SIAM J. MATRIX ANAL. APPL.
Vol. 21, No. 4, pp. 1324–1342

ON THE BEST RANK-1 AND RANK-$(R_1, R_2, \ldots, R_N)$ APPROXIMATION OF HIGHER-ORDER TENSORS*

LIEVEN DE LATHAUWER†, BART DE MOOR†, AND JOOS VANDEWALLE†

Abstract. In this paper we discuss a multilinear generalization of the best rank-R approximation problem for matrices, namely, the approximation of a given higher-order tensor, in an optimal least-squares sense, by a tensor that has prespecified column rank value, row rank value, etc. For matrices, the solution is conceptually obtained by truncation of the singular value decomposition (SVD); however, this approach does not have a straightforward multilinear counterpart. We discuss higher-order generalizations of the power method and the orthogonal iteration method.

Key words. multilinear algebra, singular value decomposition, higher-order tensor, rank reduction

AMS subject classifications. 15A18, 15A69

PII. S0895479898346995

1. Introduction.

Multilinear algebra is the algebra of higher-order tensors, which are the higher-order equivalents of vectors (first order) and matrices (second order), i.e., quantities of which the elements are addressed by more than two indices. Multilinear algebra is gaining more and more interest, largely due to its applications in the context of higher-order statistics (HOS) [17, 18, 2, 4, 5, 7].

Rank-related issues in multilinear algebra are thoroughly different from their matrix counterparts. Let us first introduce some definitions. A rank-1 tensor is a tensor that consists of the outer product of a number of vectors. For an Nth-order tensor $\mathcal{A}$ and N vectors $U^{(1)}$, $U^{(2)}$, $\ldots$, $U^{(N)}$, this means that $a_{i_1 i_2 \cdots i_N} = u^{(1)}_{i_1} u^{(2)}_{i_2} \cdots u^{(N)}_{i_N}$ for all values of the indices, which will be concisely written as $\mathcal{A} = U^{(1)} \circ U^{(2)} \circ \cdots \circ U^{(N)}$. The n-rank of a higher-order tensor is the obvious generalization of the column (row) rank of matrices: given an $(I_1 \times I_2 \times \cdots \times I_N)$-tensor $\mathcal{A}$, it equals the dimension of its n-mode vector space, i.e., the vector space spanned by the I_n-dimensional vectors obtained from $\mathcal{A}$ by varying the index i_n and keeping the other indices fixed. An important difference with the rank of matrices is that the different n-ranks of a higher-order tensor are not necessarily the same. The n-rank will be denoted as $\text{rank}_n(\mathcal{A})$. An Nth-order tensor of which $\text{rank}_1(\mathcal{A}) = R_1$, $\text{rank}_2(\mathcal{A}) = R_2$, etc., will briefly be called a rank-$(R_1, R_2, \ldots, R_N)$ tensor. This is not to be confused with a "rank-R tensor", by which one generally means a tensor that can be decomposed in a sum of R, but not less than R, rank-1 terms; see, e.g., [16].

This paper is a follow-up of [9], in which we discussed a multilinear generalization of the singular value decomposition (SVD). For convenience, we will refer to this decomposition as the higher-order SVD (HOSVD). The starting point of our dis-

*Received by the editors November 6, 1998; accepted for publication (in revised form) by A. Edelman April 4, 1999; published electronically May 4, 2000. This research was partially supported by the Flemish government through the Concerted Research Actions GOA-MIPS and GOA-MEFISTO-666, the Fund for Scientific Research–Flanders (FWO) projects G.0240.99, and the Belgian State, Prime Minister's Office–Federal Office for Scientific, Technical, and Cultural Affairs, through the Interuniversity Poles of Attraction Programmes IUAP P4-02 and IUAP P4-24. The scientific responsibility is assumed by the authors.

http://www.siam.org/journals/simax/21-4/34699.html

†ESAT—SISTA/COSIC, Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, B-3001 Leuven (Heverlee), Belgium (Lieven.DeLathauwer@esat.kuleuven.ac.be, Bart.DeMoor@esat.kuleuven.ac.be, Joos.Vandewalle@esat.kuleuven.ac.be, http://www.esat.kuleuven.ac.be/sista).

cussion is that, despite the many analogies between the SVD and the HOSVD, the tensor decomposition does not reflect a simple higher-order equivalent of the classical link between the best rank-R approximation of a given matrix and its truncated SVD. Although truncation of the HOSVD of a given tensor may lead to a good rank-$(R_1, R_2, \ldots, R_N)$ approximation ([9] contains an error bound), it turns out that this tensor is in general not the best possible (least-squares) approximation under the given n-mode rank constraints. This paper reports some research results on the estimation of best rank-$(R_1, R_2, \ldots, R_N)$ approximations.

Important research on this topic has already been carried out by Kroonenberg [13, 14], Kroonenberg and de Leeuw [15], and ten Berghe, de Leeuw, and Kroonenberg [19]. They devised an alternating least-squares (ALS) method to improve the fit, which is known as "three-mode factor analysis" in psychometrics. (The generalization to orders higher than three is briefly indicated in [13].) The basic idea is to optimize, mode per mode and in an iterative way, the components of a factorization of the given tensor; each optimization step essentially involves a best reduced-rank approximation of a positive (semi)definite symmetric matrix. We will discuss this work and present the following refinements and complementary results:

(1) In the fundamental best rank-1 approximation problem, an approximation of a given higher-order tensor can gradually be enhanced by means of a relatively simple higher-order generalization of the power method [12] (see sections 3.1 and 3.2).

