

DECOMPOSITIONS OF A HIGHER-ORDER TENSOR IN BLOCK TERMS—PART III: ALTERNATING LEAST SQUARES ALGORITHMS*

LIEVEN DE LATHAUWER[†] AND DIMITRI NION[‡]

Abstract. In this paper we derive alternating least squares algorithms for the computation of the block term decompositions introduced in Part II. We show that degeneracy can also occur for block term decompositions.

Key words. multilinear algebra, higher-order tensor, Tucker decomposition, canonical decomposition, parallel factors model

AMS subject classifications. 15A18, 15A69

DOI. 10.1137/070690730

1. Introduction.

1.1. Organization of the paper. In the companion paper [11] we introduce decompositions of a higher-order tensor in several types of block terms. In the present paper we propose alternating least squares (ALS) algorithms for the computation of these different decompositions.

In the following subsections we first explain our notation and introduce some basic definitions. In section 1.4 we briefly recall the Tucker decomposition/higher-order singular value decomposition (HOSVD) [40, 41, 6, 7, 8] and also the Canonical/Parallel Factor (CANDECOMP/PARAFAC) decomposition [3, 15] and explain how they can be computed.

In section 2 we present an ALS algorithm for the computation of the decomposition in rank- $(L_r, L_r, 1)$ terms. In section 3 we discuss the decomposition in rank- (L, M, N) terms. Section 4 deals with the type-2 decomposition in rank- (L, M, \cdot) terms. Section 5 is a note on degeneracy.

1.2. Notation. We use \mathbb{K} to denote \mathbb{R} or \mathbb{C} when the difference is not important. In this paper scalars are denoted by lowercase letters (a, b, \dots) , vectors are written in boldface lowercase $(\mathbf{a}, \mathbf{b}, \dots)$, matrices correspond to boldface capitals $(\mathbf{A}, \mathbf{B}, \dots)$, and tensors are written as calligraphic letters $(\mathcal{A}, \mathcal{B}, \dots)$. This notation is consistently used for lower-order parts of a given structure. For instance, the entry with row index i and column index j in a matrix \mathbf{A} , i.e., $(\mathbf{A})_{ij}$, is symbolized by a_{ij} (also $(\mathbf{a})_i = a_i$ and $(\mathcal{A})_{ijk} = a_{ijk}$). If no confusion is possible, the i th column vector of a matrix \mathbf{A}

*Received by the editors May 7, 2007; accepted for publication (in revised form) by J. G. Nagy April 14, 2008; published electronically September 25, 2008. This research was supported by Research Council K.U.Leuven: GOA-Ambiorics, CoE EF/05/006 Optimization in Engineering (OPTEC), CIF1; F.W.O.: project G.0321.06 and Research Communities ICCoS, ANMMM, and MLDM; the Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, “Dynamical systems, control and optimization,” 2007–2011); and EU: ERNSI.

<http://www.siam.org/journals/simax/30-3/69073.html>

[†]Subfaculty Science and Technology, Katholieke Universiteit Leuven Campus Kortrijk, E. Sabbelaan 53, 8500 Kortrijk, Belgium (Lieven.DeLathauwer@kuleuven-kortrijk.be), and Department of Electrical Engineering (ESAT), Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium (Lieven.DeLathauwer@esat.kuleuven.be, <http://homes.esat.kuleuven.be/~delathau/home.html>).

[‡]Department of Electronic and Computer Engineering, Technical University of Crete, Kounoupidiana Campus, Chania, Crete, 731 00, Greece (nion@telecom.tuc.gr).

is denoted as \mathbf{a}_i , i.e., $\mathbf{A} = [\mathbf{a}_1 \ \mathbf{a}_2 \ \dots]$. Sometimes we will use the MATLAB colon notation to indicate submatrices of a given matrix or subtensors of a given tensor. Italic capitals are also used to denote index upper bounds (e.g., $i = 1, 2, \dots, I$). The symbol \otimes denotes the Kronecker product,

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \dots \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}.$$

Let $\mathbf{A} = [\mathbf{A}_1 \ \dots \ \mathbf{A}_R]$ and $\mathbf{B} = [\mathbf{B}_1 \ \dots \ \mathbf{B}_R]$ be two partitioned matrices. Then the Khatri–Rao product is defined as the partitionwise Kronecker product and represented by \odot [34]:

$$(1.1) \quad \mathbf{A} \odot \mathbf{B} = (\mathbf{A}_1 \otimes \mathbf{B}_1 \ \dots \ \mathbf{A}_R \otimes \mathbf{B}_R).$$

In recent years, the term “Khatri–Rao product” and the symbol \odot have mainly been used in the case where \mathbf{A} and \mathbf{B} are partitioned into vectors. For clarity, we denote this particular, columnwise, Khatri–Rao product by \odot_c :

$$\mathbf{A} \odot_c \mathbf{B} = (\mathbf{a}_1 \otimes \mathbf{b}_1 \ \dots \ \mathbf{a}_R \otimes \mathbf{b}_R).$$

The superscripts \cdot^T , \cdot^H , and \cdot^\dagger denote the transpose, complex conjugated transpose, and Moore–Penrose pseudoinverse, respectively. The operator $\text{diag}(\cdot)$ stacks its scalar arguments in a square diagonal matrix. Analogously, $\text{blockdiag}(\cdot)$ stacks its vector or matrix arguments in a block-diagonal matrix. The $(N \times N)$ identity matrix is represented by $\mathbf{I}_{N \times N}$. $\mathbf{1}_N$ is a column vector of all ones of length N . The zero tensor is denoted by \mathcal{O} .

1.3. Basic definitions.

DEFINITION 1.1. Consider $\mathcal{T} \in \mathbb{K}^{I_1 \times I_2 \times I_3}$ and $\mathbf{A} \in \mathbb{K}^{J_1 \times I_1}$, $\mathbf{B} \in \mathbb{K}^{J_2 \times I_2}$, $\mathbf{C} \in \mathbb{K}^{J_3 \times I_3}$. Then the Tucker mode-1 product $\mathcal{T} \bullet_1 \mathbf{A}$, mode-2 product $\mathcal{T} \bullet_2 \mathbf{B}$, and mode-3 product $\mathcal{T} \bullet_3 \mathbf{C}$ are defined by

$$\begin{aligned} (\mathcal{T} \bullet_1 \mathbf{A})_{j_1 i_2 i_3} &= \sum_{i_1=1}^{I_1} t_{i_1 i_2 i_3} a_{j_1 i_1} & \forall j_1, i_2, i_3, \\ (\mathcal{T} \bullet_2 \mathbf{B})_{i_1 j_2 i_3} &= \sum_{i_2=1}^{I_2} t_{i_1 i_2 i_3} b_{j_2 i_2} & \forall i_1, j_2, i_3, \\ (\mathcal{T} \bullet_3 \mathbf{C})_{i_1 i_2 j_3} &= \sum_{i_3=1}^{I_3} t_{i_1 i_2 i_3} c_{j_3 i_3} & \forall i_1, i_2, j_3, \end{aligned}$$

respectively [5].

In this paper we denote the Tucker mode- n product in the same way as in [4]; in the literature the symbol \times_n is sometimes used [6, 7, 8].

DEFINITION 1.2. The Frobenius norm of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ is defined as

$$\|\mathcal{T}\| = \left(\sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K |t_{ijk}|^2 \right)^{\frac{1}{2}}.$$

DEFINITION 1.3. The outer product $\mathcal{A} \circ \mathcal{B}$ of a tensor $\mathcal{A} \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_P}$ and a tensor $\mathcal{B} \in \mathbb{K}^{J_1 \times J_2 \times \dots \times J_Q}$ is the tensor defined by

$$(\mathcal{A} \circ \mathcal{B})_{i_1 i_2 \dots i_P j_1 j_2 \dots j_Q} = a_{i_1 i_2 \dots i_P} b_{j_1 j_2 \dots j_Q}$$

for all values of the indices.

For instance, the outer product \mathcal{T} of three vectors \mathbf{a} , \mathbf{b} , and \mathbf{c} is defined by $t_{ijk} = a_i b_j c_k$ for all values of the indices.

DEFINITION 1.4. A mode- n vector of a tensor $\mathcal{T} \in \mathbb{K}^{I_1 \times I_2 \times \dots \times I_n}$ is an I_n -dimensional vector obtained from \mathcal{T} by varying the index i_n and keeping the other indices fixed [19].

Mode- n vectors generalize column and row vectors.

DEFINITION 1.5. The mode- n rank of a tensor \mathcal{A} is the dimension of the subspace spanned by its mode- n vectors.

The mode- n rank of a higher-order tensor is the obvious generalization of the column (row) rank of a matrix.

