NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 0000; **00**:1–34
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nla

New algorithms for tensor decomposition based on a reduced functional

Stefan Kindermann[1] and Carmeliza Navasca[2]

[1] *Industrial Mathematics Institute, Johannes Kepler Universitat Linz, Altenbergerstrasse 69, A-4040 Linz, Austria, kindermann@indmath.uni-linz.ac.at*
[2] *Department of Mathematics, University of Alabama, Birmingham, AL, USA, 35294-117, cnavasca@uab.edu*

SUMMARY

We study the least-squares functional of the canonical polyadic tensor decomposition for third order tensors by eliminating one factor matrix, which leads to a reduced functional. An analysis of the reduced functional leads to several equivalent optimization problem, like a Rayleigh quotient or a projection. These formulations are the basis of several new algorithms: the Centroid Projection method for efficient computation of suboptimal solutions and fixed-point iteration methods for approximating the best rank-1 and the best rank-R decompositions under certain nondegeneracy conditions. Copyright © 0000 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: tensor decomposition, nonlinear least-squares method, tensor rank

1. INTRODUCTION

In 1927, Hitchcock [1, 2] introduced the idea that a tensor is decomposable into a sum of a finite number of rank-1 tensors. Today, this decomposition is referred to as the canonical polyadic (CP) tensor decomposition (also known as CANDECOMP [3] or PARAFAC [4]). CP tensor decomposition reduces a third order tensor to a linear combination of rank-1 tensors, i.e.,

$$(\mathcal{A})_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}, \tag{1.1}$$

where $\mathcal{A} \in \mathbb{R}^{I\times J\times K}$, $\mathbf{a_r} = (a_{ir})_{i=1}^{I} \in \mathbb{R}^I, \mathbf{b_r} = (b_{jr})_{j=1}^{J} \in \mathbb{R}^J$ and $\mathbf{c_r} = (c_{kr})_{k=1}^{K} \in \mathbb{R}^K$. The column vectors $\mathbf{a_r}, \mathbf{b_r}$ and $\mathbf{c_r}$ form the so-called factor matrices $\mathbf{A} \in \mathbb{R}^{I\times R}$, $\mathbf{B} \in \mathbb{R}^{J\times R}$ and $\mathbf{C} \in \mathbb{R}^{K\times R}$. The tensorial rank [2, 5] is the minimum $R \in \mathbb{N}$ such that $\mathcal{T}$ can be expressed as a sum of R rank-1 tensors.

The problem of interest is to find – if it exists – the *best* approximate tensor representable in a CP format with a tensorial rank R from a given (possibly noisy) tensor $\mathcal{T} \in \mathbb{R}^{I\times J\times K}$. A standard approach for this task is to minimize the Frobenius norm of the residual tensor in the least-square sense:

$$\mathfrak{J}(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{i,j,k} \left((\mathcal{T})_{ijk} - \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} \right)^2. \tag{1.2}$$

*Correspondence to: Stefan Kindermann, Industrial Mathematics Institute, Johannes Kepler Universitat Linz, Altenbergerstrasse 69, A-4040 Linz, Austria. E-mail: kindermann@indmath.uni-linz.ac.at

Contract/grant sponsor: C.N. is in part supported by the National Science Foundation DMS-0915100.

A popular iterative method for approximating the given tensor $\mathcal{T}$ via its factors $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ is called the Alternating Least-Squares (ALS) technique [6, 7, 8]. Independently, ALS was introduced by Carol and Chang [3] and Harshman [4] in 1970. The ALS method is an application of the nonlinear block Gauss-Seidel algorithm [9] where the nonlinear optimization (1.2) is reduced into several least-squares subproblems which are solved iteratively with subsequent updates of the minimizing factors.

In this paper, the analysis is based on the minimization of the objective function (1.2) through an elimination of one factor, that is factor $\mathbf{A}$, and thereby a reduction to a minimization over the factors $\mathbf{B}$ and $\mathbf{C}$. This analysis is motivated by the ALS algorithm. The discussion is restricted to third order tensors, but some of the analysis applies to higher order tensors.

The reduced functional is the basis for an analysis and several numerical algorithms for the minimization problem (1.2). In Section 3, we define the reduced functional and investigate some of its properties. The corresponding minimization problem allows reformulations into several forms: as a Rayleigh quotient type functional and a weighted projection onto the Khatri-Rao range of $\mathbf{B}$ and $\mathbf{C}$. We state necessary conditions for the non-existence of minimizers, derive the optimality conditions and state some equivalent functionals in the case $R = 1, 2$. The analysis on the reduced functional leads to several numerical methods to compute suboptimal or optimal solutions in some special cases.

In Section 4 we lay the mathematical foundation for some new optimization algorithms and illustrate the theory with some elementary examples. In Section 5 we describe the corresponding algorithms, which are useful in different situations: the Centroid Projection Algorithm is a simple method to compute suboptimal solutions to the least squares minimization problem related to (1.2). In Section 4.2, we propose an algorithm, FP-R1 (Fixed Point iteration for Rank-1 decomposition), for the rank-1 least squares problem. The main idea for this algorithm arises from the optimality conditions for the reduced functional. A globalization strategy applied to FP-R1 with the goal to compute all stationary points leads to two variants of this algorithm, FP-R1(RIG) (FP-R1 with Random Initial Guess) and FP-R1(APIG) (FP-R1 with A-Priori Initial Guess). Furthermore, we derive another numerical method, FP-EX (Fixed Point iteration for EXact decomposition of rank-R tensors) which is designed to solve the least squares minimization problem when the minimal value is 0, i.e., when a rank-R decomposition exists. We additionally prove some a-priori estimates allowing us to use this algorithm for the computation of reasonable suboptimal solutions in the general case (Algorithm FP-INEX (Fixed Point iteration for INEXact decomposition)).

Finally in Section 6, we perform some numerical experiments for the proposed algorithms.

2. PRELIMINARIES

We denote the scalars in $\mathbb{R}$ with lower-case letters $(a, b, \dots)$ and the vectors with bold lower-case letters $(\mathbf{a}, \mathbf{b}, \dots)$. The matrices are written as bold upper-case letters $(\mathbf{A}, \mathbf{B}, \dots)$ and the symbol for tensors are calligraphic letters $(\mathcal{A}, \mathcal{B}, \dots)$. The subscripts represent the following scalars: $(\mathcal{A})_{ijk} = a_{ijk}$, $(\mathbf{A})_{ij} = a_{ij}$, $(\mathbf{a})_i = a_i$.

The order of a tensor refers to the cardinality of the index set. A matrix is a second-order tensor and a vector is a first-order tensor. The scalar product of $\mathcal{T}, \mathcal{R} \in \mathbb{R}^{I \times J \times K}$ is defined as

$$\langle \mathcal{T}, \mathcal{L} \rangle = \sum_{ijk} (\mathcal{T})_{ijk} (\mathcal{L})_{ijk}.$$

The Frobenius norm of $\mathcal{A} \in \mathbb{R}^{I \times J \times K}$ is defined as

$$\|\mathcal{A}\|_F^2 = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} |a_{ijk}|^2 = \langle \mathcal{A}, \mathcal{A} \rangle,$$

which is a direct extension of the Frobenius norm of a matrix. Furthermore, we denote by $\cdot$ the usual matrix product.

Definition 2.1
The Khatri-Rao product of $\mathbf{A} \in \mathbb{R}^{I \times R}$ and $\mathbf{B} \in \mathbb{R}^{J \times R}$ is defined as

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a_1} \otimes \mathbf{b_1}\ \mathbf{a_2} \otimes \mathbf{b_2}\ \dots\ \mathbf{a_R} \otimes \mathbf{b_R}] \in \mathbb{R}^{IJ \times R}$$

when $\mathbf{A} = [\mathbf{a_1}\ \mathbf{a_2}\ \dots\ \mathbf{a_R}]$ and $\mathbf{B} = [\mathbf{b_1}\ \mathbf{b_2}\ \dots\ \mathbf{b_R}]$.

Here, $\mathbf{a} \otimes \mathbf{b}$ denotes the Kronecker product of two vectors $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{b} \in \mathbb{R}^J$ yielding a vector of size IJ with entries that are all possible products of the entries in $\mathbf{a}$ and $\mathbf{b}$.

Definition 2.2 (Tucker mode-n product)
Given a tensor $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ and matrices $\mathbf{A} \in \mathbb{R}^{I_1 \times J_1}$, $\mathbf{B} \in \mathbb{R}^{I_2 \times J_2}$ and $\mathbf{C} \in \mathbb{R}^{I_3 \times J_3}$, then the Tucker mode-$n$ products are the following:

$$\begin{aligned}
\mathcal{T} \bullet_1 \mathbf{A} := (\mathcal{T} \bullet_1 \mathbf{A})_{j_1 i_2 i_3} &= \sum_{i_1=1}^{I_1} \mathcal{T}_{i_1 i_2 i_3} a_{i_1 j_1},\ \forall j_1, i_2, i_3 \quad \text{(mode-1 product)}\\
\mathcal{T} \bullet_2 \mathbf{B} := (\mathcal{T} \bullet_2 \mathbf{B})_{i_1 j_2 i_3} &= \sum_{i_2=1}^{I_2} \mathcal{T}_{i_1 i_2 i_3} b_{i_2 j_2},\ \forall j_2, i_1, i_3 \quad \text{(mode-2 product)}\\
\mathcal{T} \bullet_3 \mathbf{C} := (\mathcal{T} \bullet_3 \mathbf{C})_{i_1 i_2 j_3} &= \sum_{i_3=1}^{I_3} \mathcal{T}_{i_1 i_2 i_3} c_{i_3 j_3},\ \forall j_3, i_1, i_2 \quad \text{(mode-3 product).}
\end{aligned}$$

Moreover, the Tucker mode products can be combined as in this example:

$$\mathcal{T} \bullet_{2,3} (\mathbf{B}, \mathbf{C}) := (\mathcal{T} \bullet_{2,3} (\mathbf{B}, \mathbf{C}))_{i_1 r} := \sum_{i_2=1}^{I_2} \sum_{i_3=1}^{I_3} \mathcal{T}_{i_1 i_2 i_3} b_{i_2 r} c_{i_3 r}$$

where $\mathbf{B} \in \mathbb{R}^{I_2 \times R}$ and $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$.

Definition 2.3 (outer product of vectors)
For vectors $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{b} \in \mathbb{R}^J$ the outer product $\mathbf{a} \circ \mathbf{b}$ is the $I \times J$ matrix with entries

$$(\mathbf{a} \circ \mathbf{b})_{i,j} = a_i b_j,\ \forall i, j$$

similarly, the outer product of three vectors $\mathbf{a} \in \mathbb{R}^I$, $\mathbf{b} \in \mathbb{R}^J$, $\mathbf{c} \in \mathbb{R}^K$ is the $I \times J \times K$ tensor

$$(\mathbf{a} \circ \mathbf{b} \circ \mathbf{c})_{i,j,k} = a_i b_j c_k,\ \forall i, j, k.$$

3. THE REDUCED OBJECTIVE FUNCTIONAL

We state the reduced objective functional obtained by eliminating one matrix in $\mathfrak{J}$. This is one of the main tools for our analysis. We note that such a functional has already been considered in [10].

Recall the least-squares objective functional in (1.2):

$$\mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \left\| \mathcal{T} - \sum_{r=1}^{R} \mathbf{a_r} \circ \mathbf{b_r} \circ \mathbf{c_r} \right\|_F^2 \tag{3.1}$$

where $\|\cdot\|_F$ is the Frobenius norm. The goal is to find minimizers $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ of the problem

$$\inf_{\mathbf{A},\mathbf{B},\mathbf{C}} \mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}).$$

It is well-known that this infimum is not necessarily attained see, e.g., [11].

Lemma 3.1
Let $\mathbf{B}, \mathbf{C}$ be fixed. The solution to the minimization problem

$$\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}] := \mathrm{argmin}_{\mathbf{A} \in \mathbb{R}^{I \times R}} \mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}) \tag{3.2}$$

exists. In fact, a minimizer is given by

$$\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}] = \mathcal{T} \bullet_{2,3} (\mathbf{B}, \mathbf{C}) \cdot \mathbf{G}^{\dagger}, \tag{3.3}$$

where $\mathbf{G}^{\dagger}$ is the pseudo-inverse of $\mathbf{G}$ which is defined as

$$\mathbf{G} = (\mathbf{B} \odot \mathbf{C})^T \cdot (\mathbf{B} \odot \mathbf{C}) \in \mathbb{R}^{R \times R}. \tag{3.4}$$

Proof
With $\mathbf{B}, \mathbf{C}$ being fixed, (3.2) is a usual finite dimensional linear least squares problem for which it is well-known that a solution exists. The optimality condition

$$\mathbf{A} \cdot \mathbf{G} = \mathcal{T} \bullet_{2,3} (\mathbf{B}, \mathbf{C})$$

lead to (3.3). □

From the definition (3.4), it follows that $\mathbf{G}$ is a Gramian matrix for the vectors $\mathbf{b_r} \otimes \mathbf{c_r}$, $r = 1, \ldots R$, as well as the Hadamard product of $\mathbf{B}^T \cdot \mathbf{B}$ and $\mathbf{C}^T \cdot \mathbf{C}$. Note that $\mathbf{G}$ depends on $\mathbf{B}$ and $\mathbf{C}$ but we omitted this dependence to avoid exuberant notation. It follows easily that $\mathbf{G}$ is symmetric, and thus so is $\mathbf{G}^{\dagger}$. Moreover, the pseudo-inverse satisfies the Moore-Penrose equation $\mathbf{G}^{\dagger} \cdot \mathbf{G} \cdot \mathbf{G}^{\dagger} = \mathbf{G}^{\dagger}$.

By minimizing the original objective functional over $\mathbf{A}$, we now define the reduced objective functional as

$$\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) := \mathfrak{J}(\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}], \mathbf{B}, \mathbf{C}) \tag{3.5}$$

where $\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}]$ is a minimizer in (3.2). This definition is not dependent on the minimizer we take. Also, considering the reduced functional does not alter the original problem.

Let $\{\mathbf{B}_n, \mathbf{C}_n\}$ be a sequence. We say that $\{\mathbf{B}_n, \mathbf{C}_n\}$ is a minimizing sequence for $\mathfrak{J}_{red}$ if

$$\mathfrak{J}_{red}(\mathbf{B}_n, \mathbf{C}_n) \overset{n \to \infty}{\longrightarrow} \inf_{\mathbf{B}, \mathbf{C}} \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}).$$

Proposition 3.2
If $\{\mathbf{B}_n, \mathbf{C}_n\}$ is a minimizing sequence for $\mathfrak{J}_{red}$, then $\{\tilde{\mathbf{A}}[\mathbf{B}_n, \mathbf{C}_n], \mathbf{B}_n, \mathbf{C}_n\}$ is a minimizing sequence of $\mathfrak{J}$ and the equality,

$$\inf_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \inf_{\mathbf{B}, \mathbf{C}} \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}),$$

holds. In particular, if $(\mathbf{B}_*, \mathbf{C}_*)$ are minimizers of $\mathfrak{J}_{red}$, then $(\tilde{\mathbf{A}}[\mathbf{B}_*, \mathbf{C}_*], \mathbf{B}_*, \mathbf{C}_*)$ are minimizers of $\mathfrak{J}$.

Proof
This is a straightforward consequence of the fact that $\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}]$ in (3.3) always exists. □

3.1. *Analysis of the reduced objective function*

The introduction of $\mathfrak{J}_{red}$ reduces the number of unknown factors by one. In this section, we explicitly calculate $\mathfrak{J}_{red}$. We now define a symmetric 4th-order tensor (and its matricization) which plays a similar role as $\mathbf{A}^T\mathbf{A}$ in the theory of the singular value decomposition for linear operators.

Definition 3.3
Let us define the fourth order tensor $\mathcal{M} \in \mathbb{R}^{J \times K \times J \times K}$ as

$$\mathcal{M}_{\alpha\beta\gamma\delta} := \sum_{i=1} \mathcal{T}_{i\alpha\beta} \mathcal{T}_{i\gamma\delta} \in \mathbb{R}^{J \times K \times J \times K}. \tag{3.6}$$

The matricization of $\mathcal{M} \in \mathbb{R}^{J\times K\times J\times K}$ is denoted by $\mathbf{M} \in \mathbb{R}^{JK\times JK}$ such that

$$(\mathbf{M})_{ij} = (\mathcal{M})_{\alpha\beta\gamma\delta} \tag{3.7}$$

where $i = [\alpha + (\beta - 1)J]$ and $j = [\gamma + (\delta - 1)J]$. Clearly $\mathbf{M}$ is symmetric. Now, we find the eigendecomposition:

$$\mathbf{M} = \overline{\mathbf{V}} \cdot \bar{\mathbf{S}} \cdot \overline{\mathbf{V}}^T$$

where $\overline{\mathbf{V}}$ is an orthogonal matrix and $\mathbf{S}$ is a diagonal matrix in which the eigenvalues $(\overline{\lambda}_i)_{i=1}^{JK} = \text{diag}(\bar{\mathbf{S}})$ are in nonincreasing order. The matricization of the column vectors of $\overline{\mathbf{V}} = (\overline{\mathbf{v}}_\mathbf{1}, \dots \overline{\mathbf{v}}_\mathbf{JK}) \in \mathbb{R}^{JK\times JK}$ are denoted by $\overline{\mathbf{V}}_\mathbf{i} \in \mathbb{R}^{J\times K}$, i.e.,

$$(\overline{\mathbf{v}}_\mathbf{i})_{[\alpha+(\beta-1)J]} = (\overline{\mathbf{V}}_\mathbf{i})_{\alpha,\beta},$$

and the rank of $\mathbf{M}$ is denoted by R_M.

Remark 3.4
Here we defined $\mathbf{M}$ and the corresponding singular vectors by contracting over the first mode (of size I). Of course, the same can be done by contracting over the second or third. Which of these possibilities is appropriate, depends on the dimension. In general, we think that it is best to contract over the mode of largest dimension. This has the effect that the complexity of our algorithms (which depends on the product of the remaining dimensions) is smaller than otherwise.

Remark 3.5
It was shown in [12] that due to the isomorphic group structures between the sets of invertible matrices and tensors, structures like symmetry and eigendecomposition are preserved through a matricization.

In accordance with the notation of Section 2 we also use the Tucker product for fourth order tensors, e.g.,

$$\mathcal{M} \bullet_{1,2,3,4} (\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) := \sum_{\alpha,\beta,\gamma,\delta} \mathcal{M}_{\alpha\beta\gamma\delta} a_\alpha b_\beta c_\gamma d_\delta \in \mathbb{R}.$$

Lemma 3.6
The reduced objective functional can be expressed in the following form:

$$\begin{aligned} \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) &= \langle \mathcal{T}, \mathcal{T} \rangle - \text{Tr}(\mathbf{G}^\dagger \cdot (\mathbf{B} \odot \mathbf{C})^T \cdot \mathbf{M} \cdot \mathbf{B} \odot \mathbf{C}) && (3.8) \\ &= \left(\|\mathcal{T}\|_F^2 - \sum_{r,s}^{R} (\mathbf{G}^\dagger)_{sr} \mathcal{M} \bullet_{1,2,3,4} (\mathbf{b_r}, \mathbf{c_r}, \mathbf{b_s}, \mathbf{c_s}) \right) && (3.9) \end{aligned}$$

where $\mathcal{M} \in \mathcal{R}^{J\times K\times J\times K}$ is defined in (3.6), $\mathbf{G}^\dagger$ is the the pseudo-inverse of $\mathbf{G}$ in (3.4) and Tr denotes the matrix trace.