(2) The efficiency of the higher-order power algorithm can further be increased by updating two modes at the same time (see section 3.3).

(3) Kroonenberg, de Leeuw, and ten Berghe initialized their algorithm with a truncated HOSVD model, but it was indicated that only local optimization was guaranteed. In section 3.4 we will explicitly show that initializing optimization routines with a truncated HOSVD indeed does not always lead to the global optimum; on the other hand, it is our experience that defective cases are rarely met.

(4) For the elementary case of supersymmetric $(2 \times 2 \times \cdots \times 2)$-tensors, the symmetric stationary points of the higher-order power algorithm can be fully characterized (see section 3.5). This case is important, e.g., with respect to applications in blind source separation [11].

(5) With respect to arbitrary values of $R_1, R_2, \ldots, R_N$, we will present a square-root version of the original algorithm, aiming at a higher accuracy, and interpret it as a multilinear generalization of the technique of orthogonal iterations [12] (see section 4).

For clarity, the concepts of this paper are formulated in terms of real-valued tensors. They can be generalized for complex tensors.

Before starting with the next section, we add a comment on the notation that is used. To facilitate the distinction between scalars, vectors, matrices, and higher-order tensors, the type of a given quantity will be reflected by its representation: scalars are denoted by lower-case letters (a, b, ...; α, β, ...), vectors are written as italic capitals (A, B, ...), matrices correspond to boldface capitals ($\mathbf{A}$, $\mathbf{B}$, ...), and tensors are written as calligraphic letters ($\mathcal{A}$, $\mathcal{B}$, ...). This notation is consistently used for lower-order parts of a given structure. For example, 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 $(A)_i = a_i$ and $(\mathcal{A})_{i_1 i_2 \cdots i_N} = a_{i_1 i_2 \cdots i_N}$); furthermore, the ith column vector of a matrix $\mathbf{A}$ is denoted as A_i, i.e., $\mathbf{A} = [A_1 A_2 \cdots]$. To enhance the overall readability, we have made one exception to this rule: as we frequently use the characters i, j, r, and n in the meaning of indices (counters), I, J, R, and N will be reserved to denote (unless stated

otherwise) the index upper bounds.

2. Basic definitions. In this section, we introduce some elementary notations and definitions needed in subsequent developments.

2.1. Multiplication of a higher-order tensor by a matrix. Higher-order power and orthogonal iterations involve a multilinear equivalent of matrix-vector and matrix-matrix multiplications.

Let us first have a look at the matrix product $\mathbf{G} = \mathbf{U} \cdot \mathbf{F} \cdot \mathbf{V}^T$, involving matrices $\mathbf{F} \in \mathbb{R}^{I_1 \times I_2}$, $\mathbf{U} \in \mathbb{R}^{J_1 \times I_1}$, $\mathbf{V} \in \mathbb{R}^{J_2 \times I_2}$, and $\mathbf{G} \in \mathbb{R}^{J_1 \times J_2}$. To avoid working with "generalized transposes" in the multilinear case (in which the fact that 1-mode vectors are transpose-free would not have an inherent meaning), we observe that the relationship between $\mathbf{U}$ and $\mathbf{F}$ and the relationship between $\mathbf{V}$ (not $\mathbf{V}^T$) and $\mathbf{F}$ are in fact completely similar: in the same way that $\mathbf{U}$ makes linear combinations of the rows of $\mathbf{F}$, $\mathbf{V}$ makes linear combinations of the columns of $\mathbf{F}$; in the same way that the columns of $\mathbf{F}$ are multiplied by $\mathbf{U}$, the rows of $\mathbf{F}$ are multiplied by $\mathbf{V}$; in the same way that the columns of $\mathbf{U}$ are associated with the column space of $\mathbf{G}$, the columns of $\mathbf{V}$ are associated with the row space of $\mathbf{G}$. This typical relationship will be denoted by means of the $\times_n$ symbol: $\mathbf{G} = \mathbf{F} \times_1 \mathbf{U} \times_2 \mathbf{V}$.

In general, we make the following definition.

DEFINITION 2.1. *The n-mode product of a tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \cdots \times I_N}$ by a matrix $\mathbf{U} \in \mathbb{C}^{J_n \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{U}$, is an $(I_1 \times I_2 \times \cdots \times I_{n-1} \times J_n \times I_{n+1} \cdots \times I_N)$-tensor of which the entries are given by*

$$(\mathcal{A} \times_n \mathbf{U})_{i_1 i_2 \cdots i_{n-1} j_n i_{n+1} \cdots i_N} \stackrel{\text{def}}{=} \sum_{i_n} a_{i_1 i_2 \cdots i_{n-1} i_n i_{n+1} \cdots i_N} u_{j_n i_n}.$$

The n-mode product satisfies the following properties.