DEFINITION 1.6. A third-order tensor is rank- (L, M, N) if its mode-1 rank, mode-2 rank, and mode-3 rank are equal to L , M , and N , respectively.

A rank- $(1, 1, 1)$ tensor is briefly called rank-1. The rank of a tensor is now defined as follows.

DEFINITION 1.7. The rank of a tensor \mathcal{T} is the minimal number of rank-1 tensors that yield \mathcal{T} in a linear combination [24].

It will be useful to write tensor expressions in terms of matrices or vectors. We therefore define standard matrix and vector representations of a third-order tensor.

DEFINITION 1.8. The standard $(JK \times I)$ matrix representation $(\mathcal{T})_{JK \times I} = \mathbf{T}_{JK \times I}$, $(KI \times J)$ representation $(\mathcal{T})_{KI \times J} = \mathbf{T}_{KI \times J}$, and $(IJ \times K)$ representation $(\mathcal{T})_{IJ \times K} = \mathbf{T}_{IJ \times K}$ of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ are defined by

$$\begin{aligned} (\mathbf{T}_{JK \times I})_{(j-1)K+k, i} &= (\mathcal{T})_{ijk}, \\ (\mathbf{T}_{KI \times J})_{(k-1)I+i, j} &= (\mathcal{T})_{ijk}, \\ (\mathbf{T}_{IJ \times K})_{(i-1)J+j, k} &= (\mathcal{T})_{ijk} \end{aligned}$$

for all values of the indices [19]. The standard $(IJK \times 1)$ vector representation $(\mathcal{T})_{IJK} = \mathbf{t}_{IJK}$ of \mathcal{T} is defined by

$$(\mathbf{t}_{IJK})_{(i-1)JK+(j-1)K+k} = (\mathcal{T})_{ijk}$$

for all values of the indices.

Note that in these definitions indices to the right vary more rapidly than indices to the left. Further, the k th $(I \times J)$ matrix slice of $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ will be denoted as $\mathbf{T}_{I \times J, k}$.

1.4. HOSVD and PARAFAC. We have now enough material to introduce the HOSVD [6, 7, 8] and PARAFAC [15] decompositions.

DEFINITION 1.9. A HOSVD of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ is a decomposition of \mathcal{T} of the form

$$(1.2) \quad \mathcal{T} = \mathcal{D} \bullet_1 \mathbf{A} \bullet_2 \mathbf{B} \bullet_3 \mathbf{C}$$

in which

- the matrices $\mathbf{A} \in \mathbb{K}^{I \times L}$, $\mathbf{B} \in \mathbb{K}^{J \times M}$ and $\mathbf{C} \in \mathbb{K}^{K \times N}$ are columnwise orthonormal,

- the core tensor $\mathcal{D} \in \mathbb{K}^{L \times M \times N}$ is
 - all-orthogonal,

$$\langle \mathbf{D}_{M \times N, l_1}, \mathbf{D}_{M \times N, l_2} \rangle = \text{trace}(\mathbf{D}_{M \times N, l_1} \cdot \mathbf{D}_{M \times N, l_2}^H) = \sigma_{l_1}^{(1)^2} \delta_{l_1, l_2},$$

$$1 \leq l_1, l_2 \leq L,$$

$$\langle \mathbf{D}_{N \times L, m_1}, \mathbf{D}_{N \times L, m_2} \rangle = \text{trace}(\mathbf{D}_{N \times L, m_1} \cdot \mathbf{D}_{N \times L, m_2}^H) = \sigma_{m_1}^{(2)^2} \delta_{m_1, m_2},$$

$$1 \leq m_1, m_2 \leq M,$$

$$\langle \mathbf{D}_{I \times J, n_1}, \mathbf{D}_{I \times J, n_2} \rangle = \text{trace}(\mathbf{D}_{L \times M, n_1} \cdot \mathbf{D}_{L \times M, n_2}^H) = \sigma_{n_1}^{(3)^2} \delta_{n_1, n_2},$$

$$1 \leq n_1, n_2 \leq N;$$

- ordered,

$$\sigma_1^{(1)^2} \geq \sigma_2^{(1)^2} \geq \dots \geq \sigma_L^{(1)^2} \geq 0,$$

$$\sigma_1^{(2)^2} \geq \sigma_2^{(2)^2} \geq \dots \geq \sigma_M^{(2)^2} \geq 0,$$

$$\sigma_1^{(3)^2} \geq \sigma_2^{(3)^2} \geq \dots \geq \sigma_N^{(3)^2} \geq 0.$$

Equation (1.2) can be written in terms of the standard $(JK \times I)$, $(KI \times J)$, and $(IJ \times K)$ matrix representations of \mathcal{T} as follows:

$$(1.3) \quad \mathbf{T}_{JK \times I} = (\mathbf{B} \otimes \mathbf{C}) \cdot \mathbf{D}_{MN \times L} \cdot \mathbf{A}^T,$$

$$(1.4) \quad \mathbf{T}_{KI \times J} = (\mathbf{C} \otimes \mathbf{A}) \cdot \mathbf{D}_{NL \times M} \cdot \mathbf{B}^T,$$

$$(1.5) \quad \mathbf{T}_{IJ \times K} = (\mathbf{A} \otimes \mathbf{B}) \cdot \mathbf{D}_{LM \times N} \cdot \mathbf{C}^T.$$

This decomposition is a specific instance of the Tucker decomposition, introduced in [40, 41]; columnwise orthonormality of \mathbf{A} , \mathbf{B} , \mathbf{C} and all-orthogonality and ordering of \mathcal{D} were suggested in the computational strategy in [40, 41]. The decomposition exists for any $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$. The matrices \mathbf{A} , \mathbf{B} , and \mathbf{C} can be computed as the matrices of right singular vectors associated with the nonzero singular values of $\mathbf{T}_{JK \times I}$, $\mathbf{T}_{KI \times J}$, and $\mathbf{T}_{IJ \times K}$, respectively. The core tensor is then given by $\mathcal{D} = \mathcal{T} \bullet_1 \mathbf{A}^H \bullet_2 \mathbf{B}^H \bullet_3 \mathbf{C}^H$. The values L , M , and N correspond to the rank of $\mathbf{T}_{JK \times I}$, $\mathbf{T}_{KI \times J}$, and $\mathbf{T}_{IJ \times K}$, i.e., they are equal to the mode-1, mode-2, and mode-3 rank of \mathcal{T} , respectively. Given the way (1.2) can be computed, it comes as no surprise that the SVD of matrices and the HOSVD of higher-order tensors have some analogous properties [6].

Define $\tilde{\mathcal{D}} = \mathcal{D} \bullet_3 \mathbf{C}$. Then

$$(1.6) \quad \mathcal{T} = \tilde{\mathcal{D}} \bullet_1 \mathbf{A} \bullet_2 \mathbf{B}$$

is a (normalized) *Tucker-2 decomposition* of \mathcal{T} .

We are often interested in the best *approximation* of a given tensor \mathcal{T} by a tensor of which the mode-1 rank, mode-2 rank, and mode-3 rank are upper-bounded by L , M , and N , respectively. Formally, we want to find $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathcal{D})$ such that $\hat{\mathcal{T}} = \mathcal{D} \bullet_1 \mathbf{A} \bullet_2 \mathbf{B} \bullet_3 \mathbf{C}$ minimizes the least-squares cost function $f(\hat{\mathcal{T}}) = \|\mathcal{T} - \hat{\mathcal{T}}\|^2$. One difference between matrices and tensors is that this optimal approximation cannot in general be obtained by simple truncation of the HOSVD. The algorithms discussed in [7, 8, 14, 17, 20, 21, 22, 23, 44] aim at finding the optimal approximation. These algorithms can be initialized with the approximation obtained by truncation.

Besides the HOSVD, there exist other ways to generalize the SVD of matrices. The most well known is CANDECOMP/PARAFAC [3, 15].