Proof
Expanding (3.1) yields

$$\mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \langle \mathcal{T}, \mathcal{T} \rangle - 2\langle \mathcal{T}, (\mathbf{B} \odot \mathbf{C}) \cdot \mathbf{A}^T \rangle + \langle (\mathbf{B} \odot \mathbf{C}) \cdot \mathbf{A}^T, (\mathbf{B} \odot \mathbf{C}) \cdot \mathbf{A}^T \rangle.$$

Using (3.3), the symmetry of $\mathbf{G}$, and (3.7) this reduced to

$$\begin{aligned} \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) &= \mathfrak{J}(\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}], \mathbf{B}, \mathbf{C}) = \langle \mathcal{T}, \mathcal{T} \rangle - \langle \mathcal{T}, (\mathbf{B} \odot \mathbf{C}) \cdot \mathbf{A}[\mathbf{B}, \mathbf{C}]^T \rangle \\ &= \langle \mathcal{T}, \mathcal{T} \rangle - \text{Tr}(\mathcal{T} \bullet_{2,3} (\mathbf{B}, \mathbf{C}) \cdot \mathbf{G}^\dagger \cdot \mathcal{T} \bullet_{2,3} (\mathbf{B}, \mathbf{C})^T) \\ &= \langle \mathcal{T}, \mathcal{T} \rangle - \text{Tr}(\mathbf{G}^\dagger \cdot (\mathbf{B} \odot \mathbf{C})^T \cdot \mathbf{M} \cdot \mathbf{B} \odot \mathbf{C}). \end{aligned}$$

□

It can be seen that the reduced functional only depends on the matrix range of $\mathbf{B} \odot \mathbf{C}$ (the so-called Khatri-Rao range defined below in Definition 3.21) and not on $\mathbf{B}$ and $\mathbf{C}$ itself, as the following Lemma shows.

Lemma 3.7
Let $\mathbf{U}, \mathbf{\Sigma}, \mathbf{W}$ be the matrices in the singular value decomposition of $(\mathbf{B} \odot \mathbf{C})$, i.e., $(\mathbf{B} \odot \mathbf{C}) = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{W}^T \in \mathbb{R}^{JK \times R}$ with $\mathbf{U} \in \mathbb{R}^{JK \times JK}$ orthogonal, $\mathbf{\Sigma} \in \mathbb{R}^{JK \times R}$ diagonal and $\mathbf{W} \in \mathbb{R}^{R \times R}$ orthogonal. Then,

$$\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) = \left(\|\mathcal{T}\|_F^2 - \sum_{r=1}^{\bar{R}} \langle \mathbf{u_k}, \mathbf{M}\mathbf{u_k} \rangle \right)$$

where $\mathbf{M}$ is the matricization of $\mathcal{M}$ in (3.6) and $\bar{R} = \text{rank}(\mathbf{\Sigma}) = \text{rank}(\mathbf{B} \odot \mathbf{C})$ and $\mathbf{u_k}$ is the k-th column of $\mathbf{U}$.

Proof
Starting from (3.8) it follows that

$$\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) = \langle \mathcal{T}, \mathcal{T} \rangle - \langle \mathbf{P}_{\mathbf{B} \odot \mathbf{C}}, \mathbf{M} \rangle \tag{3.10}$$

with

$$\mathbf{P}_{\mathbf{B} \odot \mathbf{C}} = (\mathbf{B} \odot \mathbf{C}) \mathbf{G}^\dagger (\mathbf{B} \odot \mathbf{C})^T. \tag{3.11}$$

From the Moore-Penrose equations, it follows that $\mathbf{P}$ is the orthogonal projector onto the range of $\mathbf{B} \odot \mathbf{C}$,

$$\mathbf{P}_{\mathbf{B} \odot \mathbf{C}} = \sum_{k=1}^{\bar{R}} \mathbf{u_k} \circ \mathbf{u_k}, \tag{3.12}$$

which proves the Lemma. □

The previous lemma allows us to rewrite the minimization problem for $\mathfrak{J}_{red}$ into a Rayleigh quotient type problem.

Theorem 3.8
The minimization problem for $\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C})$ is equivalent to the following maximization problem

$$\sup_{\mathbf{u_1}, \dots, \mathbf{u_{\bar{R}}}} \sum_{r=1}^{\bar{R}} \langle \mathbf{u_r}, \mathbf{M}\mathbf{u_r} \rangle, \tag{3.13}$$

where $\mathbf{u_1}, \dots, \mathbf{u_{\bar{R}}}$ is an orthonormal basis of range$(\mathbf{B} \odot \mathbf{C})$ with $\bar{R} = \text{rank}(\mathbf{B} \odot \mathbf{C})$. Equivalence holds in the following sense: if $(\mathbf{B}, \mathbf{C})$ are (approximate) minimizers of $\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C})$, then any orthonormal basis of range$(\mathbf{B} \odot \mathbf{C})$ is a(n) (approximate) maximizer of (3.13). Conversely, if $\mathbf{u_1}, \dots \mathbf{u_{\bar{R}}}$ are (approximate) maximizers of (3.13), then the associated $(\mathbf{B}, \mathbf{C})$ are (approximate) minimizers of $\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C})$.

Proof
The only fact to proof is that in (3.13) we may take any orthogonal basis of the range of $\mathbf{B} \odot \mathbf{C}$, but this follows immediately from the orthogonal invariance of the trace. □

The new problem formulation (3.13) indicates why the least squares problem might not have a solution. Clearly, the functional $\sum_{r=1}^{\bar{R}} \langle \mathbf{u_r}, \mathbf{M}\mathbf{u_r} \rangle$ is continuous with respect to $\mathbf{u_1}, \dots \mathbf{u_{\bar{R}}}$. Also recall that these orthonormalized vectors lie in a compact set. However, the dependence of $\mathbf{u_i}$ on the matrix entries does not necessarily have to be continuous [13, Example 5.3]. The reason for a non-existing minimum is due to the fact that the space of matrices $\mathbf{B} \odot \mathbf{C}$ with rank $\overline{R}$ is not closed. In fact, a problem arises when the rank of the Khatri-Rao product decreases for a minimizing sequence.

Proposition 3.9
Let $(\mathbf{B}_n, \mathbf{C}_n)$ be a minimizing sequence of $\mathfrak{J}_{red}$ (and thus $(\tilde{\mathbf{A}}[\mathbf{B}_n, \mathbf{C}_n], \mathbf{B}_n, \mathbf{C}_n)$ a minimizing sequence of $\mathfrak{J}$) such that $\frac{\mathbf{B}_n}{\|\mathbf{B}_n\|}$ and $\frac{\mathbf{C}_n}{\|\mathbf{C}_n\|}$ converges to matrices $\mathbf{B}$ and $\mathbf{C}$, respectively. If the following rank condition

$$\liminf_{n \to \infty} \text{rank}(\mathbf{B}_n \odot \mathbf{C}_n) \leq \text{rank}(\mathbf{B} \odot \mathbf{C}) \tag{3.14}$$

holds, then $(\mathbf{B},\mathbf{C})$ is a minimizer of $\mathfrak{J}_{red}$ and $(\tilde{\mathbf{A}}[\mathbf{B},\mathbf{C}],\mathbf{B},\mathbf{C})$ is a minimizer of $\mathfrak{J}$. In particular, a solution to the least squares problem exists.

Proof
Denote by $\mathbf{w_i}$ the left singular vectors and by σ_i the ordered singular values of $(\mathbf{B}\odot\mathbf{C})$. Define $\tilde{\mathbf{B}}_n := \frac{\mathbf{B}_n}{\|\mathbf{B}_n\|}$, $\tilde{\mathbf{C}}_n := \frac{\mathbf{C}_n}{\|\mathbf{C}_n\|}$, and denote by $(\mathbf{u_1^n},\dots,\mathbf{u_{JK}^n})$ the left singular vectors of $\tilde{\mathbf{B}}_n\odot\tilde{\mathbf{C}}_n$, and by σ_i^n the corresponding (ordered) singular values. Since the singular vectors have norm 1, by compactness, we can find a converging subsequence (again denoted by subscript n) with some vectors $\mathbf{z_i}$ as limit

$$\mathbf{u_i^n}{\rightarrow}\mathbf{z_i} \quad i=1,\dots JK, \qquad \text{as } n\to\infty. \tag{3.15}$$

It is clear that $\operatorname{rank}(\mathbf{B}_n\odot\mathbf{C}_n)=\operatorname{rank}(\tilde{\mathbf{B}}_n\odot\tilde{\mathbf{C}}_n)$. By taking another subsequence (again denoted by subscript n) we can replace the $\liminf$ in (3.14) by a $\lim$ and assume that

$$\lim_{n\to\infty}\operatorname{rank}(\tilde{\mathbf{B}}_n\odot\tilde{\mathbf{C}}_n) =: \lim_{n\to\infty} R_n =: R^* \leq \operatorname{rank}(\mathbf{B}\odot\mathbf{C}) = \overline{R}. \tag{3.16}$$

Moreover, a scalar multiplication of a matrix does not alter its singular vectors, thus from Lemma 3.7 it follows that $(\tilde{\mathbf{B}}_n,\tilde{\mathbf{C}}_n)$ is a minimizing sequence of $\mathfrak{J}_{red}$ as well, hence using Lemma 3.7 and Theorem 3.8 we can also assume by the definition of a minimizing sequence that

$$\lim_{n\to\infty}\sum_{i=1}^{R_n}\langle\mathbf{u_i^n},\mathbf{M}\mathbf{u_i^n}\rangle = N, \tag{3.17}$$

where N is the supremum in (3.13). By assumption, it hold that $\tilde{\mathbf{B}}_n\to\mathbf{B}$ and $\tilde{\mathbf{C}}_n\to\mathbf{C}$, as $n\to\infty$, by continuity of the singular values (cf., e.g., [14, Weyl's Theorem]) we have that

$$\lim_{n\to\infty}\sigma_i^n\to\sigma_i \qquad \forall i=1,\dots JK, \tag{3.18}$$

and by definition of the rank, $\sigma_i>0$ for $i=1,\dots\overline{R}$.

We now show that the limit vectors $\mathbf{z_i}$, $i=1,\dots R^*$ are the first R^* left singular vectors of $\mathbf{B}\odot\mathbf{C}$ and that they are maximizers in (3.13). Indeed, the left singular vectors $\mathbf{u}_i^n$ are also eigenvectors of $(\tilde{\mathbf{B}}_n\odot\tilde{\mathbf{C}}_n)(\tilde{\mathbf{B}}_n\odot\tilde{\mathbf{C}}_n)^T$. Thus, by (3.15), (3.18), and the positivity of the first $\overline{R}$ singular values σ_i we find that

$$\mathbf{z_i} = \lim_{n\to\infty}\mathbf{u}_i^n = \lim_{n\to\infty}\frac{1}{(\sigma_i^n)^2}(\mathbf{B}_n\odot\mathbf{C}_n)(\mathbf{B}_n\odot\mathbf{C}_n)^T\mathbf{u_i^n} = \frac{1}{\sigma_i^2}(\mathbf{B}\odot\mathbf{C})(\mathbf{B}\odot\mathbf{C})^T\mathbf{z_i} \quad \forall i=1,\dots R^*.$$

Which means that in particular $\mathbf{z}_i=\mathbf{w}_i$ for $i=1,\dots R^*$. By this result, (3.15), the minimization property (3.17), (3.16) and $R^*\leq\overline{R}$, we find that

$$N = \lim_{n\to\infty}\sum_{i=1}^{R_n}\langle\mathbf{u_i^n},\mathbf{M}\mathbf{u_i^n}\rangle = \sum_{i=1}^{R^*}\langle\mathbf{w_i},\mathbf{M}\mathbf{w_i}\rangle \leq \sum_{i=1}^{\overline{R}}\langle\mathbf{w_i},\mathbf{M}\mathbf{w_i}\rangle \leq N,$$

where the last inequality follows from the definition of N as the supremum. Hence, equality has to hold in the previous formula, $R^*=\overline{R}$, and the vectors $(\mathbf{w_1},\dots\mathbf{w}_{R^*})$ are maximizers of (3.13). By Theorem 3.8 the corresponding matrices $(\mathbf{B},\mathbf{C})$ are minimizers of $\mathfrak{J}_{red}$. □

Remark 3.10
Note that by compactness we can always find a subsequence of any minimizing sequence for which $\frac{\mathbf{B}_n}{\|\mathbf{B}_n\|}$ and $\frac{\mathbf{C}_n}{\|\mathbf{C}_n\|}$ converge.

It follows from the proof that the strict inequality in (3.14), i.e., $R^*<\overline{R}$, cannot happen. This could also be deduced from the fact that the rank is a lower semicontinuous function.

Converse to these propositions is the following corollary for the case of non-existing minima.

Corollary 3.11
Suppose (1.2) does not have a minimum. Then there exists a minimizing sequence $(\mathbf{A}_n, \mathbf{B}_n, \mathbf{C}_n)$ where $\mathbf{B}_n$ and $\mathbf{C}_n$ converge such that

$$\liminf_{n\to\infty} \text{rank} \mathbf{B}_n \odot \mathbf{C}_n > \text{rank} \lim_{n\to\infty} \mathbf{B}_n \odot \mathbf{C}_n. \tag{3.19}$$

Proof
The assumption imply that a minimum of $\mathfrak{J}_{red}$ does not exist either. By the invariance with respect to the Khatri-Rao range, a normalization of a minimizing sequence $\mathbf{B}_n, \mathbf{C}_n$ remains a minimizing sequence. By compactness a converging subsequence exists. If (3.19) does not hold, then Proposition 3.9 implies an existence of a minimizer in (1.2) which contradicts the initial assumption. □

In the setting of the previous corollary, at least one singular value of $(\mathbf{B}_n \odot \mathbf{C}_n)$ tends to 0. It follows that the pseudo-inverse $\mathbf{G}^\dagger$ becomes unbounded, and thus, the norm of $\tilde{\mathbf{A}}_n[\mathbf{B}_n, \mathbf{C}_n]$ may become unbounded. This reflects the well-known fact of diverging summands in the case of non-existing minima; see the examples on the degenerate CP cases [15, 11, 16].

3.2. *Reduced functional in projection form*

We now derive an alternative form of the reduced functional $\mathfrak{J}_{red}$ as a weighted distance to the Khatri-Rao range. This form will be useful in the next section to design a simple algorithm for finding suboptimal solution or initial guesses to the optimization algorithm for (1.2).

Based on Lemma 3.7 we can simplify the reduced functional taking into account the diagonalization of $\mathbf{M}$. A reformulation of Lemma 3.7 in terms of the eigenvalue decomposition of $\mathbf{M}$ yields:

Lemma 3.12
Using the notation of Lemma 3.7 and Definition 3.3, we have

$$\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) = \sum_{i=1}^{JK} \overline{\lambda}_i \left(\|\overline{\mathbf{v}}_\mathbf{i}\|^2 - \sum_{r=1}^{\bar{R}} \langle \overline{\mathbf{v}}_\mathbf{i}, \mathbf{u_r} \rangle^2 \right) = \sum_{i=1}^{JK} \overline{\lambda}_i \left(1 - \sum_{r=1}^{\bar{R}} \langle \overline{\mathbf{v}}_\mathbf{i}, \mathbf{u_r} \rangle^2 \right). \tag{3.20}$$

Now we define the Khatri-Rao range, i.e., the range of the matrix $\mathbf{B} \odot \mathbf{C}$. This range is a subset of $\mathbb{R}^{IJ}$; for later use it is convenient to define the Khatri-Rao range by matricizing this range. As usual we denote the columns of the matrices $\mathbf{B}$ and $\mathbf{C}$ by $\mathbf{b_i}$ and $\mathbf{c_i}$:

Definition 3.13
Let us define the **Khatri-Rao range** as

$$\text{KR}(\mathbf{B}, \mathbf{C}) := \left\{ \mathbf{X} = \sum_{i=1}^{R} \mu_i \mathbf{b_i} \circ \mathbf{c_i} \in \mathbb{R}^{J\times K} \mid \; where \;\; \mu_i \in \mathbb{R}, \right\} \tag{3.21}$$

It is obvious that $\mathbf{X} \in \mathbb{R}^{I\times J}$ is in the Khatri-Rao range $\mathbf{X} \in \text{KR}(\mathbf{B}, \mathbf{C})$ if and only if its vectorized version $\mathbf{X^{vec}} \in \mathbb{R}^{IK}$ is in the range of $\mathbf{B} \odot \mathbf{C}$:

$$\mathbf{X^{vec}} = (\mathbf{B} \odot \mathbf{C}) \, \boldsymbol{\mu}.$$

We can rephrase the reduced functional in projection form as follows.

Theorem 3.14
Using the notation of Definition 3.3 we have

$$\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) = \sum_{i=1}^{JK} \overline{\lambda}_i \left(\|\overline{\mathbf{V}}_\mathbf{i} - \text{KR}(\mathbf{B}, \mathbf{C})\|_F \right)^2, \tag{3.22}$$

where $\|\overline{\mathbf{V}}_\mathbf{i} - \text{KR}(\mathbf{B}, \mathbf{C})\|_F$ denotes the distance of $\overline{\mathbf{V}}_\mathbf{i}$ to the Khatri-Rao range $\text{KR}(\mathbf{B}, \mathbf{C})$

$$\|\overline{\mathbf{V}}_\mathbf{i} - \text{KR}(\mathbf{B}, \mathbf{C})\|_F = \inf_{\mathbf{X} \in \text{KR}(\mathbf{B}, \mathbf{C})} \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F.$$

Proof
Let $\mathbf{P}_{\mathbf{B}\odot\mathbf{C}}$ be the orthogonal projector onto range$(\mathbf{B}\odot\mathbf{C})$, defined in (3.12). From the well-known fact about orthogonal projections,

$$\|\bar{\mathbf{v}}_\mathbf{i}\|^2 - \|\mathbf{P}_{\mathbf{B}\odot\mathbf{C}}\bar{\mathbf{v}}_\mathbf{i}\|^2 = \|\bar{\mathbf{v}}_\mathbf{i} - \mathbf{P}_{\mathbf{B}\odot\mathbf{C}}\bar{\mathbf{v}}_\mathbf{i}\|^2 = \inf_{\mathbf{X}\in\mathrm{range}(\mathbf{B}\odot\mathbf{C})} \|\bar{\mathbf{v}}_\mathbf{i} - \mathbf{X}\|^2,$$

Lemma 3.12 and a matricization, the result follows. □

A simple consequence of the previous theorem is the following.