PROPERTY 1. *Given the tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \cdots \times I_N}$ and the matrices $\mathbf{F} \in \mathbb{C}^{J_n \times I_n}$, $\mathbf{G} \in \mathbb{C}^{J_m \times I_m}$, one has*

$$(\mathcal{A} \times_n \mathbf{F}) \times_m \mathbf{G} = (\mathcal{A} \times_m \mathbf{G}) \times_n \mathbf{F} = \mathcal{A} \times_n \mathbf{F} \times_m \mathbf{G}.$$

PROPERTY 2. *Given the tensor $\mathcal{A} \in \mathbb{C}^{I_1 \times I_2 \times \cdots \times I_N}$ and the matrices $\mathbf{F} \in \mathbb{C}^{J_n \times I_n}$, $\mathbf{G} \in \mathbb{C}^{K_n \times J_n}$, one has*

$$(\mathcal{A} \times_n \mathbf{F}) \times_n \mathbf{G} = \mathcal{A} \times_n (\mathbf{G} \cdot \mathbf{F}).$$

2.2. Matrix representation of a higher-order tensor. To be able to express our results in a more common matrix language, we define "matrix unfoldings" of a given tensor, i.e., matrix representations of that tensor in which all the column vectors (row vectors, etc.) are ordered sequentially. To avoid confusion, we will retain one particular ordering of the column (row, etc.) vectors; for order three, these unfolding procedures are represented in Figure 2.1. Notice that the definitions of the matrix unfoldings involve the tensor dimensions I_1, I_2, I_3 in a cyclic way and that, when dealing with an unfolding of dimensionality $I_c \times I_a I_b$, we formally assume that the index i_a varies more slowly than i_b. In general, we make the following definition.

DEFINITION 2.2. *Assume an Nth-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. The matrix unfolding $\mathbf{A}_{(n)} \in \mathbb{R}^{I_n \times (I_{n+1} I_{n+2} \cdots I_N I_1 I_2 \cdots I_{n-1})}$ contains the element $a_{i_1 i_2 \cdots i_N}$ at the position with row number i_n and column number equal to $(i_{n+1} - 1) I_{n+2} I_{n+3} \cdots I_N I_1 I_2 \cdots I_{n-1} + (i_{n+2} - 1) I_{n+3} I_{n+4} \cdots I_N I_1 I_2 \cdots I_{n-1} + \cdots + (i_N - 1) I_1 I_2 \cdots I_{n-1} + (i_1 - 1) I_2 I_3 \cdots I_{n-1} + (i_2 - 1) I_3 I_4 \cdots I_{n-1} + \cdots + i_{n-1}$.*

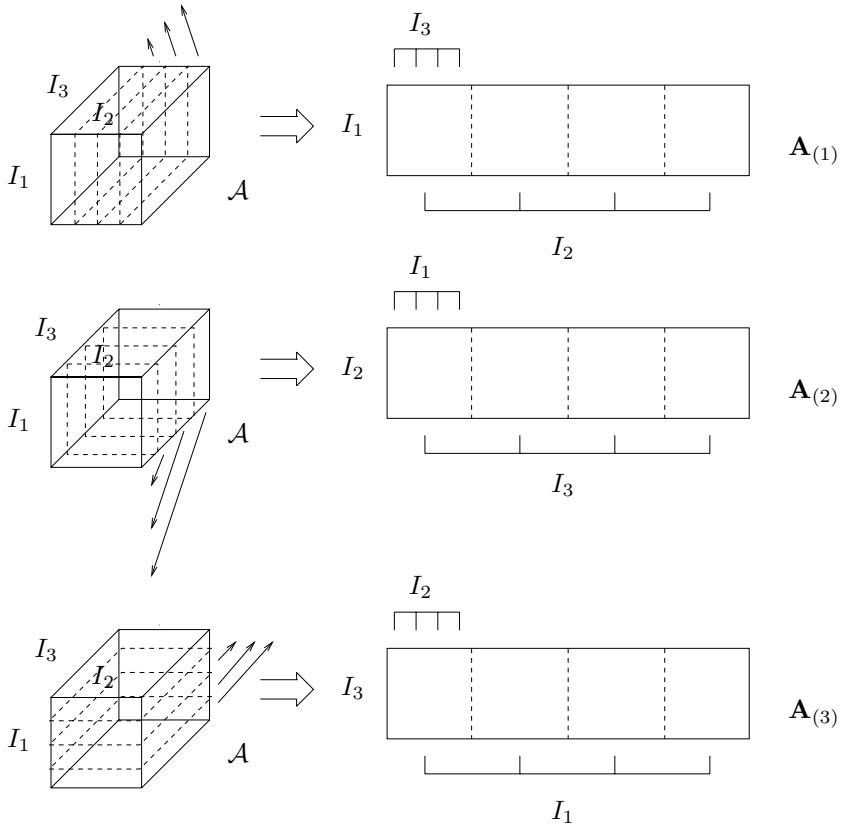

FIG. 2.1. *Unfolding of the* $(I_1 \times I_2 \times I_3)$*-tensor* $\mathcal{A}$ *to the* $(I_1 \times I_2 I_3)$*-matrix* $\mathbf{A}_{(1)}$*, the* $(I_2 \times I_3 I_1)$*-matrix* $\mathbf{A}_{(2)}$*, and the* $(I_3 \times I_1 I_2)$*-matrix* $\mathbf{A}_{(3)}$ $(I_1 = I_2 = I_3 = 4)$.