DEFINITION 1.10. A canonical or parallel factor decomposition (CANDECOMP/PARAFAC) of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ is a decomposition of \mathcal{T} as a linear combination of rank-1 terms:

$$(1.7) \quad \mathcal{T} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r.$$

In terms of the standard matrix representations of \mathcal{T} , decomposition (1.7) can be written as

$$(1.8) \quad \mathbf{T}_{JK \times I} = (\mathbf{B} \odot_c \mathbf{C}) \cdot \mathbf{A}^T,$$

$$(1.9) \quad \mathbf{T}_{KI \times J} = (\mathbf{C} \odot_c \mathbf{A}) \cdot \mathbf{B}^T,$$

$$(1.10) \quad \mathbf{T}_{IJ \times K} = (\mathbf{A} \odot_c \mathbf{B}) \cdot \mathbf{C}^T.$$

In terms of the $(IJK \times 1)$ vector representation of \mathcal{T} , the decomposition can be written as

$$(1.11) \quad \mathbf{T}_{IJK} = (\mathbf{A} \odot_c \mathbf{B} \odot_c \mathbf{C}) \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}.$$

PARAFAC components are usually estimated by minimization of the quadratic cost function

$$(1.12) \quad f(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \|\mathcal{T} - \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r\|^2.$$

This is most often done by means of an ALS algorithm, in which the vectors are updated mode per mode [3, 37]. Since PARAFAC is trilinear in its arguments, updating \mathbf{A} , given \mathbf{B} and \mathbf{C} , is just a linear least squares problem. The same holds for updating \mathbf{B} , given \mathbf{A} and \mathbf{C} , and updating \mathbf{C} , given \mathbf{A} and \mathbf{B} . The algorithm is outlined in Table 1.1. The normalization of \mathbf{B} and \mathbf{C} , in steps 2 and 3, respectively, is meant to avoid over- and underflow. Scaling factors are absorbed in the matrix \mathbf{A} . Note that the matrices $\mathbf{B} \odot_c \mathbf{C}$, $\mathbf{C} \odot_c \mathbf{A}$, and $\mathbf{A} \odot_c \mathbf{B}$ have to have at least as many rows as columns and that they have to be full column rank.

ALS iterations are sometimes slow. In addition, it is sometimes observed that the algorithm moves through a “swamp”: the algorithm seems to converge, but then the convergence speed drastically decreases and remains small for several iteration steps, after which it may suddenly increase again. Recently, it has been understood that the multilinearity of PARAFAC allows for the determination of the optimal step size, which improves convergence [33].

In many applications one can assume that \mathbf{A} and \mathbf{B} are full column rank (this implies that $R \leq \min(I, J)$) and that \mathbf{C} does not contain collinear vectors. Assume for convenience that the values c_{21}, \dots, c_{2R} are nonzero, such that $\mathbf{T}_{I \times J, 2}$ is rank- R , and that the values $c_{11}/c_{21}, \dots, c_{1R}/c_{2R}$ are mutually different. (If this is not the case, then we can consider linear combinations of slices such that the following reasoning applies.) Then \mathbf{A} follows from the eigenvalue decomposition (EVD) $\mathbf{T}_{I \times J, 1} \cdot \mathbf{T}_{I \times J, 2}^\dagger = \mathbf{A} \cdot \text{diag}(c_{11}/c_{21}, \dots, c_{1R}/c_{2R}) \cdot \mathbf{A}^\dagger$. In other words, the columns of $(\mathbf{A}^T)^\dagger$ are generalized eigenvectors of the pencil $(\mathbf{T}_{I \times J, 1}^T, \mathbf{T}_{I \times J, 2}^T)$; see [1, 13] and references therein. After having found \mathbf{A} , matrix \mathbf{B} may, up to a scaling of its columns, be

TABLE 1.1
ALS algorithm for CANDECOMP/PARAFAC.

-
- Initialize \mathbf{B} , \mathbf{C}
 - Iterate until convergence:
 1. Update \mathbf{A} :

$$\mathbf{A} \leftarrow [(\mathbf{B} \odot_c \mathbf{C})^\dagger \cdot \mathbf{T}_{JK \times I}]^T$$

2. Update \mathbf{B} :

$$\tilde{\mathbf{B}} = [(\mathbf{C} \odot_c \mathbf{A})^\dagger \cdot \mathbf{T}_{KI \times J}]^T$$

For $r = 1, \dots, R$: $\mathbf{b}_r \leftarrow \tilde{\mathbf{b}}_r / \|\tilde{\mathbf{b}}_r\|$

3. Update \mathbf{C} :

$$\tilde{\mathbf{C}} = [(\mathbf{A} \odot_c \mathbf{B})^\dagger \cdot \mathbf{T}_{IJ \times K}]^T$$

For $r = 1, \dots, R$: $\mathbf{c}_r \leftarrow \tilde{\mathbf{c}}_r / \|\tilde{\mathbf{c}}_r\|$

obtained from $(\mathbf{A}^\dagger \cdot \mathbf{T}_{I \times J, 2})^T = \mathbf{B} \cdot \text{diag}(c_{21}, \dots, c_{2R})$. Matrix \mathbf{C} may then be computed as $[(\mathbf{A} \odot_c \mathbf{B})^\dagger \cdot \mathbf{T}_{IJ \times K}]^T$. The EVD solution may subsequently be used to initialize the ALS algorithm. This approach has been proposed in [2, 26, 35, 36].

From a numerical point of view, it is preferable to take all the matrix slices of \mathcal{T} into account, instead of only two of them. We therefore proposed to compute the solution by means of simultaneous matrix diagonalization in [9]. It was shown in [10] that the solution can still be obtained by means of a simultaneous matrix diagonalization when \mathcal{T} is tall in its third mode (meaning that $R \leq K$ and $R(R-1) \leq I(I-1)J(J-1)/2$).

In [32] a Gauss–Newton method is described, in which all the factors are updated simultaneously; in addition, the inherent indeterminacy of the decomposition has been fixed by adding a quadratic regularization constraint on the component entries. Instead of the least squares error (1.12), one can also minimize the least absolute error. To this end, an alternating linear programming algorithm as well as a weighted median filtering iteration are derived in [42].

2. Decomposition in rank- $(L_r, L_r, 1)$ terms.

2.1. Definition.

DEFINITION 2.1. A decomposition of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ in a sum of rank- $(L_r, L_r, 1)$ terms, $1 \leq r \leq R$, is a decomposition of \mathcal{T} of the form

$$(2.1) \quad \mathcal{T} = \sum_{r=1}^R (\mathbf{A}_r \cdot \mathbf{B}_r^T) \circ \mathbf{c}_r,$$

in which the matrix $\mathbf{A}_r \in \mathbb{K}^{I \times L_r}$ and the matrix $\mathbf{B}_r \in \mathbb{K}^{J \times L_r}$ are rank- L_r , $1 \leq r \leq R$.

Define $\mathbf{A} = [\mathbf{A}_1 \dots \mathbf{A}_R]$, $\mathbf{B} = [\mathbf{B}_1 \dots \mathbf{B}_R]$, $\mathbf{C} = [\mathbf{c}_1 \dots \mathbf{c}_R]$. In terms of the standard matrix representations of \mathcal{T} , (2.1) can be written as

$$(2.2) \quad \mathbf{T}_{IJ \times K} = [(\mathbf{A}_1 \odot_c \mathbf{B}_1) \mathbf{1}_{L_1} \dots (\mathbf{A}_R \odot_c \mathbf{B}_R) \mathbf{1}_{L_R}] \cdot \mathbf{C}^T,$$

$$(2.3) \quad \mathbf{T}_{JK \times I} = (\mathbf{B} \odot \mathbf{C}) \cdot \mathbf{A}^T,$$

$$(2.4) \quad \mathbf{T}_{KI \times J} = (\mathbf{C} \odot \mathbf{A}) \cdot \mathbf{B}^T.$$

TABLE 2.1
ALS algorithm for decomposition in rank-($L_r, L_r, 1$) terms.

-
- Initialize \mathbf{B}, \mathbf{C}
 - Iterate until convergence:
 1. Update \mathbf{A} :

$$\mathbf{A} \leftarrow [(\mathbf{B} \odot \mathbf{C})^\dagger \cdot \mathbf{T}_{JK \times I}]^T$$

2. Update \mathbf{B} :

$$\tilde{\mathbf{B}} = [(\mathbf{C} \odot \mathbf{A})^\dagger \cdot \mathbf{T}_{KI \times J}]^T$$

For $r = 1, \dots, R$: *QR*-factorization: $\tilde{\mathbf{B}}_r = \mathbf{Q}\mathbf{R}$, $\mathbf{B}_r \leftarrow \mathbf{Q}$

3. Update \mathbf{C} :

$$\tilde{\mathbf{C}} = \left\{ [(\mathbf{A}_1 \odot_c \mathbf{B}_1) \mathbf{1}_{L_1} \dots (\mathbf{A}_R \odot_c \mathbf{B}_R) \mathbf{1}_{L_R}]^\dagger \cdot \mathbf{T}_{IJ \times K} \right\}^T$$

For $r = 1, \dots, R$: $\mathbf{c}_r \leftarrow \tilde{\mathbf{c}}_r / \|\tilde{\mathbf{c}}_r\|$

2.2. Algorithm. Like PARAFAC, the decomposition in rank- $(L_r, L_r, 1)$ terms is trilinear in the component matrices \mathbf{A} , \mathbf{B} , and \mathbf{C} . This means that updating \mathbf{A} , given \mathbf{B} and \mathbf{C} , is just a linear least squares problem. The same holds for updating \mathbf{B} , given \mathbf{A} and \mathbf{C} , and updating \mathbf{C} , given \mathbf{A} and \mathbf{B} . The update rules follow directly from (2.2)–(2.4). The algorithm is outlined in Table 2.1. The normalization in steps 2 and 3 are meant to avoid under- and overflow. Moreover, the normalization in step 2 prevents the submatrices of \mathbf{B} from becoming ill-conditioned. Analogous to the situation for PARAFAC, the matrices $\mathbf{B} \odot_c \mathbf{C}$, $\mathbf{C} \odot_c \mathbf{A}$, and $[(\mathbf{A}_1 \odot_c \mathbf{B}_1) \mathbf{1}_{L_1} \dots (\mathbf{A}_R \odot_c \mathbf{B}_R) \mathbf{1}_{L_R}]$ have to have at least as many rows as columns and have to be full column rank.