Corollary 3.15
If $\bar{\mathbf{B}}, \bar{\mathbf{C}}$ and $\mathbf{B}, \mathbf{C}$ are matrices that span the same Khatri-Rao range,

$$\mathrm{KR}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathrm{KR}(\mathbf{B}, \mathbf{C}),$$

then

$$\mathfrak{J}_{red}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}).$$

Remark 3.16
If this corollary is applied to the case when $\min \mathfrak{J} = 0$, we obtain – as a special case – a uniqueness condition. The CP decomposition $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ is called unique up to permutation and scaling if any alternative decomposition $(\bar{\mathbf{A}}, \bar{\mathbf{B}}, \bar{\mathbf{C}})$ satisfies $\bar{\mathbf{A}} = \mathbf{A}\cdot\mathbf{\Pi}\cdot\mathbf{\Lambda}_1$, $\bar{\mathbf{B}} = \mathbf{B}\cdot\mathbf{\Pi}\cdot\mathbf{\Lambda}_2$ and $\bar{\mathbf{C}} = \mathbf{C}\cdot\mathbf{\Pi}\cdot\mathbf{\Lambda}_3$ where $\mathbf{\Pi}$ is an $R\times R$ permutation matrix and $\mathbf{\Lambda}_j$ are nonsingular matrices such that $\prod_{j=1}^n \mathbf{\Lambda}_j = \mathbf{I}_R$. Certainly, if $\bar{\mathbf{B}} = \mathbf{B}\cdot\mathbf{\Pi}\cdot\mathbf{\Lambda}_2$ and $\bar{\mathbf{C}} = \mathbf{C}\cdot\mathbf{\Pi}\cdot\mathbf{\Lambda}_3$, then $\mathrm{KR}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathrm{KR}(\mathbf{B}, \mathbf{C})$ and thus, $\mathfrak{J}_{red}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C})$. From Corollary 3.15 we find that if a CP decomposition is unique up to scaling and permutation then $\mathrm{KR}(\bar{\mathbf{B}}, \bar{\mathbf{C}}) = \mathrm{KR}(\mathbf{B}, \mathbf{C})$ can only hold when $\bar{\mathbf{B}}$ and $\bar{\mathbf{C}}$ is a scaled and permuted version of $\mathbf{B}$ and $\mathbf{C}$. The remaining matrix $\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}]$ is uniquely defined if and only if $\mathbf{B}\odot\mathbf{C}$ has full rank. Thus, it is possible to rephrase a uniqueness condition purely in terms of $\mathbf{B}, \mathbf{C}$: the CP decomposition $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ is unique up to permutation and scaling if and only if $\mathbf{B}\odot\mathbf{C}$ has full rank and the Khatri-Rao range $\mathrm{KR}(\mathbf{B}, \mathbf{C})$ uniquely determines $\mathbf{B}, \mathbf{C}$ up to scaling and permutation.

Remark 3.17
With minor modifications, Theorem 3.8 and Theorem 3.14 remain valid in the case of the CP-decomposition of higher order tensors. For instance, for fourth order tensors

$$(\mathcal{A})_{ijkl} = \sum_{r=1}^R a_{ir}b_{jr}c_{kr}d_{lr},$$

the Khatri-Rao range $\mathrm{KR}(\mathbf{B}, \mathbf{C})$ there has to be replaced by the analogous set

$$\mathrm{KR}(\mathbf{B}, \mathbf{C}, \mathbf{D}) = \left\{ \mathcal{X} = \sum_{i=1}^R \mu_i \mathbf{b_i}\circ\mathbf{c_i}\circ\mathbf{d_i} \in \mathbb{R}^{J\times K\times L} \mid \ where\ \ \mu_i \in \mathbb{R}, \right\}.$$

3.3. *Some special cases and optimality condition*

We can deduce some (partly well-known) result from the previous analysis. A particular simple case is the rank-1 approximation problem, i.e., $R = 1$ in (1.2). In this case, the rank of $\mathbf{B}\odot\mathbf{C}$ is always one (ignoring the trivial case of zero matrices). From Proposition 3.9 it follows that the rank condition is satisfied, hence a minimizer always exists. (Again, this is a well-known fact). Since an orthonormal basis of the Khatri-Rao range can be obtained by $\mathbf{b}\otimes\mathbf{c}$ with the vectors normalized to 1, we find that the rank-1 minimization problem is equivalent to the maximization problem [17]:

$$\max_{\|\mathbf{b}\|=1, \|\mathbf{c}\|=1} \langle \mathbf{b}\otimes\mathbf{c}, \mathbf{M}(\mathbf{b}\otimes\mathbf{c})\rangle. \tag{3.23}$$

Remark 3.18
Using Morse-theory it can be shown that generically there are always additional stationary points for the optimization problem (3.23) besides the global maxima and minima. It is easy to see that the ALS-algorithm saturates at such points, which means that the ALS algorithm is not guaranteed to converge to a global solution of (1.2). Local convergence of the ALS method has been shown in [18].

Theorem 3.8 can be used to find an equivalent functional also in the case $R = 2$. Here we have that

$$\mathbf{B} \odot \mathbf{C} = [\mathbf{b}_1 \otimes \mathbf{c}_1 \, \mathbf{b}_2 \otimes \mathbf{c}_2].$$

Without loss of generality, we can assume that all the column vectors $\mathbf{b}_i$, $\mathbf{c}_i$ are normalized to 1, since a column-wise scaling does not change the Khatri-Rao range and hence also not the functional by Corollary 3.15. If the rank of $\mathbf{B} \odot \mathbf{C}$ is 2, an orthogonal basis of $\mathbf{B} \odot \mathbf{C}$ can be obtained by a Gram-Schmidt procedure yielding

$$\begin{aligned} \mathbf{u}_1 &= \mathbf{b}_1 \otimes \mathbf{c}_1 \\ \mathbf{u}_2 &= \frac{1}{\sqrt{1-\gamma^2}} \left(\mathbf{b}_2 \otimes \mathbf{c}_2 - \gamma \mathbf{b}_1 \otimes \mathbf{c}_1 \right) \\ \gamma &= \langle \mathbf{b}_1, \mathbf{b}_2 \rangle \langle \mathbf{c}_1, \mathbf{c}_2 \rangle \\ 1 &= \|\mathbf{b}_i\| = \|\mathbf{c}_i\| \quad i = 1, 2. \end{aligned}$$

The condition $\operatorname{rank}(\mathbf{B} \odot \mathbf{C}) = 2$ is equivalent to $|\gamma| < 1$. This yields the following equivalent functional

$$\begin{aligned} K(\mathbf{b}_1, \mathbf{b}_2, \mathbf{c}_1, \mathbf{c}_2) &= \frac{1}{1-\gamma^2} \left(\langle \mathbf{b}_1 \otimes \mathbf{c}_1, \mathbf{M}(\mathbf{b}_1 \otimes \mathbf{c}_1) \rangle + \langle \mathbf{b}_2 \otimes \mathbf{c}_2, \mathbf{M}(\mathbf{b}_2 \otimes \mathbf{c}_2) \rangle \right. \\ &\qquad \left. -2\gamma \langle \mathbf{b}_1 \otimes \mathbf{c}_1, \mathbf{M}(\mathbf{b}_2 \otimes \mathbf{c}_2) \rangle \rangle \right), \qquad (3.24) \\ \gamma &= \langle \mathbf{b}_1, \mathbf{b}_2 \rangle \langle \mathbf{c}_1, \mathbf{c}_2 \rangle). \qquad (3.25) \end{aligned}$$

In the case that $\gamma = 1$, the matrix $\mathbf{B} \odot \mathbf{C}$ is rank deficient (i.e., it has rank 1, ignoring the trivial case of zero rank), hence its range is spanned by one of the column vectors. The corresponding functional is, however, then identical to the $R = 1$ case:

Proposition 3.19
In the case $R = 2$, the optimization problem (3.13) is equivalent to the problem

$$\max \begin{cases} \sup_{\|\mathbf{b}_1\|, \|\mathbf{b}_2\|, \|\mathbf{c}_1\|, \|\mathbf{c}_2\| = 1} K(\mathbf{b}_1, \mathbf{b}_2, \mathbf{c}_1, \mathbf{c}_2) & |(\langle \mathbf{b}_1, \mathbf{b}_2 \rangle \langle \mathbf{c}_1, \mathbf{c}_2 \rangle)| < 1 \\ \max_{\|\mathbf{b}\|=1, \|\mathbf{c}\|=1} \langle \mathbf{b} \otimes \mathbf{c}, \mathbf{M}(\mathbf{b} \otimes \mathbf{c}) \rangle & \text{otherwise} \end{cases}$$

A similar result can be derived for higher ranks, however the equations quickly become more complicated than the formulas above.

Additionally, we now state an optimality condition for the reduced functional for the case that a minimizer exists and its Khatri-Rao product has full rank.

Proposition 3.20
Let $\mathbf{B}, \mathbf{C}$ be a minimizer of $\mathfrak{J}_{red}$ (and thus $(\tilde{\mathbf{A}}[\mathbf{B}, \mathbf{C}], \mathbf{B}, \mathbf{C})$ minimizers of (1.2)), and assume that $\mathbf{B} \odot \mathbf{C}$ has full rank R. Then $\mathbf{B}, \mathbf{C}$ satisfy

$$\langle (\mathbf{I} - \mathbf{P}_{\mathbf{B} \odot \mathbf{C}}) \cdot \mathbf{M} \cdot (\mathbf{B} \odot \mathbf{C}) \cdot \mathbf{G}^\dagger, \delta\mathbf{B} \odot \mathbf{C} + \mathbf{B} \odot \delta\mathbf{C} \rangle = 0 \quad \forall \delta\mathbf{B}, \delta\mathbf{C}. \qquad (3.26)$$

Proof
Under the given assumptions we have that $\mathbf{G}$ is invertible, and hence differentiable with respect $\mathbf{B}, \mathbf{C}$. Using (3.10) with

$$\mathbf{P}_{\mathbf{B} \odot \mathbf{C}} = (\mathbf{B} \odot \mathbf{C}) \cdot ((\mathbf{B} \odot \mathbf{C})^T \cdot \mathbf{B} \odot \mathbf{C})^{-1} \cdot (\mathbf{B} \odot \mathbf{C})^T,$$

a tedious but straightforward calculation using the formula for the derivative of inverses $dA^{-1} = -A^{-1} \cdot dA \cdot A^{-1}$, yields that

$$\begin{aligned}&\mathfrak{J}_{red}(\mathbf{B}+\delta\mathbf{B},\mathbf{C}+\delta\mathbf{C}) - \mathfrak{J}_{red}(\mathbf{B},\mathbf{C})\\ &\quad = 2\langle(\mathbf{I}-\mathbf{P}_{\mathbf{B}\odot\mathbf{C}})\cdot\mathbf{M}\cdot(\mathbf{B}\odot\mathbf{C})\cdot\mathbf{G}^{\dagger}, (\mathbf{B}+\delta\mathbf{B})\odot(\mathbf{C}+\delta\mathbf{C}) - \mathbf{B}\odot\mathbf{C}\rangle + O(\|\delta\mathbf{B},\delta\mathbf{C}\|^2)\end{aligned}$$

Expanding this up to linear order yields the directional derivative, which has to vanish at an extremal point. □

Note that $(\mathbf{I}-\mathbf{P}_{\mathbf{B}\odot\mathbf{C}})$ is the orthogonal projector onto the orthogonal complement of the range (which equals the nullspace of $(\mathbf{B}\odot\mathbf{C})^T$). Since we have

$$\begin{aligned}&\langle(\mathbf{I}-\mathbf{P}_{\mathbf{B}\odot\mathbf{C}})\cdot\mathbf{M}\cdot(\mathbf{B}\odot\mathbf{C})\cdot\mathbf{G}^{\dagger}, \delta\mathbf{B}\odot\mathbf{C}+\mathbf{B}\odot\delta\mathbf{C}\rangle\\ &\quad = \langle\mathbf{M}\cdot(\mathbf{B}\odot\mathbf{C})\cdot\mathbf{G}^{\dagger}, (\mathbf{I}-\mathbf{P}_{\mathbf{B}\odot\mathbf{C}})\cdot(\delta\mathbf{B}\odot\mathbf{C}+\mathbf{B}\odot\delta\mathbf{C})\rangle\end{aligned}$$

it can be seen that only those perturbation $\delta\mathbf{B},\delta\mathbf{C}$ contribute to a change of the functional (in leading order), which have that $\delta\mathbf{B}\odot\mathbf{C}$ and $\mathbf{B}\odot\delta\mathbf{C}$ are not in the range of $\mathbf{B}\odot\mathbf{C}$. This again reflect the invariance of the reduced objective functional with respect to the Khatri-Rao range (Corollary 3.15). It might also partially explain the observed bad convergence properties of many minimization algorithms, the so called swamping [19, 15, 20, 21].

Remark 3.21
The reduced functional has been already considered in [10]. Also, some of the results provided are, of course, well-known, namely Corollary 3.15 as a consequences of (3.9), Theorem 3.8 (for $R = 1$) and Theorem 3.14 (for $R = 1$).

Tendeiro et al. [10] used Lagrange multipliers for the analysis of the reduced objective functional while our approach avoids Lagrange multipliers and eliminates the occurrence of $\mathbf{A}$ directly to derive the functional $\mathfrak{J}_{red}$.

The advantage of our approach is that we obtain an optimization problem over compact sets due to the scaling invariance of the Khatri-Rao range: as it was done before, we can always consider the optimization problem being performed over matrices with columns having norm one. This set of matrices with normalized columns is closed and bounded, i.e., a compact set. The price to pay is the possibility of a discontinuous $\mathfrak{J}_{red}$ at some points where the rank of $\mathbf{B}\odot\mathbf{C}$ decreases. Another advantage is that we can work with the symmetric tensor $\mathcal{M}$ (or the self-adjoint positive semidefinite matrix $\mathbf{M}$). This is useful from a theoretical as well as a numerical point of view since the structure of the eigenvalue decomposition is well-known and its numerical computation is stable.

4. APPROXIMATE AND EXACT LEAST-SQUARES MINIMIZATION

In this section we set the stage for several computational schemes using the results of the previous sections with the aim to compute either suboptimal or exact minimizers for special cases.

At first we study a method to obtain suboptimal (i.e., approximate) minimizers for the general least-squares problem (1.2). This is done by finding a new functional which serves as an upper bound for $\mathfrak{J}_{red}$. The new functional has minimizers that are easily computable. The Centroid Projection yields a suboptimal solution which can be used as initial guess to further minimization algorithms for $\mathfrak{J}$. Moreover, we also find useful a-posteriori error estimates.

4.1. Bounds on $\mathfrak{J}_{red}$ and suboptimal solutions

Here we prove lower and upper bound on $\mathfrak{J}_{red}$ using Theorem 3.14. Moreover, we will also define a majorizing functional $\mathfrak{L}$, whose minimizers can be calculated by standard linear algebra methods.

In Theorem 3.14 we use the Eckart-Young theorem to obtain lower bounds. Recall that $\mathbf{M} = \overline{\mathbf{V}}\cdot\bar{\mathbf{S}}\cdot\overline{\mathbf{V}}^T$.

Corollary 4.1
For all matrices $(\mathbf{B}, \mathbf{C})$, a lower bound of $\mathfrak{J}_{red}$ is calculated as

$$\inf_{(\mathbf{B},\mathbf{C})} \mathfrak{J}_{red}(\mathbf{B},\mathbf{C}) \geq \sum_{i=1}^{JK} \overline{\lambda}_i \left(\sum_{k=R+1}^{\min(J,K)} (\overline{\sigma}_k^i)^2 \right) \tag{4.1}$$

with $\overline{\sigma}_k^i$ being the k-th singular values of $\overline{\mathbf{V}}_\mathbf{i}$.

Proof
The Eckart-Young Theorem gives the infimum by the truncated SVD, i.e.,

$$\inf_{rank(\mathbf{X})=R} \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F = \left(\sum_{k=R+1}^{\min\{J,K\}} (\overline{\sigma}_k^i)^2 \right)^{\frac{1}{2}}.$$

Also, observe that $\mathrm{KR}(\mathbf{B}, \mathbf{C})$ contains matrices with rank at most R, hence, by Theorem 3.14

$$\begin{aligned} \inf_{(\mathbf{B},\mathbf{C})} \mathfrak{J}_{red}(\mathbf{B},\mathbf{C}) &\geq \sum_{i=1}^{JK} \overline{\lambda}_i \inf_{\mathbf{X}\in\mathrm{KR}(\mathbf{B},\mathbf{C})} \left(\|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F\right)^2 \geq \sum_{i=1}^{JK} \overline{\lambda}_i \inf_{\text{rank}(\mathbf{X})\leq R} \left(\|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F\right)^2 \\ &\geq \sum_{i=1}^{JK} \overline{\lambda}_i \left(\sum_{k=R+1}^{\min(J,K)} (\overline{\sigma}_k^i)^2 \right). \end{aligned}$$

□

This corollary can be used to find lower bounds on the distance of a tensor to its best rank-R approximation. In particular, if a tensor has rank R, it must hold that

$$\sum_{i=1}^{JK} \overline{\lambda}_i \left(\sum_{k=R+1}^{\min(J,K)} (\overline{\sigma}_k^i)^2 \right) = 0.$$

Note that this a-priori lower bound can be calculated by the standard EVD of $\mathbf{M}$ followed by an SVD of each of the $\overline{\mathbf{V}}_\mathbf{i}$.

The next result establishes an upper bound using a majorizing functional.