Example 1. Define a tensor $\mathcal{A} \in \mathbb{R}^{3\times 2\times 3}$ by $a_{111} = a_{112} = a_{211} = -a_{212} = 1$, $a_{213} = a_{311} = a_{313} = a_{121} = a_{122} = a_{221} = -a_{222} = 2$, $a_{223} = a_{321} = a_{323} = 4$, $a_{113} = a_{312} = a_{123} = a_{322} = 0$. The matrix unfolding $\mathbf{A}_{(1)}$ is given by

$$\mathbf{A}_{(1)} = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 2 & 2 & 0 \\ 1 & -1 & 2 & 2 & -2 & 4 \\ 2 & 0 & 2 & 4 & 0 & 4 \end{array}\right).$$

2.3. Scalar product and Frobenius norm. The scalar product $\langle \mathcal{A}, \mathcal{B} \rangle$ of two tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is defined in a straightforward way as $\langle \mathcal{A}, \mathcal{B} \rangle \stackrel{\text{def}}{=} \sum_{i_1} \sum_{i_2} \cdots \sum_{i_N} a_{i_1 i_2 \cdots i_N} b_{i_1 i_2 \cdots i_N}$. The Frobenius norm of a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is then defined as $\|\mathcal{A}\| \stackrel{\text{def}}{=} \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle}$.

3. Higher-order power iteration. In this section, we investigate how a given tensor can be approximated, in an optimal least-squares sense, by a tensor of rank 1. In section 3.1, the problem is formalized in two different ways and analyzed with the technique of Lagrange multipliers. In section 3.2, we show that an approximation can be improved by means of a higher-order generalization of the power method. A more efficient variant is presented in section 3.3. Section 3.4 deals with the choice of a good

initial value. For supersymmetric $(2\times 2\times\cdots\times 2)$-tensors, the determination of the best rank-1 approximation and the other symmetric stationary points of the higher-order power algorithm is reformulated as a polynomial rooting problem in section 3.5.

Section 3.2 can be considered as a special but important case of the ALS technique for the computation of the decomposition of a given tensor in a minimal sum of rank-1 terms [1, 3]. The idea is to update the components of the approximation mode per mode, which leads to a sequence of linear least-squares problems. However, in the rank-1 case these least-squares problems have a trivial solution, which allows for a faster approach, as explained in section 3.3. More background material on the decomposition of a tensor in a sum of rank-1 terms can be found in [5].

3.1. Best rank-1 approximation. The problem we want to solve can be described mathematically as follows.

Given a real Nth-order tensor $\mathcal{A}\in\mathbb{R}^{I_1\times I_2\times\cdots I_N}$, find a scalar λ and unit-norm vectors $U^{(1)}$, $U^{(2)}$, ..., $U^{(N)}$ such that the rank-1 tensor $\hat{\mathcal{A}}\stackrel{\text{def}}{=}\lambda U^{(1)}\circ U^{(2)}\circ\cdots\circ U^{(N)}$ minimizes the least-squares cost function

$$f(\hat{\mathcal{A}})=\|\mathcal{A}-\hat{\mathcal{A}}\|^2 \tag{3.1}$$

over the manifold of rank-1 tensors.

This constrained optimization problem can be analyzed using the technique of Lagrange multipliers. Therefore, we consider the following combination of f with the constraint terms:

$$\tilde{f}\stackrel{\text{def}}{=}\sum_{i_1 i_2\cdots i_N}(a_{i_1 i_2\cdots i_N}-\lambda u^{(1)}_{i_1}u^{(2)}_{i_2}\cdots u^{(N)}_{i_N})^2+\sum_n\lambda^{(n)}\left(\sum_{i_n}(u^{(n)}_{i_n})^2-1\right), \tag{3.2}$$

in which $\lambda^{(n)}$ $(1\leqslant n\leqslant N)$ are the Lagrange multipliers. Setting the derivative with respect to $u^{(n)}_{i_n}$ equal to zero yields

$$\lambda\sum_{i_1\cdots i_{n-1}i_{n+1}\cdots i_N}a_{i_1 i_2\cdots i_N}u^{(1)}_{i_1}\cdots u^{(n-1)}_{i_{n_1}}u^{(n+1)}_{i_{n+1}}\cdots u^{(N)}_{i_N}$$

$$=\lambda^{(n)}u^{(n)}_{i_n}+\lambda^2 u^{(n)}_{i_n}\sum_{i_1\cdots i_{n-1}i_{n+1}\cdots i_N}(u^{(1)}_{i_1})^2\cdots(u^{(n-1)}_{i_{n_1}})^2(u^{(n+1)}_{i_{n+1}})^2\cdots(u^{(N)}_{i_N})^2. \tag{3.3}$$