If \mathbf{A} and \mathbf{B} are full column rank and \mathbf{C} does not have collinear vectors, then this algorithm may be initialized by means of a (generalized) EVD, as explained in the proof of [11, Theorem 4.1].

2.3. Numerical experiments. We generate tensors $\tilde{\mathcal{T}} \in \mathbb{C}^{5 \times 6 \times 5}$ in the following way:

$$(2.5) \quad \tilde{\mathcal{T}} = \frac{\mathcal{T}}{\|\mathcal{T}\|} + \sigma_N \frac{\mathcal{N}}{\|\mathcal{N}\|},$$

in which \mathcal{T} can be decomposed as in (2.1). We consider $R = 3$ rank- $(2, 2, 1)$ terms, i.e., $\mathbf{A}_r \in \mathbb{C}^{5 \times 2}$, $\mathbf{B}_r \in \mathbb{C}^{6 \times 2}$, $\mathbf{C}_r \in \mathbb{C}^{6 \times 1}$, $1 \leq r \leq 3$. The decomposition of \mathcal{T} is essentially unique by [11, Theorem 4.4]. The second term in (2.5) is a noise term. The entries of \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathcal{N} are drawn from a zero-mean unit-variance Gaussian distribution. The parameter σ_N controls the noise level.

A Monte Carlo experiment consisting of 200 runs was carried out. The algorithm was initialized with three random starting values.

The accuracy is measured in terms of the relative error $e = \|\mathbf{C} - \hat{\mathbf{C}}\| / \|\mathbf{C}\|$, in which $\hat{\mathbf{C}}$ is the estimate of \mathbf{C} , optimally ordered and scaled. The median results are plotted in Figure 2.1. We plot the median instead of the mean because, in some of the

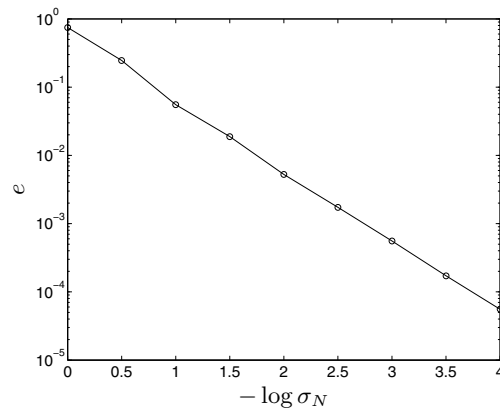


FIG. 2.1. Median relative error obtained in the first experiment in section 2.3.

runs, the convergence became too slow for the algorithm to find a sufficiently accurate estimate in a reasonable time.

In a second experiment, we generate tensors $\tilde{\mathcal{T}} \in \mathbb{C}^{10 \times 10 \times 10}$ as in (2.5). We consider $R = 5$ rank-(2,2,1) terms, i.e., $\mathbf{A}_r \in \mathbb{C}^{10 \times 2}$, $\mathbf{B}_r \in \mathbb{C}^{10 \times 2}$, $\mathbf{C}_r \in \mathbb{C}^{10 \times 1}$, $1 \leq r \leq 5$. The five rank-(2,2,1) terms are scaled such that their Frobenius norm equals 1, 3.25, 5.5, 7.75, and 10, respectively. The fact that there is a difference of 20 dB between the strongest and the weakest term makes this problem quite hard. The decomposition of \mathcal{T} is essentially unique by [11, Theorem 4.1]. In Figure 2.2 we show the median accuracy obtained when the algorithm in Table 2.1 is initialized (i) by means of a (generalized) EVD, as explained in the proof of [11, Theorem 4.1], and (ii) by means of a random starting value. It is clear that the global optimum is not found when the algorithm is initialized randomly. However, the initialization by means of a (generalized) EVD does lead to the global solution when the signal-to-noise ratio (SNR) is sufficiently high. As a matter of fact, the (generalized) EVD yields the exact solution when the data are noise-free.

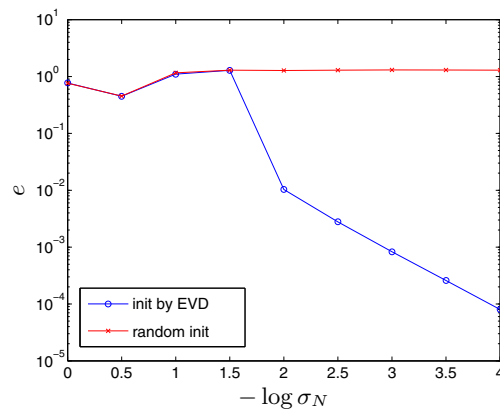


FIG. 2.2. Median relative error obtained in the second experiment in section 2.3.

3. Decomposition in rank-(L, M, N) terms.

3.1. Definition.

DEFINITION 3.1. A decomposition of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ in a sum of rank- (L, M, N) terms is a decomposition of \mathcal{T} of the form

$$(3.1) \quad \mathcal{T} = \sum_{r=1}^R \mathcal{D}_r \bullet_1 \mathbf{A}_r \bullet_2 \mathbf{B}_r \bullet_3 \mathbf{C}_r,$$

in which $\mathcal{D}_r \in \mathbb{K}^{L \times M \times N}$ are full rank- (L, M, N) and in which $\mathbf{A}_r \in \mathbb{K}^{I \times L}$ (with $I \geq L$), $\mathbf{B}_r \in \mathbb{K}^{J \times M}$ (with $J \geq M$), and $\mathbf{C}_r \in \mathbb{K}^{K \times N}$ (with $K \geq N$) are full column rank, $1 \leq r \leq R$.

Define partitioned matrices $\mathbf{A} = [\mathbf{A}_1 \dots \mathbf{A}_R]$, $\mathbf{B} = [\mathbf{B}_1 \dots \mathbf{B}_R]$, and $\mathbf{C} = [\mathbf{C}_1 \dots \mathbf{C}_R]$. In terms of the standard matrix representations of \mathcal{T} , (3.1) can be written as

$$(3.2) \quad \mathbf{T}_{JK \times I} = (\mathbf{B} \odot \mathbf{C}) \cdot \text{blockdiag}((\mathcal{D}_1)_{MN \times L}, \dots, (\mathcal{D}_R)_{MN \times L}) \cdot \mathbf{A}^T,$$

$$(3.3) \quad \mathbf{T}_{KI \times J} = (\mathbf{C} \odot \mathbf{A}) \cdot \text{blockdiag}((\mathcal{D}_1)_{NL \times M}, \dots, (\mathcal{D}_R)_{NL \times M}) \cdot \mathbf{B}^T,$$

$$(3.4) \quad \mathbf{T}_{IJ \times K} = (\mathbf{A} \odot \mathbf{B}) \cdot \text{blockdiag}((\mathcal{D}_1)_{LM \times N}, \dots, (\mathcal{D}_R)_{LM \times N}) \cdot \mathbf{C}^T.$$

In terms of the $(IJK \times 1)$ vector representation of \mathcal{T} , the decomposition can be written as

$$(3.5) \quad \mathbf{t}_{IJK} = (\mathbf{A} \odot \mathbf{B} \odot \mathbf{C}) \cdot \begin{pmatrix} (\mathcal{D}_1)_{LMN} \\ \vdots \\ (\mathcal{D}_R)_{LMN} \end{pmatrix}.$$

3.2. Algorithm. The decomposition in rank- (L, M, N) terms is quadrilinear in its factors \mathbf{A} , \mathbf{B} , \mathbf{C} , and \mathcal{D} . Hence, the conditional update of \mathbf{A} , given \mathbf{B} , \mathbf{C} , and \mathcal{D} , is a linear least squares problem. The same holds for conditional updates of \mathbf{B} , \mathbf{C} , and \mathcal{D} . The update rules follow directly from (3.2)–(3.5). The algorithm is outlined in Table 3.1. This algorithm is a generalization of the algorithm in [43] for the computation of the best rank- (L, M, N) approximation of a given tensor. The matrices $(\mathbf{B} \odot \mathbf{C}) \cdot \text{blockdiag}((\mathcal{D}_1)_{MN \times L}, \dots, (\mathcal{D}_R)_{MN \times L})$, $(\mathbf{C} \odot \mathbf{A}) \cdot \text{blockdiag}((\mathcal{D}_1)_{NL \times M}, \dots, (\mathcal{D}_R)_{NL \times M})$, and $(\mathbf{A} \odot \mathbf{B}) \cdot \text{blockdiag}((\mathcal{D}_1)_{LM \times N}, \dots, (\mathcal{D}_R)_{LM \times N})$ have to have at least as many rows as columns and have to be full column rank.