Lemma 4.2
Define the functional

$$\mathfrak{L}(\mathbf{B},\mathbf{C}) := \inf_{\mathbf{X}\in\mathrm{KR}(\mathbf{B},\mathbf{C})} \sum_{i=1}^{JK} \overline{\lambda}_i \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F^2, \tag{4.2}$$

then we have that

$$\inf_{\mathbf{B},\mathbf{C}} \mathfrak{J}_{red}(\mathbf{B},\mathbf{C}) \leq \inf_{\mathbf{B},\mathbf{C}} \mathfrak{L}(\mathbf{B},\mathbf{C}) = \inf_{\text{rank}(\mathbf{X})\leq R} \sum_{i=1}^{JK} \overline{\lambda}_i \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F^2. \tag{4.3}$$

Proof

$$\begin{aligned} \inf_{(\mathbf{B},\mathbf{C})} \mathfrak{J}_{red}(\mathbf{B},\mathbf{C}) &= \inf_{(\mathbf{B},\mathbf{C})} \sum_{i=1}^{JK} \overline{\lambda}_i \inf_{\mathbf{X}\in\mathrm{KR}(\mathbf{B},\mathbf{C})} \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F^2 \\ &\leq \inf_{(\mathbf{B},\mathbf{C})} \inf_{\mathbf{X}\in\mathrm{KR}(\mathbf{B},\mathbf{C})} \sum_{i=1}^{JK} \overline{\lambda}_i \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F^2 = \inf_{\text{rank}(\mathbf{X})\leq R} \sum_{i=1}^{JK} \overline{\lambda}_i \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F^2. \end{aligned}$$

The last equality follows from the fact that $\mathrm{KR}(\mathbf{B}, \mathbf{C})$ contains matrices with rank $\leq R$. Moreover, for any matrix $\mathbf{X}$ of rank at most R, it follows that $\mathbf{X} \in \mathrm{KR}(\mathbf{B}, \mathbf{C})$ where $\mathbf{B}$, $\mathbf{C}$ are formed by the columns of the orthogonal matrices in the SVD of $\mathbf{X}$. □

In contrast to $\mathfrak{J}_{red}$, calculating a minimizer of $\mathfrak{L}$ can be done easily. First, define the **centroid matrix** of $\overline{\mathbf{V}}_\mathbf{i}$.

$$\bar{\mathbf{V}}^C := \sum_{i=1}^{JK} \xi_i \overline{\mathbf{V}}_\mathbf{i} \qquad \text{where} \qquad \xi_i = \frac{\bar{\lambda}_i}{\sum_{j=1}^{JK} \bar{\lambda}_j}, \quad i = 1, \dots JK. \tag{4.4}$$

Theorem 4.3
Let $\mathbf{y_k}$, $\mathbf{z_k}$ be the left and right singular vectors in the SVD of $\bar{\mathbf{V}}^C$ defined in (4.4) with $\sigma_k(\bar{\mathbf{V}}^C)$ the associated singular values in descending order. Then,

$$\mathbf{B_C} = [\mathbf{y_1} \dots \mathbf{y_R}] \quad \mathbf{C_C} = [\mathbf{z_1} \dots \mathbf{z_R}]$$

is a minimizer of $\mathfrak{L}(\mathbf{B}, \mathbf{C})$. Moreover,

$$\inf_{\mathbf{B},\mathbf{C}} \mathfrak{L}(\mathbf{B}, \mathbf{C}) = \mathfrak{L}(\mathbf{B_C}, \mathbf{C_C}) = \left[(1 - \|\bar{\mathbf{V}}^C\|_F^2) \left(\sum_{i=1}^{JK} \bar{\lambda}_i \right) + \left(\sum_{i=1}^{JK} \bar{\lambda}_i \right) \left(\sum_{k=R+1}^{\min\{J,K\}} \sigma_k(\bar{\mathbf{V}}^C)^2 \right) \right]. \tag{4.5}$$

Proof
Expanding the square using $\|\overline{\mathbf{V}}_\mathbf{i}\|_F^2 = 1$ yields

$$\begin{aligned} \sum_{i=1}^{JK} \bar{\lambda}_i \|\overline{\mathbf{V}}_\mathbf{i} - \mathbf{X}\|_F^2 &= \left(\sum_{i=1}^{JK} \bar{\lambda}_i \|\overline{\mathbf{V}}_\mathbf{i}\|_F^2 \right) - 2 \left\langle \sum_{i=1}^{JK} \bar{\lambda}_i \overline{\mathbf{V}}_\mathbf{i}, \mathbf{X} \right\rangle + \left(\sum_{i=1}^{JK} \bar{\lambda}_i \right) \langle \mathbf{X}, \mathbf{X} \rangle \\ &= \left(\sum_{i=1}^{JK} \bar{\lambda}_i \right) (1 - 2\langle \bar{\mathbf{V}}^C, \mathbf{X} \rangle + \langle \mathbf{X}, \mathbf{X} \rangle) \\ &= \left(\sum_{i=1}^{JK} \bar{\lambda}_i \right) (1 - \|\bar{\mathbf{V}}^C\|_F^2) + \left(\sum_{i=1}^{JK} \bar{\lambda}_i \right) \|\bar{\mathbf{V}}^C - \mathbf{X}\|^2. \end{aligned}$$

Using the Eckart-Young Theorem, we see that a minimizer $\mathbf{X}$ is found through the truncated SVD of $\bar{\mathbf{V}}^C$ and (4.5) is obtained. □

Computing minimizers of $\mathfrak{L}$ in Theorem 4.3 yields matrices $\mathbf{B_C}$ and $\mathbf{C_C}$ (and $\mathbf{A_C}$ via (3.3)) which in turn approximate the minimizers of $\mathfrak{J}_{red}$. The corresponding algorithm, the **Centroid Projection (CePr)**, is sketched in Algorithm I in the next section.

The Centroid Projection also provides computable bounds for the optimal value of the least squares functional: combining Corollary 4.1 and Theorem 4.3 yields the following a-posteriori bounds on the quality of the output of the Centroid Projection Algorithm.

Corollary 4.4
Let $\mathbf{B_C}$ and $\mathbf{C_C}$ be computed by the Centroid Projection as in Theorem 4.3. Then with ξ_i as in (4.4),

$$|\mathfrak{J}_{red}(\mathbf{B_C}, \mathbf{C_C}) - \inf_{(\mathbf{B},\mathbf{C})} \mathfrak{J}_{red}(\mathbf{B}, \mathbf{C})| \leq \|\mathcal{T}\|_F^2 \left(\sum_{i=1}^{JK} \xi_i \left(\sum_{k=1}^{R} (\overline{\sigma}_k^i)^2 \right) - \sum_{k=1}^{R} \sigma_k(\bar{\mathbf{V}}^C)^2. \right) \tag{4.6}$$

Proof
This is a combination of Theorem 4.3 and Lemma 4.2. Note that $\sum_{i=1}^{JK} \bar{\lambda}_i = \mathrm{Tr}(\mathbf{M}) = \|\mathcal{T}\|^2$. □

Remark 4.5
Since $\bar{\mathbf{V}}^C = \sum \xi_i \overline{\mathbf{V}}_\mathbf{i}$, the positivity of the right hand side in this estimate is a consequence of the convexity of the sum of the squares of the largest singular values (the Schatten norm). We notice that the right-hand side is an a-posteriori bound on the quality of the suboptimal solutions $\mathbf{B_C}, \mathbf{C_C}$ that is easy to compute because $\sigma_k(\bar{\mathbf{V}}^C)$ is known (`Sc` in step 3 of Algorithm I).

The Centroid Projection yields a "good" (but not necessarily optimal) solution to the general least squares problem 1.2. It is a simple method and can, for instance, be used to find starting values for more advanced algorithm such as ALS. At times the standard methods (for example ALS) for CP are initialized with random guesses, but this often leads to a slowed convergence rate. More commonly, CP algorithms are started with initial guesses $(\mathbf{A}_0, \mathbf{B}_0, \mathbf{C}_0)$ that are obtained from the SVD of the matricized tensor in all modes (e.g. in HOSVD). The initialization methods of [22, 23, 24] are based on the generalized eigenvalue problem (see Van Loan's GEVD [25]) through a *slab-wise* representation of the Khatri-Rao products. The direct methods, Direct TriLinear Decomposition (DTLD) [22] and the Generalized Rank Annihilation Method (GRAM) [23], also give poor CP solutions, but provide good starters.

4.2. *Best Rank-1 Fit*

In this section we study the least squares functional (1.2) for the case $R = 1$, i.e., when we seek the best approximating rank-1 decomposition. Note that we do not assume that $\mathcal{T}$ is a rank-1 tensor itself. It turns out that we can use a similar idea as in the Centroid Projection method, but with different ξ_i in (4.4).

If $\mathcal{T}$ is a rank-1 tensor, then it is not difficult to compute its decomposition. In fact, the Centroid Projection algorithm of the previous section will do the job. This can easily be seen because then only one eigenvalue $\overline{\lambda}_i$ is different from zero. The matrix $\overline{\mathbf{V}}^C$ is identical to $\overline{\mathbf{V}}_{\mathbf{1}}$, $\xi_1 = 1$ and it is quite clear that the right hand side of (4.6) is zero.

The more challenging task is to compute the best approximating rank-1 decomposition for arbitrary tensors $\mathcal{T}$. In this case, of course, the least squares functional will not necessarily be zero at a minimum. Moreover, using Theorem 3.14 with normalized vectors (while preserving the Khatri-Rao range), the minimization problem for the reduced functional can be expressed as

$$\min_{\|\mathbf{b}\|=1, \|\mathbf{c}\|=1} \mathfrak{J}_{red}(\mathbf{b}, \mathbf{c}) = \min_{\|\mathbf{b}\|=1, \|\mathbf{c}\|=1, (\mu_1, \ldots \mu_{R_M}) \in \mathbb{R}^{R_M}} \sum_{i=1}^{R_M} \overline{\lambda}_i \|\overline{\mathbf{V}}_{\mathbf{i}} - \mu_i \mathbf{b} \otimes \mathbf{c}\|^2, \tag{4.7}$$

where R_M is the rank of $\mathbf{M}$. This yields the following lemma

Lemma 4.6
Solutions $(\mathbf{b}, \mathbf{c})$ of the minimization problem (4.7) satisfy

$$(\mathbf{b}, \mathbf{c}) \subset \left\{ (\mathbf{b}(\xi), \mathbf{c}(\xi)) \,|\, \xi_i \in \mathbb{R}^{R_M}, \sum_{i=1}^{R_M} \xi_i^2 = 1 \right\}, \tag{4.8}$$

where $(\mathbf{b}(\xi), \mathbf{c}(\xi)$ denotes the left and right singular vectors corresponding to a largest singular value of $\overline{\mathbf{V}}(\xi) := \sum_{i=1}^{R_M} \xi_i \overline{\mathbf{V}}_{\mathbf{i}}$. Moreover with $\boldsymbol{\mu} = (\mu_1, \ldots \mu_{R_M})$ it holds that

$$\begin{aligned} \min_{(\mu_1, \ldots \mu_{R_M}) \in \mathbb{R}^{R_M}} \tilde{\mathfrak{J}}(\boldsymbol{\mu}) &:= \sum_{i=1}^{JK} \overline{\lambda}_i (1 + \mu_i^2) - 2\sigma_{max}(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_{\mathbf{i}}) \\ &= \min_{\|\mathbf{b}\|=1, \|\mathbf{c}\|=1} \mathfrak{J}_{red}(\mathbf{b}, \mathbf{c}), \end{aligned} \tag{4.9}$$

where σ_{max} denotes the largest singular value.

Proof
We know that a solution $(\mathbf{b}, \mathbf{c})$ and $\boldsymbol{\mu} := (\mu_1, \ldots \mu_{R_M})$ to (4.7) exists. Keep $\boldsymbol{\mu}$ fixed. Using the normalizations of $(\mathbf{b}, \mathbf{c})$ and $\overline{\mathbf{V}}_{\mathbf{i}}$, solutions to (4.7) minimize also

$$\begin{aligned} \mathfrak{J}_{red}(\mathbf{b}, \mathbf{c}) &= \sum_{i=1}^{R_M} \overline{\lambda}_i \left(1 + \mu_i^2 - 2\langle \mu_i \overline{\mathbf{V}}_{\mathbf{i}}, \mathbf{b} \otimes \mathbf{c} \rangle\right) \\ &= \sum_{i=1}^{R_M} \overline{\lambda}_i (1 + \mu_i^2) + \|\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_{\mathbf{i}} - \mathbf{b} \otimes \mathbf{c}\|^2 - 1 - \sum_{i=1}^{R_M} \overline{\lambda}_i^2 \mu_i^2 \end{aligned}$$

over the set of normalized vectors. Thus, by the Eckart-Young theorem, $\mathbf{b} \otimes \mathbf{c}$ must be the rank-1 best approximation to $\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i}$, which shows the first statement. Note that the singular vectors are invariant under scaling of the underlying matrix, thus we can normalize the ξ_i as it is done in the lemma.

Taking such $\mathbf{b}, \mathbf{c}$ gives the equality

$$\begin{aligned} \|\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i} - \mathbf{b} \otimes \mathbf{c}\|^2 &= \sum_{k=2}^{R_M} \sigma_k(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i})^2 + (\sigma_{max}(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i}) - 1)^2 \\ &= \|\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i})\|^2 + 1 - 2\sigma_{max}(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i}). \end{aligned}$$

The orthogonality of $\overline{\mathbf{V}}_\mathbf{i}$ establishes the second assertion. □

By this lemma, the rank-1 problem (4.7) is equivalent to the R_M-dimensional optimization problem (4.9) for $\boldsymbol{\mu}$. Using (4.9) we can also find the necessary optimality conditions.

Theorem 4.7
Solutions $\mathbf{b}, \mathbf{c}$ of the minimization problem (4.7) satisfy (4.8), with

$$\xi_i = \frac{\overline{\lambda}_i \mu_i}{\sqrt{\sum_{j=1}^{R_M} \overline{\lambda}_j^2 \mu_j^2}}, \qquad i = 1, \ldots, R_M,$$

where $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_{R_M}) \in \mathbb{R}^{R_M}$ is a solution to the optimization problem (4.9). All such $\boldsymbol{\mu}$ satisfy

$$\mu_i = \langle \overline{\mathbf{V}}_\mathbf{i} \mathbf{c}(\boldsymbol{\mu}), \mathbf{b}(\boldsymbol{\mu}) \rangle, \qquad i = 1, \ldots, R_M, \tag{4.10}$$

where $\mathbf{c}(\boldsymbol{\mu}), \mathbf{b}(\boldsymbol{\mu})$ denote the left and right singular vectors corresponding to a $\sigma_{max}(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i})$.

Proof
First consider the case that the optimal $\boldsymbol{\mu}$ in (4.9) corresponds to a unique largest singular value. Then (4.10) is a consequence of the well-known formula for the derivative of eigenvalues of symmetric matrices. Note that the singular values are smooth functions in this case. We have with $\mathbf{K}(\boldsymbol{\mu}) = \sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i}$ and $\mathbf{c}(\boldsymbol{\mu}), \mathbf{b}(\boldsymbol{\mu})$ the singular vectors corresponding to the largest singular value

$$\begin{aligned} &\frac{\partial}{\partial \mu_j} \sigma_{max}(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i}) = \frac{\partial}{\partial \mu_j} \sqrt{\lambda_{max}(\mathbf{K}(\boldsymbol{\mu})^T \mathbf{K}(\boldsymbol{\mu}))} \\ &= \frac{1}{2\sigma_{max}(\mathbf{K}(\boldsymbol{\mu}))} \left(\langle \frac{\partial}{\partial \mu_j} \mathbf{K}(\boldsymbol{\mu})^T \mathbf{K}(\boldsymbol{\mu}) \mathbf{c}(\boldsymbol{\mu}), \mathbf{c}(\boldsymbol{\mu}) \rangle \right) \\ &= \frac{1}{\sigma_{max}(\mathbf{K}(\boldsymbol{\mu}))} \left\langle \mathbf{K}(\boldsymbol{\mu}) \mathbf{c}(\boldsymbol{\mu}), \frac{\partial}{\partial \mu_j} \mathbf{K}(\boldsymbol{\mu}) \mathbf{c}(\boldsymbol{\mu}) \right\rangle = \langle \mathbf{b}(\boldsymbol{\mu}), \overline{\lambda}_j \overline{\mathbf{V}}_\mathbf{j} \mathbf{c}(\boldsymbol{\mu}) \rangle, \end{aligned}$$

which proves the result in the simple case. In the case of multiple singular values, we notice that each branch corresponding to a different singular value is a smooth function of μ_i [13]. We then have with $\boldsymbol{\mu}_\epsilon = \boldsymbol{\mu} + \epsilon \mathbf{e}_j$ and $\mathbf{e}_j$ being the j-th unit vector that

$$\tilde{\mathfrak{J}}(\boldsymbol{\mu}_\epsilon) - \tilde{\mathfrak{J}}(\boldsymbol{\mu}) = 2\lambda_j \mu_j \epsilon + O(\epsilon^2) - 2 \max_{r=1,\ldots\zeta} \left[\epsilon \langle \overline{\mathbf{V}}_\mathbf{j} \mathbf{c}_\mathbf{r}(\boldsymbol{\mu}), \mathbf{b}_\mathbf{r}(\boldsymbol{\mu}) + O(\epsilon^2) \right] \geq 0,$$

where $\mathbf{c}_\mathbf{r}(\boldsymbol{\mu}), \mathbf{b}_\mathbf{r}(\boldsymbol{\mu})$ are the singular vectors associated to the multiple singular value $\sigma_{max}(\sum_{i=1}^{R_M} \overline{\lambda}_i \mu_i \overline{\mathbf{V}}_\mathbf{i})$. By a case distinction $\epsilon > 0, \epsilon < 0$ and letting $\epsilon \to 0$ we get

$$\begin{aligned} \mu_j - \max_{r=1,\ldots\zeta} \langle \overline{\mathbf{V}}_\mathbf{j} \mathbf{c}_\mathbf{r}(\boldsymbol{\mu}), \mathbf{b}_\mathbf{r}(\boldsymbol{\mu}) \rangle \geq 0 \\ \mu_j - \min_{r=1,\ldots\zeta} \langle \overline{\mathbf{V}}_\mathbf{j} \mathbf{c}_\mathbf{r}(\boldsymbol{\mu}), \mathbf{b}_\mathbf{r}(\boldsymbol{\mu}) \rangle \leq 0. \end{aligned}$$

However, this can only be the case if $\max$ and $\min$ are identical, which yields the result also in this case. □

Remark 4.8
The optimality condition (4.10) could have been obtained equally well using the Rayleigh quotient form (3.13), (3.23) comparable results were in fact obtained in [26]. However, by using the Rayleigh quotient, it is not clear a-priori that an optimal decomposition is obtained from the fixed points of (4.10) where $\mathbf{c}(\boldsymbol{\mu}), \mathbf{b}(\boldsymbol{\mu})$ correspond to the *largest* singular value. Starting from (3.23) one can only derive (4.10) where $\mathbf{c}(\boldsymbol{\mu}), \mathbf{b}(\boldsymbol{\mu})$ corresponding to *some* (not necessarily the largest) singular value. For this reason we used Theorem 3.14 here. For further algorithms and results for the rank-1 approximation, in particular for the case of an orthogonal decomposition, we refer to [26, 27].

This theorem is the basis of computational algorithms for finding optimal rank-1 decompositions, by solving the optimality condition (4.10) by a fixed-point iteration, see Algorithm II (FP-R1(local)) and its globalized variants, Algorithm III (FP-R1(RIG)) and Algorithm IV (FP-R1(APIG)).

For a slight improvement of our algorithms, we later need the following lemma, which yields an a-priori guess for $\boldsymbol{\mu}$.

Lemma 4.9
Let $\boldsymbol{\mu}_{opt}$ be a solution to minimization problem (4.9). Then it holds that

$$\sum_{i=1}^{R_M} \overline{\lambda}_i(\boldsymbol{\mu}_{opt})_i^2 - 2\overline{\lambda}_i|(\boldsymbol{\mu}_{opt})_i|\sigma_{max}(\overline{\mathbf{V}}_\mathbf{i}) \leq - \max_{k=1,\dots R_M} \overline{\lambda}_k \sigma_{max}(\overline{\mathbf{V}}_\mathbf{k})^2 \tag{4.11}$$

Proof
Note that σ_{max} is the spectral norm for matrices, thus the triangular inequality immediately gives that for any $\boldsymbol{\mu}$

$$\tilde{\mathfrak{J}}(\boldsymbol{\mu}) \geq \sum_{i=1}^{R_M} \overline{\lambda}_i(1+\mu_i^2) - 2\overline{\lambda}_i|\mu_i|\sigma_{max}(\overline{\mathbf{V}}_\mathbf{i})$$

On the other hand with $\boldsymbol{\mu} = \tau \mathbf{e}_k$, where $\mathbf{e}_k$ is the k-th unit vector and τ is a real parameter, we have

$$\tilde{\mathfrak{J}}(\boldsymbol{\mu}_{opt}) \leq \min_{\tau\in\mathbb{R}} \tilde{\mathfrak{J}}(\tau \mathbf{e}_k) = \sum_{i=1}^{R_M} \overline{\lambda}_i - \overline{\lambda}_k \sigma_{max}(\overline{\mathbf{V}}_\mathbf{k})^2.$$

Choosing that k that minimizes this upper bound yields the result. □

4.3. *Reconstruction of the tensor decomposition*

In this section we consider the case that there exists a tensor decomposition of rank R such that the infimum of the least squares functional (1.2) is zero. In other words, for a given tensor $\mathcal{T}$ we assume the existence of an unknown rank-R decomposition $\mathcal{T} = \sum_{r=1}^{R} \mathbf{a_r} \circ \mathbf{b_r} \circ \mathbf{c_r}$ and we want to find the corresponding factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$.