Derivation with respect to $\lambda^{(n)}$ and λ, respectively, yields

$$\sum_{i_n}(u^{(n)}_{i_n})^2=1, \tag{3.4}$$

$$\sum_{i_1 i_2\cdots i_N}a_{i_1 i_2\cdots i_N}u^{(1)}_{i_1}u^{(2)}_{i_2}\cdots u^{(N)}_{i_N}=\lambda\sum_{i_1 i_2\cdots i_N}(u^{(1)}_{i_1})^2(u^{(2)}_{i_2})^2\cdots(u^{(N)}_{i_N})^2. \tag{3.5}$$

Combining (3.4) with (3.5) and with the right-hand side of (3.3) yields

$$\sum_{i_1 i_2\cdots i_N}a_{i_1 i_2\cdots i_N}u^{(1)}_{i_1}u^{(2)}_{i_2}\cdots u^{(N)}_{i_N}=\lambda, \tag{3.6}$$

$$\lambda\sum_{i_1\cdots i_{n-1}i_{n+1}\cdots i_N}a_{i_1 i_2\cdots i_N}u^{(1)}_{i_1}\cdots u^{(n-1)}_{i_{n_1}}u^{(n+1)}_{i_{n+1}}\cdots u^{(N)}_{i_N}=(\lambda^2+\lambda^{(n)})u^{(n)}_{i_n}. \tag{3.7}$$

Combining (3.4) with the right-hand side of (3.7), and comparing this to (3.6), yields

$$\sum_{i_1 \cdots i_{n-1} i_{n+1} \cdots i_N} a_{i_1 i_2 \cdots i_N} u^{(1)}_{i_1} \cdots u^{(n-1)}_{i_{n_1}} u^{(n+1)}_{i_{n+1}} \cdots u^{(N)}_{i_N} = \lambda\, u^{(N)}_{i_N}. \tag{3.8}$$

Summarizing, the Lagrange equations correspond to $(1 \leqslant n \leqslant N)$:

$$\mathcal{A} \times_1 U^{(1)^T} \cdots \times_{n-1} U^{(n-1)^T} \times_{n+1} U^{(n+1)^T} \cdots \times_N U^{(N)^T} = \lambda\, U^{(n)}, \tag{3.9}$$

$$\mathcal{A} \times_1 U^{(1)^T} \times_2 U^{(2)^T} \cdots \times_N U^{(N)^T} = \lambda, \tag{3.10}$$

$$\|U^{(n)}\| = 1. \tag{3.11}$$

The best rank-1 approximation problem can also be formulated as follows.

THEOREM 3.1. *Assume a real Nth-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$, then the minimization of the cost function of* (3.1) *is equivalent to the maximization, over the unit-norm vectors $U^{(1)}, U^{(2)}, \ldots, U^{(N)}$, of the function*

$$g(U^{(1)}, U^{(2)}, \ldots, U^{(N)}) = \left| \mathcal{A} \times_1 U^{(1)^T} \times_2 U^{(2)^T} \cdots \times_N U^{(N)^T} \right|^2. \tag{3.12}$$

If the scalar λ, corresponding to the Frobenius norm of $\hat{\mathcal{A}}$ in (3.1)*, is chosen in accordance with* (3.10)*, then the functions of* (3.1) *and* (3.12) *are related by*

$$f = \|\mathcal{A}\|^2 - g. \tag{3.13}$$

Proof. We have the following:

$$f(\hat{\mathcal{A}}) = \|\mathcal{A} - \hat{\mathcal{A}}\|^2 = \|\mathcal{A}\|^2 - 2\langle \mathcal{A}, \hat{\mathcal{A}} \rangle + \|\hat{\mathcal{A}}\|^2.$$

According to the definition of λ, the value taken by $\langle \mathcal{A}, \hat{\mathcal{A}} \rangle$ equals λ^2. Since $U^{(1)}$, $U^{(2)}$, $\ldots$, $U^{(N)}$ have unit-norm, $\|\hat{\mathcal{A}}\|^2 = \lambda^2$ as well. Combination with the definition of g proves the theorem. ∎

Remark 1. We defined the best rank-1 approximation problem as the minimization of the distance between a given tensor and its approximation on the rank-1 manifold. Theorem 3.1 shows that this is equivalent to the maximization of the norm of the projection of the original tensor onto the rank-1 manifold.

Remark 2. Theorem 3.1 is the higher-order generalization of the fact that the computation of the best rank-1 approximation of a matrix $\mathbf{A}$ is equivalent to the determination of unit-vectors U and V such that $|U^T \mathbf{A} V|^2$ is maximal.

3.2. A power algorithm. For the actual computation of the best rank-1 approximation of $\mathcal{A}$, the Lagrange equations and their derivation can be interpreted in a constructive way. Imagine that the vectors $U^{(1)}, \ldots, U^{(n-1)}, U^{(n+1)}, \ldots, U^{(N)}$ are fixed and that f in (3.1) is merely a quadratic function in the unknown unconstrained vector $\lambda\, U^{(n)}$. Equation (3.9) then simply shows how the optimal $\lambda\, U$ can be computed for fixed $U^{(1)}, \ldots, U^{(n-1)}, U^{(n+1)}, \ldots, U^{(N)}$.