The order of the updates in Table 3.1 is not mandatory. We have observed in numerical experiments that it is often advantageous to alternate between a few updates of \mathbf{A} and \mathcal{D} , then alternate between a few updates of \mathbf{B} and \mathcal{D} , and so on.

3.3. Numerical experiments. We generate tensors $\tilde{\mathcal{T}} \in \mathbb{C}^{5 \times 5 \times 7}$ as in (2.5). The tensors \mathcal{T} can now be decomposed as in (3.1). We consider $R = 2$ terms characterized by $\mathbf{A}_r \in \mathbb{C}^{5 \times 2}$, $\mathbf{B}_r \in \mathbb{C}^{5 \times 2}$, $\mathbf{C}_r \in \mathbb{C}^{7 \times 3}$, and $\mathcal{D}_r \in \mathbb{C}^{2 \times 2 \times 3}$, $1 \leq r \leq 2$. The entries of \mathbf{A}_r , \mathbf{B}_r , \mathbf{C}_r , \mathcal{D}_r , and \mathcal{N} are drawn from a zero-mean unit-variance Gaussian distribution. The decomposition of \mathcal{T} is essentially unique by [11, Theorem 5.1].

A Monte Carlo experiment consisting of 200 runs was carried out. The algorithm was initialized with three random starting values.

The accuracy is measured in terms of the relative error $e = \|\mathbf{C} - \hat{\mathbf{C}}\| / \|\mathbf{C}\|$, in which $\hat{\mathbf{C}}$ is the estimate of \mathbf{C} , of which the submatrices are optimally ordered and multiplied from the right by a (3×3) matrix. The median results are plotted in Figure 3.1.

Next, we check what happens if the algorithm in Table 3.1 is used for the computation of the decomposition in rank- $(L, L, 1)$ terms. In this case, the tensors \mathcal{D}_r are of

TABLE 3.1
ALS algorithm for decomposition in rank- (L, M, N) terms.

- Initialize \mathbf{B} , \mathbf{C} , \mathcal{D}

- Iterate until convergence:

1. Update \mathbf{A} :

$$\tilde{\mathbf{A}} = \left[\text{blockdiag}((\mathcal{D}_1)_{MN \times L}^\dagger, \dots, (\mathcal{D}_R)_{MN \times L}^\dagger) \cdot (\mathbf{B} \odot \mathbf{C})^\dagger \cdot \mathbf{T}_{JK \times I} \right]^T$$

For $r = 1, \dots, R$: QR-factorization: $\tilde{\mathbf{A}}_r = \mathbf{Q}\mathbf{R}$, $\mathbf{A}_r \leftarrow \mathbf{Q}$

2. Update \mathbf{B} :

$$\tilde{\mathbf{B}} = \left[\text{blockdiag}((\mathcal{D}_1)_{NL \times M}^\dagger, \dots, (\mathcal{D}_R)_{NL \times M}^\dagger) \cdot (\mathbf{C} \odot \mathbf{A})^\dagger \cdot \mathbf{T}_{KI \times J} \right]^T$$

For $r = 1, \dots, R$: QR-factorization: $\tilde{\mathbf{B}}_r = \mathbf{Q}\mathbf{R}$, $\mathbf{B}_r \leftarrow \mathbf{Q}$

3. Update \mathbf{C} :

$$\tilde{\mathbf{C}} = \left[\text{blockdiag}((\mathcal{D}_1)_{LM \times N}^\dagger, \dots, (\mathcal{D}_R)_{LM \times N}^\dagger) \cdot (\mathbf{A} \odot \mathbf{B})^\dagger \cdot \mathbf{T}_{IJ \times K} \right]^T$$

For $r = 1, \dots, R$: QR-factorization: $\tilde{\mathbf{C}}_r = \mathbf{Q}\mathbf{R}$, $\mathbf{C}_r \leftarrow \mathbf{Q}$

4. Update \mathcal{D} :

$$\begin{pmatrix} (\mathcal{D}_1)_{LMN} \\ \vdots \\ (\mathcal{D}_R)_{LMN} \end{pmatrix} \leftarrow (\mathbf{A} \odot \mathbf{B} \odot \mathbf{C})^\dagger \cdot \mathbf{t}_{IJK}$$

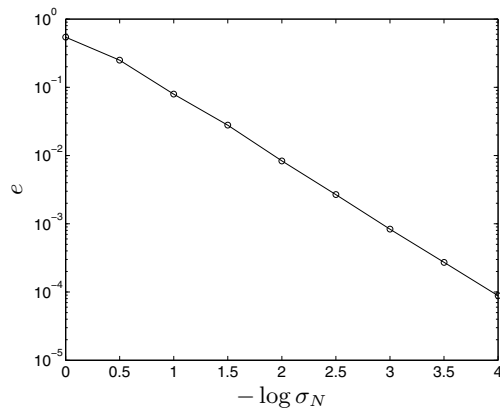


FIG. 3.1. Median relative error obtained in the first experiment in section 3.3.

dimension $(L \times L \times 1)$. The data are generated as in the first experiment in section 2.3. We compare three algorithms: (i) the algorithm of Table 2.1, which we denote as Alg $(L, L, 1)$, (ii) the algorithm of Table 3.1, which we denote as Alg (L, M, N) , and (iii) a variant of the algorithm of Table 3.1 in which one alternates between a few updates of \mathbf{A} and \mathbf{D} , then alternates between a few updates of \mathbf{B} and \mathbf{D} , and so on, as explained at the end of section 3.2. The latter algorithm is denoted as Alg $(L, M, N)^*$. The inner iteration is terminated if the Frobenius norm of the difference between two consecutive approximations of \mathcal{T} drops below $1e-6$, with a maximum of 10 inner iterations. We observed that most of the time not more than two or three inner iterations were carried out. We computed the results for one and two random initializations, respectively.

The median results for accuracy and computation time are plotted in Figures 3.2 and 3.3, respectively. From Figure 3.2 it is clear that Alg (L, M, N) does not find the global optimum if it is initialized only once. One should perform inner iterations, or initialize several times. However, both remedies increase the computational cost, as is clear from Figure 3.3. Given that Alg (L, M, N) is by itself more expensive than Alg $(L, L, 1)$, we conclude that it is advantageous to compute the decomposition in rank- $(L, L, 1)$ terms by means of Alg $(L, L, 1)$.

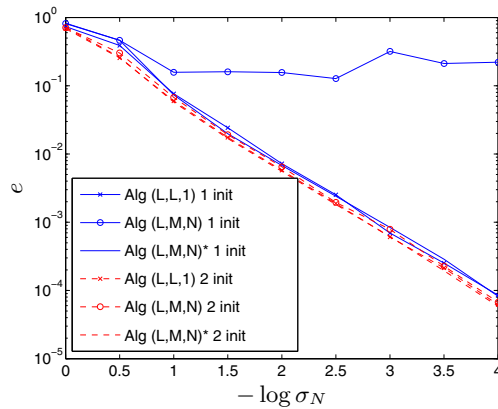


FIG. 3.2. Median relative error obtained in the second experiment in section 3.3.

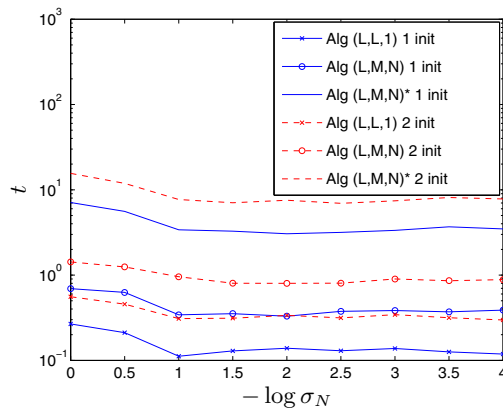


FIG. 3.3. Median computation time in the second experiment in section 3.3.