Equivalently, we can assume that with a fixed R

$$\inf_{\mathbf{B},\mathbf{C}} \mathfrak{J}_{red}(\mathbf{B},\mathbf{C}) = \min_{\mathbf{B},\mathbf{C}} \mathfrak{J}_{red}(\mathbf{B},\mathbf{C}) = 0. \tag{4.12}$$

We have the following result:

Lemma 4.10
Assume (4.12) with $(\mathbf{B},\mathbf{C})$ as minimizers. Let R_M be the rank of $\mathbf{M}$, then there exists vectors $\eta_1, \dots \eta_{R_M}$ such that

$$\overline{\mathbf{v}}_\mathbf{i} = (\mathbf{B} \odot \mathbf{C})\eta_i, \quad i = 1 \dots R_M, \tag{4.13}$$

moreover, $R \geq \operatorname{rank}(\mathbf{B} \odot \mathbf{C}) \geq R_M$.

If $R = R_M$, then the set of matrices $\overline{\mathbf{V}}_\mathbf{i}$ are an orthogonal basis of the Khatri-Rao range $\mathrm{KR}(\mathbf{B}, \mathbf{C})$. Moreover, in this case there exists a matrix $\mathbf{Z} \in \mathbb{R}^{R_M \times R_M}$ such that with $(\mathbf{Z})_{i,r} = z_{i,r}$

$$\sum_{i=1}^{R_M} z_{i,r} \overline{\mathbf{V}}_\mathbf{i} = \mathbf{b_r} \otimes \mathbf{c_r}, \quad r = 1, \ldots R_M. \tag{4.14}$$

Proof
The first part follows immediately from Theorem 3.14. If $R = \mathrm{rank}(\mathbf{B} \odot \mathbf{C}) = R_M$, then there exists a matrix $\mathbf{Y} \in \mathbb{R}^{R_M \times R_M}$ such that

$$(\overline{\mathbf{v}}_\mathbf{1} \ \ldots \ \overline{\mathbf{v}}_\mathbf{R_M}) = (\mathbf{B} \odot \mathbf{C})\mathbf{Y},$$

and $\mathbf{Y}$ must have full rank. With $\mathbf{Z} = \mathbf{Y}^{-1}$, (4.14) follows. □

Numerical experiments indicate that in many practical situations $R = R_M$ holds true. Thus, if (4.12) and the rank condition $R = R_M$ apply, the task of reconstructing the tensor decomposition reduces to finding the matrix $\mathbf{Z}$ in (4.14). This can be done by similar methods as in the previous section.

Theorem 4.11
Let (4.12) and $R = R_M$ hold, then the column vectors $\mathbf{b_r}, \mathbf{c_r}$ of the minimizers $\mathbf{B}, \mathbf{C}$ in (4.12) are the left and right singular vectors corresponding to the largest singular value of the matrix $\sum_{i=1}^{R_M} z_{i,r} \overline{\mathbf{V}}_\mathbf{i}$, where each $\mathbf{z_r} := (z_{i,r})_{i=1}^{R_M}$ is a solution to the optimization problem

$$\max_{\|z\|=1} \sigma_{max}(\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i}) \tag{4.15}$$

with $\sigma_{max}(\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i}) = 1$.

Proof
From the previous lemma it is clear that there exists R_M vectors $\mathbf{z_r}$, $r = 1, \ldots R_M$, such that (4.14) holds. By orthogonality, we have for arbitrary $\mathbf{z}$ with $\|\mathbf{z}\|^2 = 1$ that

$$1 = \|\mathbf{z}\|^2 = \| \sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i} \|_F^2 = \sum_{k=1}^{R_M} \sigma_k \left(\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i} \right)^2 \geq \sigma_{max} \left(\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i} \right)^2 .$$

However, if $\mathbf{z} = \mathbf{z_r}$ such that (4.14) holds, then the corresponding matrix $\sum_{i=1}^{R_M} z_{i,r} \overline{\mathbf{V}}_\mathbf{i}$ is rank one, hence only the maximal singular value is nonzero. Thus

$$\sigma_{max} \left(\sum_{i=1}^{R_M} z_{i,r} \overline{\mathbf{V}}_\mathbf{i} \right)^2 = 1,$$

which shows that $\mathbf{z_r}$ is a solution to the optimization problem with objective value 1. □

This theorem leads to an algorithm similar to that of the previous section to compute the composition if the assumptions in the theorem holds. One has to find all global solutions of the optimization problem (4.15). Since the structure of this problem is the same as (4.8), we can derive the optimality conditions as before and use a fixed-point iteration. Analogous to (4.10) we obtain:

Lemma 4.12
The solution vectors $\mathbf{z}$ to the maximization problem (4.15) satisfy the optimality condition

$$z_i = \gamma \langle \overline{\mathbf{V}}_\mathbf{i} \mathbf{c}(\mathbf{z}), \mathbf{b}(\mathbf{z}) \rangle \quad i = 1, \ldots R_M, \tag{4.16}$$

where $\mathbf{b}(\mathbf{z}), \mathbf{c}(\mathbf{z})$ are right and left singular vectors corresponding to the largest singular value of $\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i}$, and $\gamma \in \mathbb{R}$ is a normalization constant such that $\|\mathbf{z}\| = 1$.

Algorithm V in the next section is based on (4.16) and attempts to compute approximations to the following set with $R = R_M$

$$\mathrm{Sol}_R := \{\mathbf{z} \in \mathbb{R}^R \,|\, \mathbf{z} \text{ solves (4.16) with } \|z\| = 1 \text{ such that } \sigma_{max}(\sum_{i=1}^{R} z_i \overline{\mathbf{V}}_{\mathbf{i}}) = 1\},$$

where all elements in Sol_R are pairwise not collinear. Note that Sol_R can be empty, discrete, or a continuum.

The following theorem indicates, when the approach to compute an exact decomposition using the optimization of (4.15) will be successful, namely for the case that $R = R_M$ and Sol_R contains more than R linear independent solutions.

Theorem 4.13
The following assertions are equivalent

1. (4.12) holds with R, $\mathbf{B}$, $\mathbf{C}$ that satisfy $R = R_M$
2. there exists a set of R_M linearly independent vectors $(\mathbf{z_r})_{r=1}^{R_M}$ in Sol_{R_M}.

Proof
Theorem 4.11 and Lemma 4.12 is the statement $1. \to 2.$ For the converse we observe that as in the proof of Theorem 4.11, $\mathbf{z_r} \in \mathrm{Sol}_{R_M}$ imply that $\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_{\mathbf{i}}$ is a rank one matrix, hence the collection of the vectors $\mathbf{z_r}$ from Sol_{R_M} satisfy (4.14), and since the vectors $\mathbf{z_r}$ are assumed to be linear independent, $\mathrm{rank}(\mathbf{B} \odot \mathbf{C}) = R_M$. The inverse of the corresponding matrix $\mathbf{Z}$ yields R_M vectors such that (4.13) holds and thus (4.12). □

Following the same line of proof, we can also find in some situations an exact tensor decomposition if the rank condition is not satisfied, i.e., $R > R_M$, namely if we find enough linearly independent solutions in Sol_R.

Corollary 4.14
Let $R > R_M$ and suppose that at least R linearly independent vectors $(\mathbf{z_r})_{r=1}^{R}$ in Sol_R exist. Then (4.14) holds with some vectors $\mathbf{b_r}$, $\mathbf{c_r}$, $r = 1, \ldots, R$, and the corresponding matrices $\mathbf{B}$, $\mathbf{C}$ yield an exact rank-R decomposition (4.12)

Before we discuss the numerical results, we explain the previous theoretical result at some concrete (small sized) examples, where we can do the calculations analytically.

Example I Consider the "KHL"-data [28, 29, 10]. This is a $2 \times 2 \times 2$ tensor with top and bottom slices given by

$$T_{1,*,*} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad T_{2,*,*} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Kruskal showed that the KHL tensor has a tensorial rank of $R = 3$ (over $\mathbb{R}$). Using our methodology, we can prove that KHL tensor does not have an exact decomposition for $R = 1$ and $R = 2$. The associated tensor $\mathcal{M}$ in (3.6) (contracting over the first mode) is reordered into a 4×4 matrix $\mathbf{M}$

$$\mathbf{M} = \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}.$$

This matrix has rank $R_M = 2$ and an eigenvalue decomposition

$$(\overline{\lambda}_i)_{i=1}^4 = (2, 2, 0, 0) \quad \text{and} \quad \overline{\mathbf{V}} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ -1 & 0 & 1 & 0 \end{pmatrix}.$$

The columns in the matrix $\overline{\mathbf{V}}$ correspond to the normalized eigenvectors of $\mathbf{M}$. By Lemma 4.10 it always holds that $R \geq R_M = 2$, hence there is no exact rank-1 decomposition.

Is there an exact rank-2 decomposition? If this is the case, then the rank condition in Theorem 4.11 holds with $R = 2$ and hence, there must be at least two linear independent solution in Sol_2. Moreover, the matrix in the maximization (4.15) is (rearranging the first two eigenvectors of $\mathbf{M}$ into matrices)

$$\sum_{i=1}^{2} z_i \overline{\mathbf{V}}_{\mathbf{i}} = z_1 \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} + z_2 \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} z_1 & z_2 \\ z_2 & -z_1 \end{pmatrix}.$$

With the constraint $z_1^2 + z_2^2 = 1$, the matrix $\sum_{i=1}^{2} z_i \overline{\mathbf{V}}_{\mathbf{i}}$ has determinant $-\frac{1}{2}$, thus, no normalized vector (z_1, z_2) exists yielding a rank-1 matrix, which means that (4.14) cannot hold for such vectors. Hence, Sol_2 is empty in this case, which means no exact rank-2-decomposition exists.

However, we can find a rank-3 decomposition using Corollary 4.14. Set $R = R_M + 1 = 3$, i.e., we add an eigenvector of $\mathbf{M}$ corresponding to the 0-eigenvalue to the sum $\sum_i z_i \overline{\mathbf{V}}_{\mathbf{i}}$. For instance, we may add $\overline{\mathbf{V}}_{\mathbf{3}}$ yielding a matrix

$$\sum_{i=1}^{3} z_i \overline{\mathbf{V}}_{\mathbf{i}} = \frac{1}{\sqrt{2}} \begin{pmatrix} z_1 + z_3 & z_2 \\ z_2 & -z_1 + z_3 \end{pmatrix}$$

and maximize the first singular value over the vectors $\mathbf{z} = (z_1, z_2, z_3)$ normalized to one. We obtain a rank-1 matrix with $\sigma_{max}\left(\sum_{i=1}^{3} z_i \overline{\mathbf{V}}_{\mathbf{i}}\right) = 1$ if $z_3 = \pm\frac{1}{\sqrt{2}}$ and $z_1^2 + z_2^2 = \frac{1}{2}$. Thus, this describes the set Sol_3, which is a continuum here. Out of this set, we can easily extract three linear independent vectors; e.g., for any $t \neq 0$

$$\mathbf{z}_1 = \frac{1}{\sqrt{2}}(1, 0, 1), \quad \mathbf{z}_2 = \frac{1}{\sqrt{2}}(1, 0, -1), \quad \mathbf{z}_3 = \frac{1}{\sqrt{2}}(\cos(t), \sin(t), 1).$$

For these choices of the vectors, we obtain the matrices $\sum_{i=1}^{3} z_i \overline{\mathbf{V}}_{\mathbf{i}}$ (each one of rank 1):

$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 \\ 0 & -1 \end{pmatrix}, \quad \frac{1}{2} \begin{pmatrix} 1 + \cos(t) & \sin(t) \\ \sin(t) & 1 - \cos(t) \end{pmatrix}.$$

The singular vectors of the matrices corresponding to the singular value 1 are $\mathbf{u}_1 = \mathbf{v}_1 = (1, 0)^T$, $\mathbf{u}_2 = \mathbf{v}_2 = (0, 1)^T$, $\mathbf{u}_3 = \mathbf{v}_3 = (\cos(\frac{t}{2}), \sin(\frac{t}{2}))^T$, respectively. Collecting these vectors gives for any $t \neq 0$ the factor matrices $\mathbf{B}, \mathbf{C}$ of a rank-3 decomposition:

$$\mathbf{B} = \mathbf{C} = \begin{pmatrix} 1 & 0 & \cos(\frac{t}{2}) \\ 0 & 1 & \sin(\frac{t}{2}) \end{pmatrix},$$

with

$$\mathbf{A} = \begin{pmatrix} 1 & -1 & 0 \\ -\frac{\cos(\frac{t}{2})}{\sin(\frac{t}{2})} & -\frac{\sin(\frac{t}{2})}{\cos(\frac{t}{2})} & \frac{1}{\cos(\frac{t}{2})\sin(\frac{t}{2})} \end{pmatrix}.$$

Another interesting aspect is the rank-1 least squares minimization problem for the KHL tensor. Using Lemma 4.6, the functional in (4.9) is

$$\tilde{\mathfrak{J}}(\boldsymbol{\mu}) = 4 + 2(\mu_1^2 + \mu_2^2) - 2\sigma_{max}(\sqrt{2} \begin{pmatrix} \mu_1 & \mu_2 \\ \mu_2 & -\mu_1 \end{pmatrix}) = 4 + 2(\mu_1^2 + \mu_2^2) - 2\sqrt{2}\sqrt{\mu_1^2 + \mu_2^2},$$

which has minimizers μ_1 and μ_2 such that $\sqrt{\mu_1^2 + \mu_2^2} = \frac{1}{\sqrt{2}}$ with $\min_{\boldsymbol{\mu}} \tilde{\mathfrak{J}}(\boldsymbol{\mu}) = 3$. It can be calculated that the corresponding singular vectors associated to each value of (μ_1, μ_2) satisfies the optimality condition (4.10). Thus, the rank-1 least squares problem for this tensor has a continuum of solutions.

Example II Let us consider a second example $\mathcal{T} \in \mathbb{R}^{2\times 2\times 2}$ with the following entries:

$$\mathcal{T}_{1,*,*} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \mathcal{T}_{2,*,*} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{2\times 2\times 2}. \tag{4.17}$$

This is an example of a degenerate tensor, i.e., it has a tensorial rank of 3, but it can be approximated arbitrary well by rank-2 tensors [11]. Hence, this is the case where the infimum of the $R = 2$ least squares approximation is 0 but no minimum exists. The corresponding matrix $\mathbf{M}$ can be calculated to

$$\mathbf{M} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

The eigensystem is given by

$$(\overline{\lambda}_i)_{i=1}^4 = (2, 1, 0, 0) \qquad \text{and} \qquad \overline{\mathbf{V}} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{1}{\sqrt{2}} & 0 & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

We might try to find a $R = 2$ decomposition. Since $R_M = 2$ we can again use Theorem 4.11. The matrix of interest is

$$\sum_{i=1}^{2} z_i \overline{\mathbf{V}}_{\mathbf{i}} = \begin{pmatrix} z_2 & \frac{1}{\sqrt{2}} z_1 \\ \frac{1}{\sqrt{2}} z_1 & 0 \end{pmatrix}.$$

This matrix is of rank 1 (and its largest singular values is 1) if and only if $z_1 = 0$. Hence, we have the only possible maximizers $\mathbf{z} = (0, \pm 1)^T$ in Sol_2. Since they are clearly linearly dependent, we see that we cannot find 2 linear independent maximizer in this case, which shows that a rank-2 decomposition does not exist. We observe that the maximizers are degenerate in the sense that at $\mathbf{z} = (0, 1)^T$ not only the first derivative but also the second one of $\sigma_{max}(\sum_{i=1}^{2} z_i \overline{\mathbf{V}}_{\mathbf{i}})$ vanishes (e.g., by taking the derivative of the functional with $\mathbf{z} = (\sin(t), \cos(t))^T$ with respect to t at $t = 0$.).

We conjecture that it is a general feature (at least in the situation $R = R_M$) of degenerate problems, i.e., when the infimum in (4.12) is 0 but no minimum exists, that the maximizers in (4.15) appear with multiplicities.

4.4. *Inexact tensor decomposition*

Since the previous results are based on an optimization problem (4.15) is is clear that it can be modified to compute suboptimal solutions to the minimization problem (1.2). We consider now the case that (4.12) does not hold with a small R, and the objective is to find reasonable factor matrices that make (1.2) small.

Then, the modifications to the previous case are twofold and simple: firstly, we replace R_M in Theorem 4.11 by a smaller value $R_C < R_M$ and secondly we are satisfied with suboptimal (i.e. not necessarily maximizers) of the optimization problem (4.15)

Let us describe the difference to the case of an exact decomposition in the previous subsection. If we replace R_M by R_C and thus only consider expression of the form $\sigma_{max}(\sum_{i=1}^{R_C} z_i \overline{\mathbf{V}}_{\mathbf{i}})$ in Theorem 4.11, this has the same effect as replacing the original tensor $\mathcal{T}$ with one $\mathcal{T}'$ that has a corresponding matrix $\mathbf{M}$ with only R_C nonzero eigenvalues. This can be thought of as performing a kind of spectral cutoff. It is easy to see that we can estimate the norm of the difference by

$$\|\mathcal{T} - \mathcal{T}'\|^2 = \sum_{i=R_C+1}^{JK} \overline{\lambda}_i^2. \tag{4.18}$$

Concerning the second modification of a relaxation of the condition for a maximizer in Theorem 4.11, $\sigma_{max}(\sum_{i=1}^{R} z_i \overline{\mathbf{V}}_{\mathbf{i}}) = 1$, we have the following estimate.

Proposition 4.15
Let $R_M = R_C$, and $(\mathbf{z_r})$ be a set of R_C normalized linearly independent vectors that form the columns of a matrix $\mathbf{Z} \in \mathbb{R}^{R_C \times R_C}$ and which satisfy

$$\sigma_{max}(\sum_{i=1}^{R_C} z_{i,r}\overline{\mathbf{V}}_{\mathbf{i}}) = 1 - \alpha_r \quad \text{with } \alpha_r \geq 0 \quad \forall r = 1, \dots R_C,$$

Then if $\mathbf{b_r}, \mathbf{c_r}$ are the left and right singular vectors corresponding to the matrix $\sum_{i=1}^{R_C} z_{i,r}\overline{\mathbf{V}}_{\mathbf{i}}$, and if we take $\mathbf{B} = (\mathbf{b_r})_{r=1,\dots R_C}$ and $\mathbf{C} = (\mathbf{c_r})_{r=1,\dots R_C}$ we have the following estimate

$$\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) \leq \sigma_{max}(\mathbf{Z}^{-1}\Lambda)^2 \sum_{i=1}^{R_C} \alpha_r^2 \tag{4.19}$$

with $\Lambda = \text{diag}(\sqrt{\overline{\lambda}_1}, \dots \sqrt{\overline{\lambda}_{R_C}}) \in \mathbb{R}^{R_C \times R_C}$.