The full set of Lagrange equations $(1 \leqslant n \leqslant N)$ can be linked to an ALS algorithm for the (local) minimization of $f(\hat{\mathcal{A}})$: in each step the estimate of the scalar λ and the estimate of one of the vectors $U^{(1)}, U^{(2)}, \ldots, U^{(N)}$ are optimized, while the other vector estimates are kept constant. The result is shown in step 2 of Algorithm 3.2. We remark that the expression

$$\tilde{U}^{(n)}_{k+1} = \mathcal{A} \times_1 U^{(1)^T}_{k+1} \cdots \times_{n-1} U^{(n-1)^T}_{k+1} \times_{n+1} U^{(n+1)^T}_{k} \cdots \times_N U^{(N)^T}_{k}$$

ALGORITHM 3.2.
HIGHER-ORDER POWER METHOD.
In: $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$.
Out: $\hat{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$: estimator of best rank-1 approximation of $\mathcal{A}$.

1. Initial values: $U_0^{(n)}$ is the dominant left singular vector of $\mathbf{A}_{(n)}$ $(2 \leqslant n \leqslant N)$ and/or repeat the algorithm for several initial values.
2. Iterate until convergence:
   - $\tilde{U}_{k+1}^{(1)} = \mathcal{A} \times_2 {U_k^{(2)}}^T \times_3 {U_k^{(3)}}^T \cdots \times_N {U_k^{(N)}}^T$;
     $\lambda_{k+1}^{(1)} = \|\tilde{U}_{k+1}^{(1)}\|$;
     $U_{k+1}^{(1)} = \tilde{U}_{k+1}^{(1)} / \lambda_{k+1}^{(1)}$;
   - $\tilde{U}_{k+1}^{(2)} = \mathcal{A} \times_1 {U_{k+1}^{(1)}}^T \times_3 {U_k^{(3)}}^T \cdots \times_N {U_k^{(N)}}^T$;
     $\lambda_{k+1}^{(2)} = \|\tilde{U}_{k+1}^{(2)}\|$;
     $U_{k+1}^{(2)} = \tilde{U}_{k+1}^{(2)} / \lambda_{k+1}^{(2)}$;
   - $\ldots$
   - $\tilde{U}_{k+1}^{(N)} = \mathcal{A} \times_1 {U_{k+1}^{(1)}}^T \times_2 {U_{k+1}^{(2)}}^T \cdots \times_{N-1} {U_{k+1}^{(N-1)}}^T$;
     $\lambda_{k+1}^{(N)} = \|\tilde{U}_{k+1}^{(N)}\|$;
     $U_{k+1}^{(N)} = \tilde{U}_{k+1}^{(N)} / \lambda_{k+1}^{(N)}$.

   Converged values: $U^{(1)}, U^{(2)}, \ldots, U^{(N)}, \lambda$.
3. $\hat{\mathcal{A}} = \lambda\, U^{(1)} \circ U^{(2)} \circ \cdots \circ U^{(N)}$.

can be written in matrix format as follows:

$$\tilde{U}_{k+1}^{(n)} = \mathbf{A}_{(n)} \cdot (U_{k+1}^{(1)} \otimes \cdots \otimes U_{k+1}^{(n-1)} \otimes U_k^{(n+1)} \otimes \cdots \otimes U_k^{(N)}),$$

in which $\otimes$ represents the Kronecker product.

Clearly this technique is a higher-order extension of the power method for matrices [12]. The termination criterion could be formulated in terms of the accuracy of components (e.g., $|{U_{k+1}^{(n)}}^T U_k^{(n)}| < \epsilon$ for $1 \leqslant n \leqslant N$) or in terms of the quality of the fit (e.g., $\lambda_{k+1}^{(N)} - \lambda_k^{(N)} < \epsilon$). With respect to the initialization, we refer to section 3.4.

Remark 3. For the best approximation of a supersymmetric tensor $\mathcal{A}$ by a supersymmetric rank-1 tensor $\hat{\mathcal{A}} = \lambda\, U \circ U \circ \cdots \circ U$, the derivation is analogous. (Supersymmetric higher-order tensors are tensors that are invariant under arbitrary permutations of their indices; examples are the basic quantities of HOS.) The equivalent of (3.9) is

$$\mathcal{A} \times_1 U^T \cdots \times_{n-1} U^T \times_{n+1} U^T \cdots \times_N U^T = \lambda U. \tag{3.14}$$

This equation can as well be interpreted as an update of the unknown n-mode vector $\lambda U^{(n)}$, while the other vectors $U^{(1)}, \ldots, U^{(n-1)}, U^{(n+1)}, \ldots, U^{(N)}$ are fixed to U. However, such an update breaks the symmetry of the estimate unless λ and U already corresponded to a supersymmetric solution of the Lagrange equation. A symmetric version of the algorithm, based on the iteration

$$\tilde{U}_{k+1} = \mathcal{A} \times_1 U_k^T \times_2 U_k^T \cdots \times_{N-1} U_k^T, \tag{3.15}$$

for which $U_k^{(1)} = U_k^{(2)} = \cdots = U_k^{(N)} = U_k$, is unreliable since it does not necessarily decrease the cost function $f(\hat{\mathcal{A}})$ in a monotonous way, as can easily be verified from