4. Type-2 decomposition in rank- (L, M, \cdot) terms.

4.1. Definition.

DEFINITION 4.1. A type-2 decomposition of a tensor $\mathcal{T} \in \mathbb{K}^{I \times J \times K}$ in a sum of rank- (L, M, \cdot) terms is a decomposition of \mathcal{T} of the form

$$(4.1) \quad \mathcal{T} = \sum_{r=1}^R \mathcal{C}_r \bullet_1 \mathbf{A}_r \bullet_2 \mathbf{B}_r,$$

in which $\mathcal{C}_r \in \mathbb{K}^{L \times M \times K}$ (with mode-1 rank equal to L and mode-2 rank equal to M), and in which $\mathbf{A}_r \in \mathbb{K}^{I \times L}$ (with $I \geq L$) and $\mathbf{B}_r \in \mathbb{K}^{J \times M}$ (with $J \geq M$) are full column rank, $1 \leq r \leq R$.

TABLE 4.1
 ALS algorithm for type-2 decomposition in rank- (L, M, \cdot) terms.

- Initialize \mathbf{B} , $\mathcal{C}_1, \dots, \mathcal{C}_R$

- Iterate until convergence:

1. Update \mathbf{A} :

$$\tilde{\mathbf{A}} = \left\{ [(\mathcal{C}_1 \bullet_2 \mathbf{B}_1)_{JK \times L} \dots (\mathcal{C}_R \bullet_2 \mathbf{B}_R)_{JK \times L}]^\dagger \cdot \mathbf{T}_{JK \times I} \right\}^T$$

For $r = 1, \dots, R$: QR-factorization: $\tilde{\mathbf{A}}_r = \mathbf{Q}_r \mathbf{R}$, $\mathbf{A}_r \leftarrow \mathbf{Q}$

2. Update \mathbf{B} :

$$\tilde{\mathbf{B}} = \left\{ [(\mathcal{C}_1 \bullet_1 \mathbf{A}_1)_{KI \times M} \dots (\mathcal{C}_R \bullet_1 \mathbf{A}_R)_{KI \times M}]^\dagger \cdot \mathbf{T}_{KI \times J} \right\}^T$$

For $r = 1, \dots, R$: QR-factorization: $\tilde{\mathbf{B}}_r = \mathbf{Q}_r \mathbf{R}$, $\mathbf{B}_r \leftarrow \mathbf{Q}$

3. Update $\mathcal{C}_1, \dots, \mathcal{C}_R$:

$$\begin{pmatrix} (\mathcal{C}_1)_{(LM \times K)} \\ \vdots \\ (\mathcal{C}_R)_{(LM \times K)} \end{pmatrix} \leftarrow (\mathbf{A} \odot \mathbf{B})^\dagger \cdot \mathbf{T}_{IJ \times K}$$

Define partitioned matrices $\mathbf{A} = [\mathbf{A}_1 \dots \mathbf{A}_R]$ and $\mathbf{B} = [\mathbf{B}_1 \dots \mathbf{B}_R]$. In terms of the standard matrix representations of \mathcal{T} , (4.1) can be written as

$$(4.2) \quad \mathbf{T}_{IJ \times K} = (\mathbf{A} \odot \mathbf{B}) \cdot \begin{pmatrix} (\mathcal{C}_1)_{(LM \times K)} \\ \vdots \\ (\mathcal{C}_R)_{(LM \times K)} \end{pmatrix},$$

$$(4.3) \quad \mathbf{T}_{JK \times I} = [(\mathcal{C}_1 \bullet_2 \mathbf{B}_1)_{JK \times L} \dots (\mathcal{C}_R \bullet_2 \mathbf{B}_R)_{JK \times L}] \cdot \mathbf{A}^T,$$

$$(4.4) \quad \mathbf{T}_{KI \times J} = [(\mathcal{C}_1 \bullet_1 \mathbf{A}_1)_{KI \times M} \dots (\mathcal{C}_R \bullet_1 \mathbf{A}_R)_{KI \times M}] \cdot \mathbf{B}^T.$$

4.2. Algorithm. Since the type-2 decomposition in rank- (L, M, \cdot) terms is trilinear in \mathbf{A} , \mathbf{B} , and \mathcal{C} , an ALS algorithm consists of successive linear least squares problems. The update rules for \mathbf{A} , \mathbf{B} , and \mathcal{C} follow directly from (4.3), (4.4), and (4.2), respectively. The algorithm is outlined in Table 4.1. The matrices $\mathbf{A} \odot \mathbf{B}$, $[(\mathcal{C}_1 \bullet_2 \mathbf{B}_1)_{JK \times L} \dots (\mathcal{C}_R \bullet_2 \mathbf{B}_R)_{JK \times L}]$, and $[(\mathcal{C}_1 \bullet_1 \mathbf{A}_1)_{KI \times M} \dots (\mathcal{C}_R \bullet_1 \mathbf{A}_R)_{KI \times M}]$ have to have at least as many rows as columns and have to be full column rank.

4.3. Numerical experiment. We generate tensors $\tilde{\mathcal{T}} \in \mathbb{C}^{5 \times 6 \times 6}$ as in (2.5). The tensors \mathcal{T} can now be decomposed as in (4.1). We consider $R = 3$ terms characterized by $\mathbf{A}_r \in \mathbb{C}^{5 \times 2}$, $\mathbf{B}_r \in \mathbb{C}^{6 \times 2}$, and $\mathcal{C}_r \in \mathbb{C}^{2 \times 2 \times 6}$, $1 \leq r \leq 3$. The entries of \mathbf{A}_r , \mathbf{B}_r , \mathcal{C}_r , and \mathcal{N} are drawn from a zero-mean unit-variance Gaussian distribution. The decomposition of \mathcal{T} is essentially unique by [11, Example 3].

A Monte Carlo experiment consisting of 200 runs was carried out. The algorithm was initialized with three random starting values.

The accuracy is measured in terms of the relative error $e = \|\mathbf{B} - \hat{\mathbf{B}}\| / \|\mathbf{B}\|$, in which $\hat{\mathbf{B}}$ is the estimate of \mathbf{B} , of which the submatrices are optimally ordered and multiplied from the right by a (2×2) matrix. The median results are plotted in Figure 4.1.

5. Degeneracy. In the real field, PARAFAC algorithms sometimes show the following behavior. The norm of individual terms in (1.12) goes to infinity, but these terms almost completely cancel each other, such that the overall error continues to

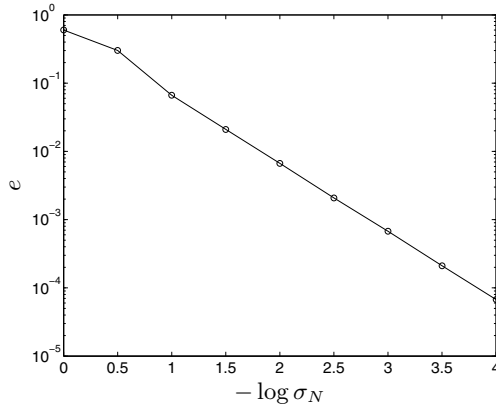


FIG. 4.1. Median relative error obtained in the experiment in section 4.3.

decrease. This phenomenon is known as “degeneracy” [16, 25, 27]. It is caused by the fact that for $\tilde{R} \geq 2$, the set

$$U_{\tilde{R}} = \{T \in \mathbb{R}^{I \times J \times K} \mid \text{rank}(T) \leq \tilde{R}\}$$

is not closed [12, 25, 38]. The set of tensors that are the sum of at most $\tilde{R} \geq 2$ rank- (L, M, N) terms,

$$V_{\tilde{R}} = \{T \in \mathbb{R}^{I \times J \times K} \mid T \text{ decomposable as in (3.1), with } R \leq \tilde{R} \text{ and } \tilde{R} \geq 2\},$$

is not closed either. We give an explicit example that is a straightforward generalization of the example given for PARAFAC in [12]. Analogous results hold for the other types of block term decompositions.