Proof
By construction, with σ_i being the singular values of $\sum_{i=1}^{R_C} z_{i,r}\overline{\mathbf{V}}_{\mathbf{i}}$ we have using orthogonality

$$\| \sum_{i=1}^{R_C} z_{i,r}\overline{\mathbf{V}}_{\mathbf{i}} - \mathbf{b_r} \otimes \mathbf{c_r} \|_F^2 = \sum_{i=2}^{IK} \sigma_i^2 + (1-\sigma_1)^2 = \| \sum_{i=1}^{R_C} z_{i,r}\overline{\mathbf{V}}\|_F^2 + \alpha_r^2 - 1 = \alpha_r^2.$$

Thus, using Theorem 3.14 we get (with $\mathbf{V} = (\overline{\mathbf{v}}_{\mathbf{1}} | \dots \overline{\mathbf{v}}_{\mathbf{R_C}})$)

$$\begin{aligned}
\mathfrak{J}_{red}(\mathbf{B}, \mathbf{C}) &\leq \sum_{i=1}^{R_M} \overline{\lambda}_i \| \overline{\mathbf{v}}_{\mathbf{i}} - (\mathbf{B} \odot \mathbf{C})\mathbf{Z}^{-1})_{i-\text{th column}} \|_{\mathbb{R}^{JK}}^2 \\
&= \sum_{i=1}^{R_C} \overline{\lambda}_i \| \left(\mathbf{VZ} - \mathbf{B} \odot \mathbf{C}\right) (\mathbf{Z}^{-1})_{i-\text{th column}} \|_{\mathbb{R}^{JK}}^2 \\
&= \| \left(\mathbf{VZ} - \mathbf{B} \odot \mathbf{C}\right) (\mathbf{Z}^{-1}\Lambda) \|_F^2 \leq \sigma_{max}(\mathbf{Z}^{-1}\Lambda)^2 \| \left(\mathbf{VZ} - \mathbf{B} \odot \mathbf{C}\right) \|_F^2 \\
&\leq \sigma_{max}(\mathbf{Z}^{-1}\Lambda)^2 \sum_{r=1}^{R_C} \alpha_r^2.
\end{aligned}$$

□

Thus, good approximate minimizers arise from a compromise between the objective (4.15) with value almost 1 and the need for linear independence characterized by the norm of $\mathbf{Z}^{-1}$.

We utilize Proposition 4.15 for the tensor $\mathcal{T}'$ which is obtained by cutting off the eigenvalues in M which are larger than R_C. In this situation we obtain a matrix $\mathbf{M}$ which has rank $R_M = R_C$ and Proposition 4.15 can be applied.

Altogether we obtain that such a decomposition agrees with a given tensor $\mathcal{T}$ up to computational bound that depends on R_C and α: let $\mathbf{Z}$ be a linear independent R_C-tuple of vectors satisfying the assumptions of Proposition 4.15 which form the columns of a $R_C \times R_C$ matrix $\mathbf{Z}$, let $\mathbf{B}, \mathbf{C}$, be the corresponding matrices in Proposition 4.15 and $\mathbf{A}$ given by (3.3) and Λ as in Proposition 4.15, then

$$\mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}) \leq \left(\sqrt{\sum_{i=R_C+1}^{JK} \overline{\lambda}_i^2} + \sqrt{\sigma_{max}(\mathbf{Z}^{-1}\Lambda)^2 \, R_C \alpha^2} \right)^2 . \tag{4.20}$$

Finally, let us emphasize that Proposition 4.15 does not require $\mathbf{z_r}$ to be solutions of the fixed points equation (4.16). They could have been obtained by any other method as well.

Example II (continued) Above, we showed that the tensor in (4.17) does not have a rank-2 decomposition. However, using Proposition 4.15 with $R_C = R_M = 2$ we may show that it can be approximated arbitrary well by a rank-2 tensor and hence the infimum in (4.12) is 0. Let us take a small perturbation of the maximum in Example II, $\mathbf{z} = (0, 1)$,

$$\mathbf{z}_1 = (\sin(\epsilon), \cos(\epsilon))^T \quad \mathbf{z}_2 = (-\sin(\epsilon), \cos(\epsilon))^T.$$

These two vectors are linearly independent for $\epsilon > 0$ and approximate the maximum. By a Taylor series, we can find α_r in Proposition 4.15 as

$$\alpha_1 = \alpha_2 = \frac{1}{8}\epsilon^4 + O(\epsilon^6)$$

The matrix $\mathbf{Z}$ having columns $\mathbf{z}_1$, $\mathbf{z}_2$ has an inverse

$$\mathbf{Z}^{-1} = \frac{1}{2}\begin{pmatrix} \frac{1}{\sin(\epsilon)} & \frac{1}{\cos(\epsilon)} \\ -\frac{1}{\sin(\epsilon)} & \frac{1}{\cos(\epsilon)} \end{pmatrix}.$$

For small ϵ we obtain that

$$\sigma_{max}(\mathbf{Z}^{-1}\Lambda) = \sigma_{max}(\mathbf{Z}^{-1}\text{diag}(\sqrt{2}, 1)) = \frac{1}{\sin(\epsilon)}.$$

Thus, the bound in Proposition 4.15 can be estimated to

$$\mathfrak{J}_{red} \leq \frac{1}{64}\frac{\epsilon^8 + O(\epsilon^{10})}{\sin(\epsilon)^2} \sim O(\epsilon^6),$$

which shows that $\inf \mathfrak{J}_{red} = 0$ we can approximate the tensor arbitrary well by rank-2 tensors.

5. ALGORITHMS

In this section we explain several algorithm which implement the theoretical concepts of the previous section. In particular, we describe the following methods: the Centroid Projection Algorithm (CePr) for finding suboptimal solutions to the general least squares problem (1.2), cf. Algorithm I, based on Theorem 4.3. The fixed point algorithm (FP-R1(local)), cf. Algorithm II for the computation of the best rank-1 approximation, based on Theorem 4.7 and the fixed point equation (4.10). This algorithm will be coupled with a globalization strategy yielding Algorithms III and IV, (FP-R1(RIG)) and (FP-R1(APIG)). Moreover, we discuss the methods for computing an exact rank-R decomposition, based on Theorem 4.11 using again a fixed-point iteration (FP-EX), cf. Algorithm V. Finally we sketch a method for computing a (suboptimal) general decomposition using the ideas of Section 4.4 and Proposition 4.15. The corresponding algorithm is Algorithm VI, (FP-INEX).

5.1. *Suboptimal solutions: the Centroid Projection Algorithm*

In Theorem 4.3 we proposed a way to calculate the matrices $\mathbf{B_C}$ and $\mathbf{C_C}$, which are supposed to be reasonable minimizers (with explicit bounds given in Corollary 4.4) of the general minimization problem (1.2) using the reduced functional. The corresponding algorithm is described in Algorithm I as a pseudo-code on the left column and as a MATLAB-pseudocode on the right-hand side.

The output of this algorithm is considered a suboptimal solution to the minimization problem for (1.2). In the MATLAB-pseudocode on the right, `katrao` is a user-defined function to build the Khatri-Rao product. For computational efficiency, we note that it is not necessary to form the matrix $\mathbf{M}$ (compare the code in step 1) because the eigenvectors $\overline{\mathbf{v}}_\mathbf{i}$ are the singular vectors of a matrix unfolding of $\mathcal{T}$. The same step is also part of the HOSVD algorithm [30].

If CePr is used as an initial guess for an ALS iteration, step 4 can be omitted, because it is already the first iteration of the ALS. The SVD and the matrix inversion in step 4 are the computationally most expensive parts.

Algorithm I. **Centroid Projection Algorithm (CePr)**

```
Input: Tensor $\mathcal{T} \in \mathbb{R}^{I\times J\times K}$, $R \in \mathbb{N}, 1 \le R \le \min(J,K)$
1.  Compute the eigensystem $(\bar\lambda_i, \bar{\mathbf{v}}_{\mathbf{i}})$ of $\mathbf{M}$   [V,D,U]=svd(reshape(T,[I J*K])')
2.  Compute $\bar{\mathbf{V}}^C$                                   xi=diag(D.^2)./sum(diag(D.^2))
                                                        Vc=reshape(V*xi,J,K)
3.  Compute the first $R$ left and right               [Uc,Sc,Vc]=svd(Vcentroid)
    singular vectors of $\bar{\mathbf{V}}^C$, and form              Bc=Uc(:,1:R)
    the matrix $\mathbf{B_C}$ and $\mathbf{C_C}$ from them.           Cc=Vc(:,1:R)
4.  Compute $\mathbf{A_C} := \tilde{\mathbf{A}}[\mathbf{B_C}, \mathbf{C_C}]$                  Ac=reshape(T,[I J*K])/...
    (from (3.2))                                              khatrao(Bc,Cc)
Output: $\mathbf{A_C}$, $\mathbf{B_C}$, $\mathbf{C_C}$                           return Ac, Bc, Cc
```

Remark 5.1
The complexity of the Centroid Projection algorithm can roughly be estimated as follows (SVD denotes the numerical complexity of a singular value decomposition).

$$\begin{aligned}\text{Complexity}_{CePr} \sim\ & \text{SVD(for matrix of size } I\times JK) + \text{SVD(for matrix of size } J\times K)\\ & (+\text{solving linear equation of size } JK\times JK)\,,\end{aligned}$$

where the last term can be omitted if the Centroid Projection is used as an initial guess for ALS. The complexity of this algorithm is comparable to the HOSVD algorithm, which needs three singular value decompositions of matrices of sizes $I\times JK$, $J\times IK$ and $K\times IJ$. Without the matrix inversion, CePr requires less work than HOSVD. In Section 6 we compare the running time of CePr with other algorithms.

5.2. *Best rank-1 fit by the fixed-point iterations*

In Theorem 4.7 we derived the optimality conditions (4.10) for the general minimization problem (1.2) in the rank-1 case $R = 1$ utilizing the reduced functional. Our algorithm is based on a fixed-point iteration for solving these optimality conditions which yields Algorithm II (FP-R1(local)): starting with an initial guess $\boldsymbol{\mu}_0$, we try to compute solutions of (4.10) by a fixed-point iteration until some convergence criteria (defined by the input parameter N_{max}, ϵ_{stop}) is satisfied. The corresponding output vectors $\mathbf{b}(\boldsymbol{\mu}_k)$, $\mathbf{c}(\boldsymbol{\mu}_K)$ are supposed to satisfy the optimality conditions of the $R = 1$ least squares minimization problem for $\mathfrak{J}_{red}$.

For an actual implementation of FP-R1(local), the code in Algorithm II should be modified to take into account that μ and ξ are only of size R_M, and the value of R_M should be precomputed. (For the sake of a simpler presentation, the MATLAB pseudocode is slightly inconsistent in this point.) Moreover, it is assumed here that svd stores the largest singular values at the first diagonal entry of D. This is not necessarily always true in MATLAB and should be checked as well. Finally, there are other options for the stopping criteria (step 6), which in our case terminates the iteration if $\|\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k-1}\| \le \epsilon_{stop}$ or the maximal number of iterations N_{max} is reached.

Remark 5.2
The main contribution to the numerical complexity is the singular value decomposition of $\mathbf{V}_\xi$, which has to be done once for each iteration. The total number of iterations is not known a-priori, so we can only roughly sketch the numerical complexity as

$$\begin{aligned}\text{Complexity}_{FP-R1(local)} \sim\ & \text{SVD(for matrix of size } I\times JK)\\ & + \text{SVD(for matrix of size } J\times K) * \text{number of iterations.}\end{aligned}$$

The fixed point iteration will only converge locally at best. Moreover, a limit of a convergent iteration is only a solution of the first order optimality condition (4.10) of (4.7) and thus only a stationary point, but not necessarily a global minimum. (This problem is not specific to the fixed point iteration, but appears as well for the ALS algorithm, see Remark 3.18).

Algorithm II. **Fixed Point Algorithm for Rank-1 Minimization (FP-R1(local))**

```
Input: Tensor $\mathcal{T} \in \mathbb{R}^{I\times J\times K}$, initial guess $\boldsymbol{\mu}_0 \in \mathbb{R}^{R_M}$, stopping parameters $\epsilon_{stop} > 0$,
       maximal number of iteration $N_{max}$
1.  Compute the eigensystem                      [V,D,U]=svd(reshape(T,[I J*K])')
      $(\overline{\lambda}_i, \overline{\mathbf{v}}_\mathbf{i})$ of $\mathbf{M}$                         mu=mu0;noconv=1
2.  Fixed point iteration:
    for $k = 0$ until convergence                 while(noconv)
3.    Compute $\xi$ by $\xi_i^k = \frac{\overline{\lambda}_i\ (\mu_k)_i}{\sqrt{\sum(\overline{\lambda}_i\ (\mu_k)_i)^2}}$      xi=diag(D).*mu
                                                   xi=xi./norm(xi)
4.    Form Matrix $\mathbf{V}_\xi = \sum_{i=1}^{R_M} \xi_i \overline{\mathbf{V}}_\mathbf{i}$          [Bxi,Dxi,Cxi]=...
      and compute $\mathbf{b}(\boldsymbol{\mu}_k), \mathbf{c}(\boldsymbol{\mu}_k)$               svd(reshape(V*xi,J,K))
      from singular vectors                        B=Bxi(:,1);C=Cxi(:,1)
      corresponding to the larges singular         muold=mu
      value of $V_{xi}$
5.    Update $\mu$ according to (4.10)                  for i=1:length(mu)
                                                   mu(i)=B'*(reshape(V(:,i),J,K)*C)
                                                   end
                                                   k=k+1
6.  If not converged, goto step 3                 noconv=(norm(mu-muold)>eps)...
                                                             &&(k<Nmax)
                                                  end
Output: $\mathbf{b}(\boldsymbol{\mu}_k)$,$\mathbf{c}(\boldsymbol{\mu}_k)$                            return B,C
```

In order to fix these problems, we use some globalization strategies. In the first approach, we simply run FP-R1(local) for several randomly generated initial guesses $\boldsymbol{\mu}_0$. For each of these guesses, we obtain several vectors $\mathbf{b}, \mathbf{c}$ as outputs of FP-R1(local). The final decomposition is taken as that pair of vectors $\mathbf{b}, \mathbf{c}$ for which $\mathfrak{J}_{red}(\mathbf{b}, \mathbf{c})$ has the minimal value amongst all outputs. Note that because the singular vectors do not change when scaling a matrix by a factor, the initial guess can also be scaled and, in particular, chosen as a point on the ellipsoid $\sum_{i=1}^{R_M} ((\boldsymbol{\mu}_0)_i \overline{\lambda}_i)^2 = 1$. This yields Algorithm III (FP-R1(RIG)).

Algorithm III. **FP-R1 with random initial guess (FP-R1(RIG))**

```
Input: Tensor $\mathcal{T} \in \mathbb{R}^{I\times J\times K}$, number of random initial guesses $N_{IG}$
       stopping parameters $\epsilon_{stop} > 0$, maximal number of iteration $N_{max}$
1.  Compute $R_M = \text{rank}(\mathbf{M})$
2.  Iterate from $i = 1$ To $N_{IG}$
3.    Generate random vectors $\boldsymbol{\mu}_0 \in \mathbb{R}^{R_M}$ with $\sum_{i=1}^{R_M} ((\boldsymbol{\mu}_0)_i \overline{\lambda}_i)^2 = 1$,
4.    Run (FP-R1(local)) with $\boldsymbol{\mu}_0$ as initial guesses; output: $\mathbf{b}_i, \mathbf{c}_i$
5.    Compute function value $\mathfrak{J}_i = \mathfrak{J}_{red}(\mathbf{b}_i, \mathbf{c}_i)$ using output of step 4.
6.    If $\mathfrak{J}_i < \mathfrak{J}_{min}$
        then $\mathbf{b}, \mathbf{c} = \mathbf{b}_i, \mathbf{c}_i, \mathfrak{J}_{min} = \mathfrak{J}_i$
7.  Goto Step 2.
Output: $\mathbf{b}, \mathbf{c}$
```

Instead of using random initial guesses $\boldsymbol{\mu}_0$ we can incorporate some a-priori information. According to Lemma 4.9, the values of $|(\boldsymbol{\mu}_{opt})_i|$ lie in an ellipsoid in $\mathbb{R}^{R_M}$. Now, an alternative to algorithm FP-R1(RIG) is to use initial guesses that are the "centers" of these ellipsoids. They are obtained by minimizing the lower bound (4.11) over $\boldsymbol{\mu}_{opt}$, which yields that $|\boldsymbol{\mu}_0| = \left(\sigma_{max}(\overline{\mathbf{V}}_\mathbf{i})\right)_{i=1}^{R_M}$. Here, however, we do not have any information on the sign of the components of $\boldsymbol{\mu}_0$, so we use all possible combination of signs and run FP-R1(local) for each of them. This gives Algorithm IV: FP-R1(APIG), quite similar to FP-R1(RIG).

Algorithm IV. **FP-R1 with a-priori guess (FP-R1(APIG))**

```
Input: Tensor $\mathcal{T} \in \mathbb{R}^{I\times J\times K}$,
        stopping parameters $\epsilon_{stop} > 0$, maximal number of iteration $N_{max}$
1.   Compute the eigensystem $(\overline{\lambda}_i, \overline{\mathbf{v}}_\mathbf{i})$ of $\mathbf{M}$ and $R_M$,
2.   Compute the vectors $\boldsymbol{\mu}_i = (\sigma_{max}(\overline{\mathbf{V}}_\mathbf{i}))_{i=1}^{R_M}$,
     where $\overline{\mathbf{V}}_\mathbf{i}$ is the matricized version of $\overline{\mathbf{v}}_\mathbf{i}$ in step 1.
2.   For all $s \in \{1, -1\}^{R_M}$
3.     Set $(\boldsymbol{\mu}_0)_i = s_i\mu_i, \quad i = 1, \dots, R_M$
4.     Run (FP-R1(local)) with $\boldsymbol{\mu}_0$ as initial guesses; output: $\mathbf{b}_i, \mathbf{c}_i$
5.     Compute function value $\mathfrak{J}_i = \mathfrak{J}_{red}(\mathbf{b}_i, \mathbf{c}_i)$ using output of step 4.
6.     If $\mathfrak{J}_i < \mathfrak{J}_{min}$
         then $\mathbf{b}, \mathbf{c} = \mathbf{b}_i, \mathbf{c}_i$, $\mathfrak{J}_{min} = \mathfrak{J}_i$
7.   End
Output: $\mathbf{b}, \mathbf{c}$
```

Remark 5.3

The complexity of the algorithm (FP-R1(RIG)) is clearly $\sim N_{IG} *$ Complexity$_{FP-R1(local)}$, where N_{IG} is the number of random initial guesses. Hence, for the performance, N_{IG} is important. If it is too large, the computation time is too high, if it is too small, we might miss the global minimum. We cannot give a practically useful value for N_{IG}; this depends on the behavior of the fixed point algorithm. We note that this algorithm becomes problematic for high values of R_M, which is the dimension of the initial guesses. To have an equally dense net of initial guesses in any dimensions, one would have to choose $N_{IG} \sim Const.^{R_M}$, i.e., exponentially increasing in R_M.