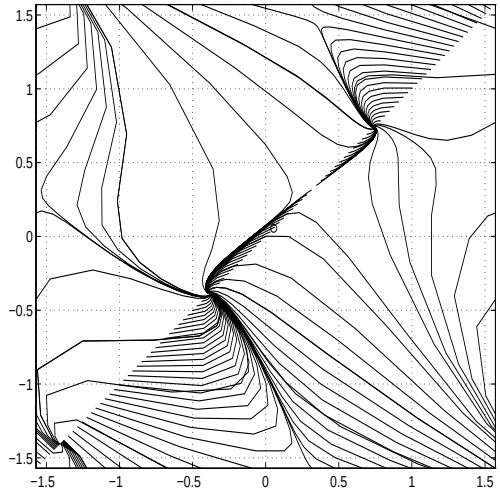

FIG. 3.1. *Depiction of Algorithm* 1 *for a typical example of a supersymmetric tensor in* $\mathbb{R}^{2\times 2\times 2}$. *Abscis*: *the angle* $\theta_k^{(2)}$ *in* $U_k^{(2)} = (\cos\theta_k^{(2)} \ \sin\theta_k^{(2)})^T$ (*in radians*). *Ordinate*: *the angle* $\theta_k^{(3)}$ *in* $U_k^{(3)} = (\cos\theta_k^{(3)} \ \sin\theta_k^{(3)})^T$ (*in radians*). *Both angles are normalized to the interval* $(-\pi/2, +\pi/2]$. *The small circle shows the initial guess obtained by HOSVD. The global optimum is* $(-0.3860, -0.3860)$.

examples. The fact that a power iteration produces unsymmetric intermediate results is not observed for symmetric matrices: at the second-order level, each iteration step involves only the knowledge of one intermediate vector, such that it is simply impossible to compare the different n-mode vector estimates at a given iteration step.

Example 2. Figure 3.1 depicts step 2 of Algorithm 3.2 for a $(2 \times 2 \times 2)$ example. For different choices of θ_0, determining initial vectors $U_0^{(2)} = U_0^{(3)} = (\cos\theta_0 \ \sin\theta_0)^T$, we plotted after each iteration the angle $\theta_k^{(3)}$ in $U_k^{(3)} = (\cos\theta_k^{(3)} \ \sin\theta_k^{(3)})^T$ versus the angle $\theta_k^{(2)}$ in $U_k^{(2)} = (\cos\theta_k^{(2)} \ \sin\theta_k^{(2)})^T$. The angles were normalized to the interval $(-\pi/2, +\pi/2]$. (The sign of the vectors does not matter, as can be seen from the definition of the function g; as far as f is concerned, the sign can be incorporated in the scalar λ.) The tensor $\mathcal{A}$ that we consider is supersymmetric and defined by

$$\begin{cases} a_{111} = 1.5578, & a_{222} = 1.1226, \\ a_{112} = -2.4443, & a_{221} = -1.0982. \end{cases}$$

We observe that Algorithm 3.2 leads to unsymmetric intermediate results not located on the main diagonal of the figure.

We also remark that there are two stable $((-0.3860, -0.3860), (0.7413, 0.7413))$ and two unstable $((-1.4052, -1.4052), (0.3347, 0.3347))$ symmetric stationary points of the higher-order power algorithm. The first three correspond to the three solutions of the Lagrange equation

$$\mathcal{A} \times_2 U^T \times_3 U^T = \lambda U \tag{3.16}$$

on the unit circle $\|U\| = 1$. The solution (0.3347, 0.3347) is special: even after convergence the three substeps of each iteration produce unsymmetric results, but these effects compensate such that the overall power iteration is symmetric. The global optimum is $(-0.3860, -0.3860)$.

3.3. Joint updates in the higher-order power algorithm. In the iteration scheme developed in section 3.2, the estimates of the vectors $U^{(1)}, U^{(2)}, \ldots, U^{(N)}$ are updated one at a time. However, it is possible to update the estimates of two of the vectors in a single iteration step.

Let us assume, for example, that the values of $U^{(1)}, \ldots, U^{(n-1)}, U_k^{(n+2)}, \ldots, U^{(N)}$ are fixed and that $U^{(n)}$ and $U^{(n+1)}$ have to be updated. Theorem 3.1 then implies that the optimal $U^{(n)}$ and $U^{(n+1)}$ can be found as the dominant left and right singular vectors of the matrix $\mathcal{A} \times_1 {U^{(1)}}^T \cdots \times_{n-1} {U^{(n-1)}}^T \times_{n+2} {U^{(n+2)}}^T \cdots \times_N {U^{(N)}}^T$, since $g = |{U^{(n)}}^T \cdot (\mathcal{A} \times_1 {U^{(1)}}^T \cdots \times_{n-1} {U^{(n-1)}}^T \times_{n+2} {U^{(n+2)}}^T \cdots \times_N {U^{(N)}}^T) \cdot U^{(n+1)}|^2$; the corresponding value of g is the squared dominant singular value.