Let $\mathbf{I}_1 \in \mathbb{R}^{4 \times 2}$ and $\mathbf{I}_2 \in \mathbb{R}^{4 \times 2}$ consist of the first (resp., last) two columns of $\mathbf{I}_{4 \times 4}$. Consider the tensor $\mathcal{E} \in \mathbb{R}^{2 \times 2 \times 2}$ defined by

$$\begin{aligned} e_{111} &= e_{221} = e_{122} = 1, \\ e_{121} &= e_{211} = e_{112} = e_{212} = e_{222} = 0. \end{aligned}$$

This tensor is rank-3 in \mathbb{R} ; see [5, pp. 21–22] and [18, section 3]. Now define $T \in \mathbb{R}^{4 \times 4 \times 4}$ as follows:

$$\begin{aligned} T(1 : 2, 1 : 2, 1 : 2) &= T(3 : 4, 3 : 4, 1 : 2) = T(1 : 2, 3 : 4, 3 : 4) = \mathcal{E}, \\ T(3 : 4, 1 : 2, 1 : 2) &= T(3 : 4, 1 : 2, 3 : 4) = T(1 : 2, 3 : 4, 1 : 2) \\ &= T(1 : 2, 1 : 2, 3 : 4) = T(3 : 4, 3 : 4, 3 : 4) = \mathcal{O}_{2 \times 2 \times 2}. \end{aligned}$$

This tensor can be decomposed in three rank- $(2, 2, 2)$ terms:

$$(5.1) \quad T = \mathcal{E} \bullet_1 \mathbf{I}_1 \bullet_2 \mathbf{I}_1 \bullet_3 \mathbf{I}_1 + \mathcal{E} \bullet_1 \mathbf{I}_1 \bullet_2 \mathbf{I}_2 \bullet_3 \mathbf{I}_2 + \mathcal{E} \bullet_1 \mathbf{I}_2 \bullet_2 \mathbf{I}_2 \bullet_3 \mathbf{I}_1.$$

However, it cannot be decomposed in two rank- $(2, 2, 2)$ terms. We prove this by contradiction. Assume that a decomposition in two rank- $(2, 2, 2)$ terms does exist:

$$(5.2) \quad T = \mathcal{D}_1 \bullet_1 \mathbf{A}_1 \bullet_2 \mathbf{B}_1 \bullet_3 \mathbf{C}_1 + \mathcal{D}_2 \bullet_1 \mathbf{A}_2 \bullet_2 \mathbf{B}_2 \bullet_3 \mathbf{C}_2.$$

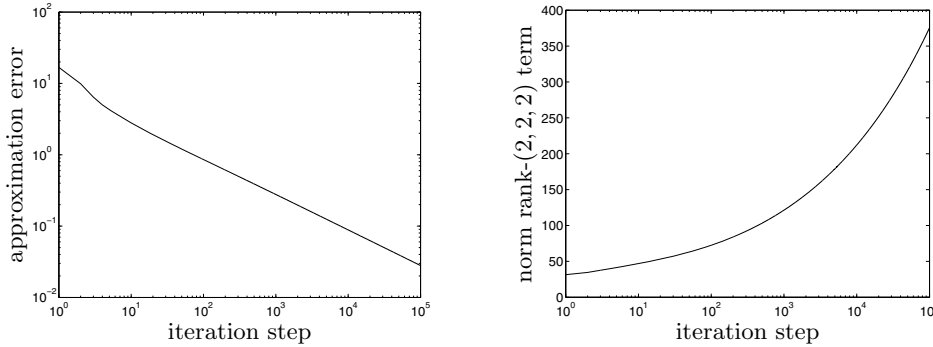


FIG. 5.1. Visualization of the degeneracy in Example 1. Left: evolution of the approximation error. Right: evolution of the norm of the rank-(2, 2, 2) terms.

We can normalize this decomposition such that the first row of $\mathbf{C} = [\mathbf{C}_1 \ \mathbf{C}_2]$ is equal to $(1 \ 0 \ 1 \ 0)$, and $\mathcal{D}_1 \bullet_3 (1 \ 0) = \mathcal{D}_2 \bullet_3 (1 \ 0) = \mathbf{I}_{2 \times 2}$. Define $\mathbf{A} = [\mathbf{A}_1 \ \mathbf{A}_2]$ and $\mathbf{B} = [\mathbf{B}_1 \ \mathbf{B}_2]$. We have $\mathbf{T}_{I \times J, 1} = \mathbf{I}_{4 \times 4} = \mathbf{A} \cdot \mathbf{B}^T$. Hence, \mathbf{A} and \mathbf{B} are nonsingular. Define $\mathbf{X} = [\mathbf{x}_1 \ \dots \ \mathbf{x}_4] = \mathbf{A}^{-1}$ and $\mathbf{Y} = [\mathbf{y}_1 \ \dots \ \mathbf{y}_4] = \mathbf{B}^{-1}$. From (5.2) we have that all the $(I \times J)$ slices of $\tilde{\mathcal{T}} = \mathcal{T} \bullet_1 \mathbf{X} \bullet_2 \mathbf{Y}$ are block-diagonal, consisting of two (2×2) blocks. From the definition of \mathcal{T} , we have that $\tilde{\mathbf{T}}_{I \times J, 4} = \mathbf{x}_1 \cdot \mathbf{y}_4^T$. From the block-diagonality of this rank-1 matrix follows that, without loss of generality, we can assume that the third and fourth entries of \mathbf{x}_1 and \mathbf{y}_4 are zero. Further, we have that $\tilde{\mathbf{T}}_{I \times J, 3} = \mathbf{x}_1 \cdot \mathbf{y}_3^T + \mathbf{x}_2 \cdot \mathbf{y}_4^T$. From the block-diagonality of this rank-2 matrix and the structure of \mathbf{x}_1 and \mathbf{y}_4 follows that the third and fourth entries of \mathbf{x}_2 and \mathbf{y}_3 are zero. Finally, we have that $\tilde{\mathbf{T}}_{I \times J, 2} = \mathbf{x}_1 \cdot \mathbf{y}_2^T + \mathbf{x}_3 \cdot \mathbf{y}_4^T$. From the block-diagonality of this rank-2 matrix and the structure of \mathbf{x}_1 and \mathbf{y}_4 follows that the third and fourth entries of \mathbf{x}_3 and \mathbf{y}_2 are zero. We have a contradiction with the fact that \mathbf{X} and \mathbf{Y} are full rank. We conclude that \mathcal{T} cannot be decomposed in a sum of two rank-(2, 2, 2) terms.

On the other hand, there does not exist an approximation $\hat{\mathcal{T}}$, consisting of a sum of two rank-(2, 2, 2) terms, that is optimal in the sense of minimizing the error $\|\mathcal{T} - \hat{\mathcal{T}}\|$. Define $\hat{\mathcal{T}}_n$ as follows, for increasing integer values of n :

$$(5.3) \quad \hat{\mathcal{T}}_n = \mathcal{E} \bullet_1 \mathbf{I}_1 \bullet_2 (\mathbf{I}_1 - n\mathbf{I}_2) \bullet_3 \mathbf{I}_1 + \mathcal{E} \bullet_1 \left(\mathbf{I}_1 + \frac{1}{n}\mathbf{I}_2 \right) \bullet_2 (n\mathbf{I}_2) \bullet_3 \left(\mathbf{I}_1 + \frac{1}{n}\mathbf{I}_2 \right).$$

We have

$$\hat{\mathcal{T}}_n = \mathcal{T} + \frac{1}{n} \mathcal{E} \bullet_1 \mathbf{I}_2 \bullet_2 \mathbf{I}_2 \bullet_3 \mathbf{I}_2.$$

Clearly, $\|\mathcal{T} - \hat{\mathcal{T}}_n\|$ goes to zero as n goes to infinity. However, at the same time the norms of the individual terms in (5.3) go to infinity. This shows that degeneracy also exists for block term decompositions.

Example 1. Figure 5.1 shows a typical degeneracy. We constructed a tensor \mathcal{T} as in (5.1) with \mathcal{E} , however, defined by

$$\begin{aligned} e_{111} &= -14 & e_{121} &= -4 & e_{211} &= 6 & e_{221} &= 7, \\ e_{112} &= 8 & e_{122} &= 13 & e_{212} &= 7 & e_{222} &= 7. \end{aligned}$$

The eigenvalues of $\mathbf{E}_{I \times J, 1} \cdot \mathbf{E}_{I \times J, 2}^{-1}$ are complex, so \mathcal{E} is rank-3 in \mathbb{R} . The algorithm in Table 3.1 was used to approximate \mathcal{T} by a sum of two rank-(2, 2, 2) terms. The

left plot shows a monotonous decrease of the approximation error. The right plot shows the evolution of the norm of the rank-(2, 2, 2) terms (the curves for both terms coincide).

6. Conclusion. We have derived ALS algorithms for the different block term decompositions that were introduced in [11]. ALS is actually a very simple approach. For PARAFAC, combining ALS with (exact) line search improves the performance [33]. An other technique that has proved useful for PARAFAC is the Levenberg–Marquardt type optimization [39]. When the tensor is tall in one mode, PARAFAC may often be computed by means of a simultaneous matrix decomposition [10]. Since the submission of this manuscript, we have been studying generalizations of such methods to block term decompositions [28, 29, 30, 31].