Comparing this with the algorithm (FP-R1(APIG), we observe that there we need 2^{R_M} runs of FP-R1(local), which is still of exponential order in R_M, but the number of runs is not an input parameter. In the later section, we see that at least for small R_M we get comparable results.

For Algorithm IV, there is no guarantee, that a global minimum is found. However, the numerical experiments show that in many situations (FP-R1(APIG)) yields similar results than (FP-R1(RIG)) with less computational effort. In this sense, it is a practically useful tool, whose theoretical convergence properties are still open.

5.3. *Exact rank-R decomposition*

In Theorem 4.11 we have found an equivalent way of computing the factor matrices of a rank-R tensor (4.12) if the rank-condition $R = R_M$ is satisfied. By solving the optimization problem (4.15) we can find the column vectors $\mathbf{b_r}$, $\mathbf{c_r}$ from the singular vectors of $\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i}$. The optimality conditions for (4.15) are stated in Lemma 4.12 and the idea is to apply a fixed-point iteration — similar as in the rank-1 case — to (4.16). Thus, we try to compute (approximations) to the set Sol$_{R_M}$.

However, it is not enough to find one solution of (4.16) but we have to look for all global maxima (in view of Theorem 4.13 we need at least R_M linearly independent solutions for a success). We try to achieve this in a similar manner as for the rank-1 fit, by starting with randomly selected initial guesses, performing the fixed point algorithm for each initial guess and eliminating those solution with are not maxima. This leads to Algorithm V (FP-EX).

The parameter ϵ_{stop}, N_{max} play the same role of stopping parameters for the fixed point iteration as they do in the Algorithm FP-R1(local). The value N_{IG} is the number of random initial guesses similar as in FP-R1(RIG). The parameter α is used to numerically test for $\sigma_{max}(\sum_{i=1}^{R_M} z_i \overline{\mathbf{V}}_\mathbf{i}) = 1$. If exact computations would be available it can be set to $\alpha = 0$, otherwise it can be set to the numerical accuracy of the singular value decomposition. The parameter ϵ_{test} is used to rule out antipodal pairs in the solution set. Note that $\mathbf{z}^\infty$ is always normalized to 1, so that the inner product abs$(\mathbf{z}^\infty, \mathbf{z}^{sol})$ in step 4 is a measure of the cosine of the angle spanned by these two vectors. New solutions $\mathbf{z}^\infty$ are removed from the solution set if they form an angle close to 0 or π with any of the previously

Algorithm V. **Fixed Point Algorithm for Rank-R tensors (FP-EX)**

```
Input:Tensor $\mathcal{T} \in \mathbb{R}^{I \times J \times K}$                                      ,
        tentative rank of decomposition $R > 1$
        stopping parameters for fixed point iteration $\epsilon_{stop} > 0$,
        maximal number of iteration $N_{max}$
        parameter to test for linear independence $\epsilon_{test}$
        parameter to test for optimality of function value $\alpha > 0$
        number of random initial guesses $N_{IG}$
1.   Compute the eigensystem $(\overline{\lambda}_i, \overline{\mathbf{v}}_\mathbf{i})$ of $\mathbf{M}$
2.   Outer iteration                          FOR OUTER=1:NIG
     Generate a random initial guess            z=rand(R,1);z=z./norm(z)
     $\mathbf{z}_0 \in \mathbb{R}^R$ with $\|\mathbf{z}_0\| = 1$                 Sol=[];BSol=[];CSol=[];
3.   Fixed point iteration $k = 0, \ldots$           while(noconv)
       Compute singular vectors                  [Bz,Dz,Cz]=...
         $\mathbf{c}(\mathbf{z^k}), \mathbf{b}(\mathbf{z^k})$                           svd(reshape(V*z,J,K))
                                                  B=Bz(:,1),C=Cz(:,1)
                                                  FOR i=1:R
       $\tilde{z}_i^{k+1} = \langle \overline{\mathbf{V}}_\mathbf{i} \mathbf{c}(\mathbf{z^k}), \mathbf{b}(\mathbf{z^k}) \rangle$            z(i)=B'*(reshape(V(:,i),J,K)*C)
                                                  END
       $\mathbf{z}^{k+1} = \frac{1}{\|\tilde{\mathbf{z}}^{k+1}\|}$                             z=z./norm(z)
     until convergence;                          %Converg. test (epsstop, Nmax)
     If converged, set $\mathbf{z}^\infty = \mathbf{z}^{k+1}$              zinf=z
4.   Test if $\sigma_{max}(\sum_{i=1}^{R_M} z_i^\infty \overline{\mathbf{V}}_\mathbf{i}) = 1$ and          IF(Dz(1,1)>1-alpha)&&...
     for linear independence of $\mathbf{z}^\infty$ to all         (abs(zinf'*zsol)>1-epstest)
     previously computed solutions $\mathbf{z}^\infty$           for all zsol in Sol
     if test passed, add $\mathbf{z}^\infty$ to solution set         Sol=[Sol zinf];
     and corresponding singular vectors          Bsol=[Bsol B];Csol=[Csol C];
                                                END %End of if-test
5.   End of outer iteration                   END
Output:
     Sol, the set of $\mathbf{z}^\infty$ which passed test
     and the corresponding singular vectors   return Sol,Bsol,Csol
```

computed solutions. In principle, this test is not needed, but it is used because for any solution to the fixed point equation, $\mathbf{z}^\infty$, its antipode, $-\mathbf{z}^\infty$ is a solution as well. Thus, by this test we obtain a smaller output `Sol` than without it.

Remark 5.4
The numerical complexity of this algorithm is mainly determined by the singular value decompositions to find $\mathbf{c}(\mathbf{z^k}), \mathbf{b}(\mathbf{z^k})$, and it is comparable to that of FP-R1(RIG)

$$\text{Complexity}_{FP-EX} \sim N_{IG} * (\text{no. of fixed point iterations}) * \text{SVD}(\text{for matrix of size } J \times K).$$

If we use the same rule-of-thumb as for FP-R1(RIG), $N_{IG} \sim \text{Const}^R$, this indicates that the algorithm is only useful for small values of R.

Remark 5.5
The exponential scaling of the complexity of tensor decompositions has been observed in literature, e.g., in [31]. There, a multigrid approach using so-called reduced HOSVD algorithms was proposed for an orthogonal Tucker approximation having linear complexity bounds in the grid size and ranks.

Algorithm FP-EX requires some postprocessing and a choice of the input value R, the tentative rank of $\mathcal{T}$ (which is not know a-priori). The postprocessing involves testing for linear independent elements in Sol_R. In the MATLAB code, `Sol` is a matrix of size $\mathbb{R}^{R \times S}$, with S the number of

$\mathbf{z}^\infty$ that passed the test in step 4. We can easily test if there is an R-tuple of linear independent columns in `Sol` by testing the rank of `Sol`. In view of Theorem 4.13 and because R_M can easily be computed we therefore propose to start FP-EX with $R = R_M$. We sketch the processing scheme for an exact decomposition in the following table.

I. Run FP-EX with $R = R_M$
  Ia. If rank `Sol`$= R$ then a solution exists.
    (if `Sol`$\in \mathbb{R}^{R\times R}$ a unique solution exists)
    (if `Sol`$\in \mathbb{R}^{R\times S}$, $S > R$ possibly multiple solutions exit)
  Ib. If rank `Sol`$< R$, then no R_M-decomposition exists
   set $R = R_M + 1$ and goto step II.
    (test for degeneracy)
II. Run FP-EX with $R > R_M$ (ambiguity in Algorithm FP-EX !)
  IIa. rank `Sol`$= R$ then a solution exists.
  IIb. rank `Sol`$< R$ then set $R = R + 1$ and goto step II.

Let us comment on this scheme: The case that Ia. yields a solution (if exact computations would be available) is the statement of Theorem 4.13. The case $S > R$, i.e., that we have more than R vectors in Sol_R is no problem, but it indicates that multiple solution to the rank decomposition exist. In fact, here any R-tuple of linear independent vectors in Sol_R (columns in `Sol`) gives a solution by Theorem 4.13. The corresponding matrices $\mathbf{B}, \mathbf{C}$ can be read from the output `Bsol`, `Csol` of FP-EX . In the case 1b), by Theorem 4.13, no solution exists. However, it should be noted that we can use our algorithm to test for a possible degeneracy, i.e., when the infimum in (4.12) is 0 but no minimum exists with $R = R_M$. Above we expressed the conjecture that this is the case when some extrema of the optimization problem (4.15) appear with multiplicities such that higher order derivatives of the functional in (4.15) vanishes. If our conjecture is true, then testing for degeneracy only requires computing higher order derivatives of (4.15) at all $\mathbf{z}$ in `Sol`.

In case Ib. we propose to increase the tentative rank by one and rerun the algorithm again (step II.). We are then in the situation of Corollary 4.14. However, the reader should be cautioned, because in this case the algorithm is not well-defined: if $R > R_M$, then in step 3 in FP-EX we have to take the first R_M singular vectors but also a number of $R - R_M$ singular vectors $\mathbf{c}(\mathbf{z^k}), \mathbf{b}(\mathbf{z^k})$ corresponding to singular values with $\sigma_i = 0$. In this step it is not clear which one of them we should choose. For instance if $R = R_M + 1$ we have to add one pair of singular vectors out of the $\max\{J,K\} - R_M$ possibilities corresponding to the zero singular values. If $R = R_M + 2$ we have $\begin{pmatrix} \max\{J,K\} - R_M \\ 2 \end{pmatrix}$ possibilities. It is possible to modify Algorithm FP-EX accordingly to test for all these possibilities, but soon this becomes computationally unfeasible. Hence, unless the tensor rank R of $\mathcal{T}$ satisfies $R = R_M + 1$, or $R = R_M + 2$, continuing with step II. in the above scheme is not really practically. Note, however, that the MATLAB code does run without any problem even for $R > R_M$ because numerically the singular values are never 0 and MATLAB always returns an ordered vector of singular values, such that everything is well-defined.

Still if we were able to compute R linear independent solutions in (step IIa.), by Corollary 4.14 a solution to the decomposition problem exists. Even if we were able to run step II. for all $R = R_M, \dots \max\{J,K\}$, we do not yet have a full understanding if we would always find a solution. We leave this issue to future work.

We suggest to view Algorithm FP-EX mainly as a method to compute exact solutions in the "nice" case $\mathrm{rank}(\mathcal{T}) = R_M$.

5.4. *Inexact tensor decomposition*

As it was indicated in Section 4.4, Algorithm FP-EX can be modified to compute approximate solutions to the general least squares problem even if (4.12) does not hold. By using a cutoff-index R_C for the spectrum and a relaxation parameter α for the optimality condition we come to Algorithm VI, which output might not be a solution of the least squares problem, but at least it satisfies the bounds in (4.20).

Algorithm VI. **Inexact Fixed Point Algorithm tensors (FP-INEX)**

```
 Input:   $R_C < R_M$, a cutoff index of the singular value decomposition
          $\alpha > 0$, threshold parameter,
          $\epsilon_{stop}, N_{max}, \epsilon_{test}, N_{IG}$
     1.   Run FP-EX with $R = R_C < R_M$ and $\alpha > 0$
     2.   If Sol has rank $R$ Stop
          Else modify $R$ and $\alpha$ and goto step 1
Output:   Sol, Bsol, Csol
```

As before, FP-INEX, requires some pre- and postprocessing: an initial value R_C can be found by looking at the eigenvalues of $\mathbf{M}$ with the objective that the bound on the right of (4.18) is small and R_C is small as well. Moreover, α is chosen sufficiently small. If in step 2 of FP-INEX a rank-$R = R_C$ matrix Sol was found, the algorithm run successfully. Otherwise we have to modify R and α. It can happen, that the initial value of α was too small, but with a larger choice of α we find a rank-R output Sol. In this case we succeeded again. However, it can be the case that because there are too few fixed-points, Sol does not have sufficiently many linear independent vectors, even with increasing α. In this case we have to modify R: we can try a value of R given by the rank of Sol and run FP-INEX again. However, it is not guaranteed that such a modification always gives a success in the sense of a rank-R matrix Sol. In this case, we can modify R again or admit that FP-INEX fails.

In case of a success of FP-INEX we might have an output Sol containing more than one linear independent R-tuple of vectors. In this case, we have many optimal solutions satisfying the above bound (4.20). Unfortunately, FP-INEX does not tell which R-tuple of vectors in Sol we should pick. We can look for one which have small value of $\sigma_{max}(\mathbf{Z}^{-1}\Lambda)$, but we are not aware of an algorithm that can do this, other than testing all R-tuples in Sol.

6. NUMERICAL RESULTS

We performed several numerical experiments for the algorithms presented here: the Centroid Projection method (CePr), the fixed point iteration for rank-1 minimization (FP-R1) (and its variants) and the fixed point iteration for exact and approximate reconstruction of a rank-R decomposition (FP-(IN)EX). All the computations were done in MATLAB on an Intel Xeon 2.8 GHz processor.

Comparison of CePr and FP-R1. We first study the performance of CePr and FP-R1 on small-sized tensors. We randomly generate 6400 tensors of size $I \times J \times K$ with zero-mean Gaussian entries. We let the dimensions run from $I, J, K = 3, \ldots 6$. For each fixed tuple (I, J, K), a sample of 100 random tensors was generated yielding a total of 6400 tensors. For each tensor, the factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are computed by various methods listed in Table I for different choices of R. We then calculate the relative deviation of $\mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C})$ from the optimal functional value $\mathfrak{J}_{opt}$ defined as

$$\text{dev} := \frac{(\mathfrak{J}(\mathbf{A}, \mathbf{B}, \mathbf{C}) - \mathfrak{J}_{opt})}{\mathfrak{J}_{opt}}.$$

The optimal value $\mathfrak{J}_{opt}$ is computed by the ALS iteration [32]. We used three variants of ALS, which differ only in the way how the initial guess is chosen (ALS+HOSVD ALS+CePr, ALS+Random, see below). The value $\mathfrak{J}_{opt}$ is taken as the minimal residual of the output of these three ALS iterations. Of course, we cannot guarantee here that the output is a global minimum because it could be just a stationary point as well, but the results are consistent with the assumption that $\mathfrak{J}_{opt}$ is a global minimum in the sense that no smaller values $\mathfrak{J}$ (up to numerical tolerances) were observed by the alternative optimization routines (dev was always positive). For advanced method to improve this situation for ALS we refer to [33, 34].

For the case of $R = 1$, $\mathfrak{J}_{opt}$ is computed by FP-R1. For all test cases, we first calculated and contracted $\mathcal{M}$ over the first mode; i.e., $\mathbf{M}$ is of size $JK \times JK$. The parameters for FP-R1(RIG) and FP-R1(APIG) are set as follow: $\epsilon_{stop} = 10^{-6}$, $N_{IG} = N_{max} = 100$. The ALS iteration was stopped when the norm of the difference between the factor matrices of two successive iterations is less than 10^{-6} times the norm of the initial matrices. Table I compares several methods tested on the random tensors through the following variables: the mean value of the deviation, $\overline{\text{dev}}$, the median, med(dev) (both scaled by a factor 10^{-2}), and the mean of the computation time, $\overline{\text{time}}$, in 10^{-3} seconds. The first two lines show the results for the CePr method. CePr_{init} refers to the Centroid Projection algorithm where the factor matrix $\mathbf{A}_C$ is not computed (see the discussion on the complexity); this method only differs in computation time from CePr. The next lines refer to the HOSVD and DTLD methods, followed by the two variants of the fixed-point algorithm FP-R1 (which are only meaningful for $R = 1$). In the last three lines, we run the ALS algorithm once for each tensor with an initial guess that is obtained either from HOSVD method, CePr_{init} or by a random choice (ALS+Random).

Table I. Comparison of deviation of optimality of the algorithms for randomly generated tensor

	$R=1$			$R=2$			$R=3$		
Method	$\overline{\text{dev}}$	med(dev)	$\overline{\text{time}}$	$\overline{\text{dev}}$	med(dev)	$\overline{\text{time}}$	$\overline{\text{dev}}$	med(dev)	$\overline{\text{time}}$
CePr	7.40	6.27	1.1	30.8	26.5	1.1	94.5	68.7	1.1
CePr_{init}			0.4			0.4			0.4
HOSVD	16.45	14.43	0.9	64.1	56.3	0.9	176.7	132.4	0.9
DTLD	5.86	3.83	30.6	4.0	0.0	20.7	30.1	15.5	30.6
FP-R1(RIG)	0	0	1496.0						
FP-R1(APIG)	1.13	0	498.0						
ALS+HOSVD	0.2	0	12.1	0.5	0.0	100.4	0.9	0.0	245.4
ALS+CePr	1.2	0	13.6	1.1	0.0	111.7	0.8	0.0	249.0
ALS+Random	1.4	0	14.2	1.2	0.0	109.2	1.1	0.0	255.5

The results in Table I show that CePr outperforms the HOSVD method. CePr_{init} is about twice as fast as HOSVD and yields residuals smaller than that of HOSVD. In the rank-1 case with $R = 1$, we also observe that CePr already yields a small residual without using any further processing; its function value is on average only 7.40% higher than the optimal value. However, when using ALS with CePr as an initial guess, the ALS+HOSVD method is slightly faster because it requires less iterations. The reason for this strange behavior is unclear to us. Nevertheless, the ALS with all three starters are comparable. (Note that the deviation in the table is given in $\%$.)

A comparison of DTLD to both CePr and HOSVD shows that DTLD gives better results but it requires more time. DTLD's performance is in between the fast approximate methods (CePr, HOSVD) and the slower minimization methods of ALS-type.

In our experiments, FP-R1(RIG) always finds the optimum in the rank-1 case, while FP-R1(APIG) is faster but fails to find the optimum in a few cases (the method still converges to a stationary point). However for both methods, the computation time is much higher than the ALS method. Although the fixed point iterations in FP-R1 converge quite fast (usually less than 10 steps), the globalization procedures makes the algorithm computationally very expensive. The results below indicate that a reduction in the number of initial guesses, N_{IG}, can reduce the running time, but probably this is not enough to beat ALS.

Exact/Inexact decomposition for different tensor ranks and sizes. In the second example, we investigate the numerical performance for an exact reconstruction of the fixed point method FP-EX. We generate 6400 tensors from random factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ of size $I \times 3$, $J \times 3$ and $K \times 3$. By construction, each tensor is of rank at most 3. Similarly as before, we generate 100 tensors from the factor matrices for each modal dimension $I, J, K = 3, \ldots 6$, adding up to 6400 test tensors. We run FP-EX with parameters $\alpha = 10^{-1}$, $\epsilon_{stop} = 10^{-6}$, $\epsilon_{test} = 10^{-3}$ $N_{IG} = N_{max} = 100$ and $R = 3$. In all of these cases, FP-EX is able to obtain a very good least squares fit. We find a fit with relative least squares norm less than 10^{-5} in 89% of the cases, 10^{-3} in 93% of the cases and 10^{-1} in 98% of the cases. The average computation time is one second (comparable to FP-R1(RIG) in Table I). Note that FP-EX fails if the output does not contains R linearly independent solutions in Sol_R. This was no problem for the tensors created by random factor matrices; FP-EX always run successfully.

