–167, 2014.

[8] H. Becker, L. Albera, P. Comon, M. Haardt, G. Birot, F. Wendling, M. Gavaret, C.-G. Bénar, and I. Merlet, "Eeg extended source localization: tensor-based vs. conventional methods," *NeuroImage*, vol. 96, pp. 143–157, 2014.

[9] S. Sahnoun and P. Comon, "Joint source estimation and localization," *IEEE Trans. on Signal Processing*, vol. 63, no. 10, 2015.

[10] B. Savas, *Algorithms in Data Mining using Matrix and Tensor Metho ds*, Ph.D. thesis, Linköping Univ. Tech., 2008.

[11] H. A. L. Kiers, "Towards a standardized notation and terminology in multiway analysis," *J. Chemometrics*, pp. 105–122, 2000.

[12] J. M. Ten Berge, "Kruskal's polynomial for 2× 2× 2 arrays and a generalization to 2× n× n arrays," *Psychometrika*, vol. 56, no. 4, pp. 631–636, 1991.

[13] J. Brachat, P. Comon, B. Mourrain, and E. Tsigaridas, "Symmetric tensor decomposition," *Linear Algebra and its Applications*, vol. 433, no. 11, pp. 1851–1872, 2010.

[14] L. De Lathauwer, "A link between the canonical decomposition in multilinear algebra and simultaneous matrix diagonalization," *SIAM Journal on Matrix Analysis and Applications*, vol. 28, no. 3, pp. 642–666, 2006.

[15] L. Oeding and G. Ottaviani, "Eigenvectors of tensors and algorithms for waring decomposition," *Journal of Symbolic Computation*, vol. 54, pp. 9–35, 2013.

[16] C. J. Hillar and L.-H. Lim, "Most tensor problems are np-hard," *Journal of the ACM (JACM)*, vol. 60, no. 6, pp. 45, 2013.

[17] V. De Silva and L.-H. Lim, "Tensor rank and the ill-posedness of the best low-rank approximation problem," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1084–1127, 2008.

[18] P. Comon, X. Luciani, and A. L. F. de Almeida, "Tensor decompositions, alternating least squares and other tales," *Journal of Chemometrics*, vol. 23, no. 7-8, pp. 393–405, 2009.

[19] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[20] M. Rajih, P. Comon, and R. A. Harshman, "Enhanced line search: A novel method to accelerate parafac," *SIAM Journal on Matrix Analysis and Applications*, vol. 30, no. 3, pp. 1128–1147, 2008.

[21] L. Sorber, M. Van Barel, and L. De Lathauwer, "Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-(l_r,l_r,1) terms, and a new generalization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 695–720, 2013.

[22] G. Tomasi and R. Bro, "A comparison of algorithms for fitting the parafac model," *Computational Statistics & Data Analysis*, vol. 50, no. 7, pp. 1700–1734, 2006.

[23] P. Paatero, "A weighted non-negative least squares algorithm for three-way Parafac factor analysis," *Chemometrics Intell. Lab. Syst.*, vol. 38, pp. 223–242, 1997.

[24] W. Hackbusch, *Tensor spaces and numerical tensor calculus*, vol. 42, Springer Science & Business Media, 2012.

[25] P. Comon, "Tensors: a brief introduction," *IEEE Sig. Proc. Magazine*, vol. 31, no. 3, pp. 44–53, May 2014, hal-00923279.

[26] H. Abo, G. Ottaviani, and C. Peterson, "Induction for secant varieties of segre varieties," *Transactions of the American Mathematical Society*, vol. 361, no. 2, pp. 767–792, 2009.

[27] J. M. Landsberg, *Tensors: geometry and applications*, American Mathematical Society, 2012.

[28] T. Zhang and G. H. Golub, "Rank-one approximation to high order tensors," *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 2, pp. 534–550, 2001.

[29] X. Shi, H. Ling, J. Xing, and W. Hu, "Multi-target tracking by rank-1 tensor approximation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2387–2394.

[30] Y. Yang, Y. Feng, X. Huang, and J. AK Suykens, "Rank-1 tensor properties with applications to a class of tensor optimization problems," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 171–196, 2016.

[31] J. B. Lasserre, "Global optimization with polynomials and the problem of moments," *SIAM Journal on Optimization*, vol. 11, no. 3, pp. 796–817, 2001.

[32] M. A. Bucero and B. Mourrain, "Border basis relaxation for polynomial optimization," *arXiv preprint arXiv:1404.5489*, 2014.

[33] J. Nie and L. Wang, "Semidefinite relaxations for best rank-1 tensor approximations," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 3, pp. 1155–1179, 2014.

[34] A. Uschmajew, "Local convergence of the alternating least squares algorithm for canonical tensor approximation," *SIAM Jour. matrix Analysis*, vol. 33, no. 2, pp. 639–652, 2012.

[35] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[36] N. Vannieuwenhoven, R. Vandebril, and K. Meerbergen, "A new truncation strategy for the higher-order singular value decomposition," *SIAM Journal on Scientific Computing*, vol. 34, no. 2, pp. A1027–A1052, 2012.

[37] P. Comon and G. H. Golub, "Tracking a few extreme singular values and vectors in signal processing," *Proceedings of the IEEE*, vol. 78, no. 8, pp. 1327–1343, 1990.

[38] A. P. da Silva, P. Comon, and de Almeida A. L. F., "Rank-1 tensor approximation methods and application to deflation," *arXiv:1508.05273*, August 2015.