Acknowledgment. The authors wish to thank A. Stegeman (Heijmans Institute, The Netherlands) for proofreading an early version of the manuscript. A large part of this research was carried out when L. De Lathauwer and D. Nion were with the ETIS lab of the French Centre National de la Recherche Scientifique (C.N.R.S.).

REFERENCES

- [1] G. BOUTRY, M. ELAD, G.H. GOLUB, AND P. MILANFAR, *The generalized eigenvalue problem for nonsquare pencils using a minimal perturbation approach*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 582–601.
- [2] D. BURDICK, X. TU, L. MCGOWN, AND D. MILLICAN, *Resolution of multicomponent fluorescent mixtures by analysis of the excitation-emission-frequency array*, J. Chemometrics, 4 (1990), pp. 15–28.
- [3] J. CARROLL AND J. CHANG, *Analysis of individual differences in multidimensional scaling via an N -way generalization of “Eckart-Young” decomposition*, Psychometrika, 9 (1970), pp. 267–283.
- [4] P. COMON, G. GOLUB, L.-H. LIM, AND B. MOURRAIN, *Symmetric tensors and symmetric tensor rank*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1254–1279.
- [5] L. DE LATHAUWER, *Signal Processing Based on Multilinear Algebra*, Ph.D. thesis, K.U.Leuven, Belgium, 1997.
- [6] L. DE LATHAUWER, B. DE MOOR, AND J. VANDEWALLE, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278.
- [7] L. DE LATHAUWER, B. DE MOOR, AND J. VANDEWALLE, *On the best rank-1 and rank- (R_1, R_2, \dots, R_N) approximation of higher-order tensors*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1324–1342.
- [8] L. DE LATHAUWER AND J. VANDEWALLE, *Dimensionality reduction in higher-order signal processing and rank- (R_1, R_2, \dots, R_N) reduction in multilinear algebra*, Linear Algebra Appl., 391 (2004), pp. 31–55.
- [9] L. DE LATHAUWER, B. DE MOOR, AND J. VANDEWALLE, *Computation of the Canonical Decomposition by means of a simultaneous generalized Schur decomposition*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 295–327.
- [10] L. DE LATHAUWER, *A link between the Canonical Decomposition in multilinear algebra and simultaneous matrix diagonalization*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 642–666.
- [11] L. DE LATHAUWER, *Decompositions of a higher-order tensor in block terms—Part II: Definitions and uniqueness*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1033–1066.
- [12] V. DE SILVA AND L.-H. LIM, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1084–1127.
- [13] M. ELAD, P. MILANFAR, AND G.H. GOLUB, *Shape from moments—an estimation theory perspective*, IEEE Trans. Signal Process., 52 (2004), pp. 1814–1829.
- [14] L. ELDÉN AND B. SAVAS, *A Newton–Grassmann Method for Computing the Best Multi-Linear Rank- (r_1, r_2, r_3) Approximation of a Tensor*, Tech. report LITH-MAT-R-2007-6-SE, Department of Mathematics, Linköping University, 2007.
- [15] R.A. HARSHMAN, *Foundations of the PARAFAC Procedure: Model and Conditions for an “Explanatory” Multi-Mode Factor Analysis*, UCLA Working Papers in Phonetics, 16 (1970), pp. 1–84.

- [16] R.A. HARSHMAN AND M.E. LUNDY, *Data preprocessing and the extended Parafac model*, in Research Methods for Multimode Data Analysis, H.G. Law, C.W. Snyder, J.A. Hattie, and R.P. McDonald, eds., Praeger, New York, 1984, pp. 216–284.
- [17] M. ISHTEVA, L. DE LATHAUWER, P.-A. ABSIL, AND S. VAN HUFFEL, *Dimensionality reduction for higher-order tensors: algorithms and applications*, Int. J. Pure Appl. Math., 42 (2008), pp. 337–343.
- [18] J. JA'JA', *Optimal evaluation of bilinear forms*, SIAM J. Comput., 8 (1979), pp. 443–462.
- [19] H. KIERS, *Towards a standardized notation and terminology in multiway analysis*, J. Chemometrics, 14 (2000), pp. 105–122.
- [20] E. KOFIDIS AND P.A. REGALIA, *On the best rank-1 approximation of higher-order supersymmetric tensors*, SIAM J. Matrix Anal. Appl., 23 (2002), pp. 863–884.
- [21] P.M. KROONENBERG AND J. DE LEEUW, *Principal component analysis of three-mode data by means of alternating least squares algorithms*, Psychometrika, 45 (1980), pp. 69–97.
- [22] P.M. KROONENBERG, *Three-mode principal component analysis: illustrated with an example from attachment theory*, in Research Methods for Multimode Data Analysis, H.G. Law, C.W. Snyder, J.A. Hattie, and R.P. McDonald, eds., Praeger, New York, 1984, pp. 64–103.
- [23] P.M. KROONENBERG, *Applied Multiway Data Analysis*, Wiley, New York, 2008.
- [24] J.B. KRUSKAL, *Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics*, Linear Algebra Appl., 18 (1977), pp. 95–138.
- [25] J.B. KRUSKAL, R.A. HARSHMAN, AND M.E. LUNDY, *How 3-MFA data can cause degenerate PARAFAC solutions, among other relationships*, in Multiway Data Analysis, R. Coppi and S. Bolasco, eds., North-Holland, Amsterdam, 1989, pp. 115–122.
- [26] S.E. LEURGANS, R.T. ROSS, AND R.B. ABEL, *A decomposition for three-way arrays*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1064–1083.
- [27] B.C. MITCHELL AND D.S. BURDICK, *Slowly converging Parafac sequences: Swamps and two-factor degeneracies*, J. Chem., 8 (1994), pp. 155–168.
- [28] D. NION AND L. DE LATHAUWER, *Levenberg-Marquardt computation of the block factor model for blind multi-user access in wireless communications*, Proceedings of the 14th European Signal Processing Conference (Eusipco 2006), Florence, Italy, 2006.
- [29] D. NION AND L. DE LATHAUWER, *A tensor-based blind DS-CDMA receiver using simultaneous matrix diagonalization*, Proceedings of the VIII IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2007), Helsinki, Finland, 2007.
- [30] D. NION AND L. DE LATHAUWER, *Block component model based blind DS-CDMA receivers*, IEEE Trans. Signal. Process., to appear.
- [31] D. NION AND L. DE LATHAUWER, *An enhanced line search scheme for complex-valued tensor decompositions. Application in DS-CDMA.*, Signal Process., 88 (2008), pp. 749–755.
- [32] 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. Graphical Statist., 8 (1999), pp. 854–888.
- [33] M. RAJIH, P. COMON, AND R.A. HARSHMAN, *Enhanced line search: a novel method to accelerate parafac*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1128–1147.
- [34] C.R. RAO AND S.K. MITRA, *Generalized Inverse of Matrices and Its Applications*, Wiley, New York, 1971.
- [35] E. SANCHEZ AND B.R. KOWALSKI, *Tensorial resolution: A direct trilinear decomposition*, J. Chemometrics, 4 (1990), pp. 29–45.
- [36] R. SANDS AND F. YOUNG, *Component models for three-way data: An alternating least squares algorithm with optimal scaling features*, Psychometrika, 45 (1980), pp. 39–67.
- [37] A. SMILDE, R. BRO, AND P. GELADI, *Multi-way Analysis. Applications in the Chemical Sciences*, Wiley, Chichester, UK, 2004.
- [38] 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., 30 (2008), pp. 988–1007.
- [39] G. TOMASI AND R. BRO, *A comparison of algorithms for fitting the PARAFAC model*, Comp. Stat. Data Anal., 50 (2006), pp. 1700–1734.
- [40] L.R. TUCKER, *The extension of factor analysis to three-dimensional matrices*, in Contributions to Mathematical Psychology, H. Gulliksen and N. Frederiksen, eds., Holt, Rinehart & Winston, New York, 1964, pp. 109–127.
- [41] L.R. TUCKER, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 31 (1966), pp. 279–311.
- [42] S.A. VOROBYOV, Y. RONG, N.D. SIDIROPOULOS, AND A.B. GERSHMAN, *Robust iterative fitting of multilinear models*, IEEE Trans. Signal Process., 53 (2005), pp. 2678–2689.

- [43] J. WEESIE AND H. VAN HOUWELINGEN, *GEPCAM Users' Manual: Generalized Principal Components Analysis with Missing Values*, Tech. report, Institute of Mathematical Statistics, University of Utrecht, Netherlands, 1983.
- [44] T. ZHANG AND G.H. GOLUB, *Rank-one approximation to high order tensors*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 534–550.