Furthermore, we compare FP-EX and FP-INEX with ALS for tensors of larger sizes to investigate the performance for practically relevant problems. The results are displayed in Table II. Here we use tensors with a certain rank (rank $\mathcal{T}$) and size (I, J, K) by generating one random set of factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ of size $(I, \text{rank } \mathcal{T})$, $(J, \text{rank } \mathcal{T})$, and $(K, \text{rank } \mathcal{T})$. We run FP-INEX and FP-EX against ALS+HOSVD for different choices of R (see first column of II). When the tensorial rank of $\mathcal{T}$ is R, FP-INEX is equal to FP-EX, and thus, both methods are displayed in the same column. The parameter setup for the fixed point iterations is the same as in the previous paragraphs; in particular, we use a fixed number of random initial guesses, $N_{IG} = 100$. In the case R is less than the tensorial rank of $\mathcal{T}$, FP-INEX successfully computes R fixed points if α is sufficiently small. (Note that varying α does not need extra work as this can be done after the fixed point iterations). For the case $R = \text{rank } \mathcal{T}$, FP-EX runs successfully with the choice $\alpha = 10^{-1}$.

The table gives the relative residual and the computation time (in seconds) of the respective methods. We observe that each of the two methods recover the rank correctly as indicated by a zero residual in the last row. Also, the residuals are comparable but it can be seen that the computation time for FP-(IN)EX is longer than for ALS. Note that ALS is extremely fast when R is equal to the rank of $\mathcal{T}$. We cannot explain this phenomenon at this point. For the fixed point methods, the computation time is rather independent of R because we use a fixed number of initial guesses N_{IG}. The results are good results, but it also indicate more room for improvements (see also the next example). Obviously, the difference in the running time between ALS and FP-(IN)EX is smaller for the case of JK being small which is favorable for FP-(IN)EX.

FP-INEX for examples from tensor database. Finally, we investigate the performance of the fixed point algorithms for more practical and relevant problems. In particular, we consider applying FP-INEX to experimental tensor data. Here we used two datasets from Bro's tensor database [35]: the amino acid data [36, 35] of size $5 \times 201 \times 61$ and the "Kojima Girls“ data [37] of size $153 \times 4 \times 20$. At first we considered the amino acid data which has a suitable fit when $R = 3$ according to [35]. As this is a $5 \times 201 \times 61$ tensor, we did all the computations by contracting over the largest mode $J = 201$ (cf. Remark 3.4).

Note that FP-INEX requires some non-automatic pre- and postprocessing methods. We describe the procedures as follows. First of all, we have to select a suitable value of R_C. This can be found by looking at the eigenvalues of $\mathbf{M}$ and finding an appropriate cutoff value. Here the choice $R_C = 3$ seems to be useful, because if we calculate the relevant error in (4.18) we observe that $\frac{1}{\overline{\lambda}_{max}^2} \sum_{i=R_C+1}^{IK} \overline{\lambda}_i^2 < 10^{-7}$ for $R_C = 3$ while it is of order 10^{-2} for $R_C = 2$. This dramatic drop indicates $R_C = 3$ to be a good choice. We then run FP-INEX with parameter setup: $N_{max} = 100$, $\epsilon_{stop} = 10^{-6}$, $\epsilon_{test} = 10^{-3}$, $\alpha = 10^{-1}$. The output `Sol` contains exactly 3 fixed points, hence this is a well-behaved case when the second condition in Theorem 4.13 is satisfied. The corresponding residual (see the last line in Table III) is of the same order as that obtained by ALS with the same setup as above. This shows that the two methods yield comparable results.

In order to compare the computation times, we test how the output changes when the number of initial guess N_{IG} changes. Since a small number N_{IG} yields a faster method, it is of interest to reduce this number. So we test the algorithm for the choices $N_{IG} = 10, 30, 70, 100$: the running time, the number of computed fixed points (i.e., the number of columns in `Sol`) and the corresponding relative residual are displayed in Table III.

The table indicates that the output (3 fixed points and a similar residual) is almost the same for all the choices of N_{IG}, but the running time can be reduced using a smaller number of N_{IG}. It can

Table II. Comparison of FP-INEX/FP-EX and ALS for different dimensions and ranks

	rank $\mathcal{T} = 5$ $(I, J, K) = (30, 30, 30)$				rank $\mathcal{T} = 15$ $(I, J, K) = (30, 30, 30)$			
R	residual		time		residual		time	
	ALS	FP-(IN)EX	ALS	FP-(IN)EX	ALS	FP-(IN)EX	ALS	FP-(IN)EX
1	252.58	252.94	3.14	42.20	610.34	614.52	3.12	41.42
2	169.10	187.51	3.66	47.28	536.97	550.54	3.66	49.59
3	100.00	111.12	4.15	48.68	480.25	538.31	4.10	52.28
4	45.06	51.23	4.54	48.03	419.74	487.30	4.52	51.25
5	0.00	0.00	0.08	48.37	372.95	424.06	5.07	49.53
14					12.72	13.88	10.48	48.30
15					0.00	0.00	0.20	48.00

	rank $\mathcal{T} = 5$ $(I, J, K) = (100, 10, 10)$				rank $\mathcal{T} = 12$ $(I, J, K) = (100, 10, 10)$			
R	residual		time		residual		time	
	ALS	FP-(IN)EX	ALS	FP-(IN)EX	ALS	FP-(IN)EX	ALS	FP-(IN)EX
1	123.79	124.00	1.14	6.38	263.62	270.49	1.14	5.93
2	76.55	80.06	1.43	6.95	194.28	194.83	1.43	6.38
3	29.31	29.32	1.72	7.27	152.19	152.39	1.72	6.52
4	7.46	7.47	1.99	7.42	122.60	153.19	1.98	8.67
5	0.00	0.00	0.07	7.29	99.53	101.22	2.31	8.51
11					8.47	15.15	4.51	9.39
12					0.00	0.00	0.21	8.16

Table III. Comparison of FP-INEX and ALS for amino acid data using different numbers of initial guesses

	FP-INEX $N_{IG} = 10$	FP-INEX $N_{IG} = 30$	FP-INEX $N_{IG} = 70$	FP-INEX $N_{IG} = 100$	ALS
time	6.4 s	18.8 s	43.3 s	62.8 s	12.8
|`Sol`|	3	3	3	3	
rel. res.	$0.68.\ 10^{-3}$	$0.68.\ 10^{-3}$	$0.68.\ 10^{-3}$	$0.68.\ 10^{-3}$	$0.62.\ 10^{-3}$

be observed that for the choice $N_{IG} = 10$, FP-INEX is actually faster (by a factor of 2) than ALS. This shows that with some tuning, FP-INEX is competitive to traditional methods at least when R and IK are sufficiently small.

We did the same computation using $R_C = 4$. In this case, the choice $N_{IG} = 10$ was not enough to compute 4 fixed points. However, with $N_{IG} = 30$, FP-INEX successfully gives 4 fixed points and the relative residuals were: relative residual$_{FP-INEX} = 0.52.\ 10^{-3}$, relative residual$_{ALS} = 0.55.\ 10^{-3}$ (time$_{FP-INEX} = 18.6s$, time$_{ALS} = 14.4s$).

The second example out of this database is called the "Kojima Girls", originally by H. Kojima [37]. The data is given by a $153 \times 4 \times 20$ tensor, thus a contraction over the first index is used in the computations. According to the data source, this tensor has a degeneracy. A first look at the eigenvalues of $\mathbf{M}$ indicates no clear cutoff value R_C to choose since the eigenvalues decay smoothly and there is no sudden "drop" to 0. We therefore test FP-INEX for increasing values of R_C with the same parameter setup as for the amino data (in particular $N_{IG} = 100$ is the same for all values of R_C). For ALS we use a random initial guess, because HOSVD cannot be used for $R > \min(J, K)$. The algorithm FP-INEX runs successfully in the sense that in each case, the number of fixed points in `Sol` is larger than the input value R, which indicates a multiplicity of solutions. In Table IV we

Table IV. Comparison of FP-INEX and ALS for Kojima Girls data

R	rel. residual		time		$\lvert\texttt{Sol}\rvert$
	ALS	FP-IN.	ALS	FP-IN.	
2	0.0158	0.0164	2.34	6.74	28
3	0.0121	0.0131	2.80	6.82	33
4	0.0103	0.0120	3.26	6.95	41
7	0.0072	0.0084	4.93	7.37	46
12	0.0049	0.0068	8.08	8.05	49
18	0.0036	0.0046	13.19	8.64	50
25	0.0025	0.0036	21.54	9.70	50

display the residuals, computation time and the number of computed fixed points in $\texttt{Sol}$ (= $|\texttt{Sol}|$). Since in this situation, there were more fixed points than the value of R, we simply took the first R columns in $\texttt{BSol}$, $\texttt{CSol}$ to extract the factor matrices $\mathbf{B}$, $\mathbf{C}$.

It can be seen that the obtained residuals of the two method are comparable (with ALS having a slightly smaller residual) and that FP-INEX actually is faster than ALS for larger values of R. This is because we are in the case that the tensor has one large dimension and two rather small ones. Moreover, since we took N_{IG} constant, the computation time does not increase with R but it does so for ALS. What is also important is that we find in any case more fixed points in $\texttt{Sol}$ than R, which indicates a sort of nonuniqueness of the optimal decomposition.

To demonstrate this nonuniqueness we considered the case $R = 3$ in detail. Recall that if $|\texttt{Sol}|$ is larger than R, we can take any linear independent R-tuple of vectors in $|\texttt{Sol}|$ to compute the corresponding matrices $\mathbf{B}$, $\mathbf{C}$ as in Proposition 4.15. However, only those tuples which have a small value of $\sigma_{max}(\mathbf{Z}^{-1}\Lambda)$ yield a small residual according to (4.19). Unfortunately, FP-INEX does not give information which tuples satisfy this condition. So we test all 3-tuples in $|\texttt{Sol}|$ (these are $\begin{pmatrix} 33 \\ 3 \end{pmatrix} = 5456$ possibilities). Out of these tuples, we select those which have a value of $\sigma_{max}(\mathbf{Z}^{-1}\Lambda)$ smaller than a certain threshold. Then, for each of them we calculate the corresponding residual. We found 6 solutions that have the same or a slightly smaller (within 0.1%) residual than that one computed by ALS. Moreover, 58 solutions had a residual with less than 1%. Further analysis shows that these solutions are independent, in the sense, that they cannot be transformed into another by scaling or permutations. This shows that this data is problematic, because it has a multiplicity of (almost) optimal solutions.

7. CONCLUSION

We investigated the three components CP decomposition based on the reduced functional. The corresponding optimization problem was reformulated into several equivalent forms: the Rayleigh quotient form and the projection form. The latter formula focuses on the Khatri-Rao range of the factors rather than the factor matrices. The analysis led to several new algorithms relying only on basic matrix decompositions.

The first one, the Centroid Projection method (CePr), allows us to compute suboptimal solution for the general least squares minimization problem for CP decomposition together with suitable a-posteriori estimates. The numerical tests show that CePr is competitive to the traditional methods such as HOSVD. In many cases, it is faster and gives more accurate solutions than the standard techniques, in particular, for small R. It can also provide initial guesses to iterative methods such as ALS.

The second method, FP-R1, (with the variants FP-R1(RIG), FP-R1(APIG)) is a fixed-point method which is derived from the optimality condition of the rank-1 reduced least squares minimization problem. Numerically, we observed that it does compute the optimal solutions in our test cases, but it is slow for many problems. The main obstacle in the efficiency is the globalization strategy and the number of initial guesses, N_{IG}. The method FP-R1 attempts to compute all local minima of the rank-1 least squares problem. This is in contrast to the conventional methods where only one solution is computed at a time. In this sense, it delivers more information than standard methods.

Finally, the third method is also based on a fixed point algorithm which recovers a rank-R decomposition (Algorithm FP-EX) or finds good solution for the general minimization problem (1.2) (using FP-INEX). The method FP-EX works if the condition $R = R_M$ is satisfied (or for $R - R_M \in \{1, 2\}$ with some limitations). The numerical results indicate that for random tensors this condition seems to hold generically. Although its computation time is, in general, longer than the required time for ALS. For the case when one of the mode has a large dimension in comparison to the other two factors, the method is competitive with ALS and the fixed point methods can be tuned to improve performance and become faster than ALS.

Similar conclusions apply to the inexact method FP-INEX, which obtains (sub)optimal solution to the general least squares minimization problem (1.2). Its success depends on a condition such that a sufficiently large number of linear independent fixed points have to exist. By some tuning of N_{IG}, we can obtain residuals up to the ALS precision. In addition, if the dimensions I, J, K and rank R are appropriate, then it can beat ALS.

The main difference between the algorithms presented and the traditional methods is that our methods yield more qualitative information on the solutions. The fixed point method gives indications if the least squares problem has a unique solution or many solutions. Moreover, a degenerate problem can be detected. Neither of these characterizations can be done with traditional algorithms immediately. Furthermore, our algorithms are based on variational formulations with a-posteriori estimates on the quality of the computed solutions.

The understanding of the role of some important parameters, in particular N_{IG}, can hopefully be enhanced in future work. Moreover, since the basis of the computation are on the functionals (4.9) and (4.15), there is still plenty of room for improvement to our methods. For instance, alternative optimization routines can improve the fixed-point methods.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous referees for their valuable comments.

REFERENCES

1. F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys.* 1927; **6**: 164–189.
2. F. L. Hitchcock. Multilple invariants and generalized rank of a p-way matrix or tensor. *J. Math. Phys.* 1927; **7**: 39–79.
3. J.D. Carroll, J.J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of ‘Eckart-Young’ decomposition. *Psychometrika* 1970; **35**: 283–319.
4. R.A. Harshman. Foundations of the PARAFAC procedure: models and conditions for an ”explanatory” multi-modal factor analysis. *UCLA working papers in phonetics* 1970; **16**: 1–84.
5. T. Kolda, B.W. Bader. Tensor decompositions and applications. *SIREV* 2009; **51**: 455–500.
6. P. Comon, X. Luciani, A.L.F. de Almeida. Tensor decompositions, altenating least squares and other tales. *J. Chemometrics* 2009; **23**: 393–405.
7. M. Rajih, P. Comon. *Enhanced line search: A novel method to accelerate Parafac.* in the 13th Proceedings of the European Signal Processing Conference, Antalya, Turkey, September 2005.
8. P. Paatero. The Multilinear Engine - a table-driven least squares program for solving multilinear problems, including the n-way Parallel Factor Analysis model. *J. Comput. Graph. Stat.* 1999; **8**: 854–888.
9. N. Li, S. Kindermann, C. Navasca. *Some convergence results of a regularized alternating least-squares method for tensor decomposition, Linear Algebra Appl.* 2013; **438**: 796-812.
10. J. Tendeiro, M. B. Dosse, J. M. F. ten Berge. First and second-order derivatives for CP and INDSCAL, *Chemometrics and Intelligent Laboratory System* 2011; **106**: 27–36.
11. V. de Silva, L.-H. Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM J. Matrix Anal. Appl.* 2008; **30**: 1084–1127.
12. M. Brazell, N. Li, C. Navasca, C. Tamon. Tensor and matrix inversions and applications. Preprint, http://arxiv.org/abs/1109.3830
13. T. Kato. *Perturbation Theory for Linear Operators.* Springer, Berlin, 1995.

14. R. Bhatia. *Matrix Analyis.* Springer, New York, 1996.
15. W.P. Krijnen, T.K. Dijkstra, A. Stegeman. On the non-existence of optimal solutions and the occurrence of "degeneracy" in the Candecomp/Parafac model. *Psychometrika* 2008; **73**: 431–439.
16. A. Stegeman, L. De Lathauwer. A method to avoid diverging components in the Candecomp/Parafac model for generic $I \times J \times 2$ arrays. *SIAM J. Matrix. Anal. Appl.* 2009; **30**: 1614–1638.
17. L. De Lathauwer, B. De Moor, J. Vandewalle. On the best rank-1 and rank-$(R_1, R_2, ..., R_N)$ approximation of higher-order tensors. *SIAM J. Matrix Anal. Appl.* 2000; **21**: 1324–1342.
18. A. Uschmajew. Local convergence of the alternating least squares algorithm for canonical tensor approximation, Preprint #112, DFG-SPP 1324, University Marburg, 2011.
19. P. Paatero. Construction and analysis of degenerate PARAFAC models. *J. Chemometrics* 2000; **14**: 285–299.
20. A. Stegeman. Degeneracy in Candecomp/Parafac explained for $p \times p \times 2$ arrays of rank $p + 1$ or higher. *Psychometrika* 2006; **71**: 483–501.
21. A. Stegeman. Low-rank approximation of generic $p \times q \times 2$ arrays and diverging components in the Candecomp/Parafac model. *SIAM J. Matrix. Anal. Appl.* 2008; **30**: 988–1007.
22. A. Lorber. Features of quantifying chemical-composition from two-dimensional data array by the rank annihilation factor-analysis method. *Anal. Chem.* 1985; **12**: 2395–2397.
23. E. Sanchez, B.R. Kowalski. Tensorial resolution: a direct trilinear decomposition. *J. Chemometrics* 1990; **4**: 29–45.
24. S.E. Leurgans, T. Ross, R.B. Abel. Decomposition for Three-Way Arrays. *SIAM J. Matrix Anal. Appl.* 1993; **14**: 1064–1083.
25. C.F. Van Loan. A general matrix eigenvalue algorithm. *SIAM J. Num. Anal.* 1975; **12**: 819–834.
26. T. Zhang, G. H. Golub. Rank-one approximation to high order tensors. *SIAM J. Matrix Anal. Appl.* 2001; **23**: 534–550.
27. T. Kolda. Orthogonal tensor decomposition. *SIAM J. Matrix Anal. Appl.* 2001; **23**: 243–255.
28. J.B. Kruskal, R. A. Harshman, M. E. Lundy. *Some relationsships between Tucker's three-mode factor analysis and PARAFAC/CANDECOMP.* Paper Presented at the annual Meeting of the Psychometric Society, Los Angles, 1983.
29. J. M. F. Ten Berge, H. A. L. Kiers, J. De Leeuw. Explicit CANDECOMP/PARAFAC solutions for a contrived $2 \times 2 \times 2$ array of rank three, *Psychometrica* 1988; **53**: 579–583.
30. L. De Lathauwer, B. De Moor, J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.* 2000; **21**: 1253–1278.
31. B. N. Khoromskij and V. Khoromskaja. Multigrid accelerated tensor approximation of function related multidimensional arrays. *SIAM J. Sci. Comput.* 2009; **31**: 3002–3026.
32. C. Navasca, L. De Lathauwer, S. Kindermann. Swamp reducing technique for tensor decomposition, in the 16th Proceedings of the European Signal Processing Conference, Lausanne, August 2008.
33. E. Acar, D. M. Dunlavy, T. G. Kolda. A scalable optimization approach for fitting canonical tensor decompositions. *J. Chemometrics* 2011; **25**: 67-86.
34. H. de Sterck. A nonlinear GMRES optimization algorithm for Canonical Tensor Decomposition. *SIAM J. Sci. Comput.* 2012; **34**: A1351–A1379.
35. R. Bro. Public data sets for multivariate data analysis, http://www.models.kvl.dk/datasets.
36. R. Bro. PARAFAC: Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems* 1997; **38**: 149–171.
37. H. Kojima. Inter-battery factor analysis of parents and children reports of parental behavior. *Japanese Psychological Bulletin* 1975; **17**: 33–48.