The efficiency of Algorithm 3.2 can be increased by resorting to modern techniques for the estimation of the dominant singular triplet [12]. (Computation of the dominant singular triplet by a matrix power algorithm would correspond to an iteration between substeps n and $n+1$ of step 2 in Algorithm 3.2.) Note that, in the terminology of Algorithm 3.2, $U_k^{(n)}$ and $U_k^{(n+1)}$ can be used as first approximations of $U_{k+1}^{(n)}$ and $U_{k+1}^{(n+1)}$.

3.4. Starting value. Example 2 illustrates a major difference between the second-order and the higher-order problem of best rank-1 approximation: for tensors, the Lagrange equations can have several stable solutions. A descent algorithm, like the higher-order power method, will reveal only the global optimum if the starting point is in the attraction region of this global optimum.

Considering the fact that the best rank-1 approximation of a matrix is obtained by truncating its SVD after the first singular value, it is clear that estimation of $U^{(n)}$ as the dominant left singular vector of the matrix unfolding $\mathbf{A}_{(n)}$ $(1 \leqslant n \leqslant N)$ may yield a good rank-1 approximation. As a matter of fact, this corresponds to truncation of the HOSVD after the first term [9]. Another important difference between matrices and higher-order tensors is that this approximation is not necessarily the globally optimal one. (For example, in Figure 3.1 the estimate obtained by truncation of the HOSVD is indicated by means of a small circle.)

However, in our simulations we have observed that the HOSVD estimate usually belongs to the attraction region of the best rank-1 approximation. Hence we propose to use the HOSVD estimate as the starting value for the power iteration derived above; we assume that this iteration will not cause "jumps" to other attraction regions. However, there is no absolute guarantee: we have been able to generate special cases in which monotonous descent techniques eventually lead to a local optimum with a close-to-optimal fit, but nevertheless different from the global optimum. In this respect, repeating the optimization procedure for several initial values could be envisaged.

Example 3. Consider a tensor $\mathcal{B} \in \mathbb{R}^{2\times2\times2\times2}$, of which the nonzero entries are given by

$$\begin{cases} b_{1111} = 25.1, & b_{1212} = 25.6, \\ b_{2121} = 24.8, & b_{2222} = 23. \end{cases}$$

Due to the symmetries $b_{ijkl} = b_{kjil}$ and $b_{ijkl} = b_{ilkj}$, the best rank-1 approximation of $\mathcal{B}$ has the form $\lambda U^{(1)} \circ U^{(2)} \circ U^{(1)} \circ U^{(2)}$; we write $U^{(1)} = (\cos\alpha \ \ \sin\alpha)^T$ and $U^{(2)} = (\cos\beta \ \ \sin\beta)^T$. The function $\tilde{g}_1 \stackrel{\text{def}}{=} \mathcal{B} \times_1 {U^{(1)}}^T \times_2 {U^{(2)}}^T \times_3 {U^{(1)}}^T \times_4 {U^{(2)}}^T$ corresponds to a weighted arithmetic mean of the nonzero entries of $\mathcal{B}$:

$$\tilde{g}_1(\alpha, \beta) = 25.1 \cos^2 \alpha \cos^2 \beta + 25.6 \cos^2 \alpha \sin^2 \beta + 24.8 \sin^2 \alpha \cos^2 \beta + 23 \sin^2 \alpha \sin^2 \beta.$$

The global maximum of this function is 25.6, reached for $\alpha = 0$ and $\beta = \pi/2$; there are no essentially different local maxima. According to Theorem 3.1, setting $\lambda = 25.6$, $U^{(1)} = (1\ 0)^T$, and $U^{(2)} = (0\ 1)^T$ gives the best rank-1 approximation of $\mathcal{B}$. On the other hand, $\mathcal{B}$ is such that $\mathbf{B}_{(1)}$ and $\mathbf{B}_{(2)}$ each consist of mutually orthogonal rows, arranged in order of decreasing Frobenius norm; hence truncation of the HOSVD yields the vectors $U^{(1)} = (1\ 0)^T$ and $U^{(2)} = (1\ 0)^T$, which belong to the correct attraction region in a trivial way.

Now consider a tensor $\mathcal{A} \in \mathbb{R}^{2\times 2\times 2\times 2}$, equal to $\mathcal{B}$, except for the entries

$$a_{1121} = 0.3, \quad a_{2111} = 0.3.$$

$\mathcal{A}$ has similar symmetries as $\mathcal{B}$. The function $\tilde{g}_2 \stackrel{\text{def}}{=} \mathcal{A} \times_1 {U^{(1)}}^T \times_2 {U^{(2)}}^T \times_3 {U^{(1)}}^T \times_4 {U^{(2)}}^T$, given by

$$\tilde{g}_2(\alpha, \beta) = \tilde{g}_1(\alpha, \beta) + 0.6 \cos^2 \beta \cos \alpha \sin \alpha, \tag{3.17}$$

is shown in Figure 3.2 by means of a mesh and a contour plot. The extra term in (3.17) is small enough such that $\tilde{g}_1$ and $\tilde{g}_2$ are similar—in particular, the global maximum is the same—but it induces a local maximum (namely, $(0.5536, 0)$) on the axis $\beta = 0$. On the contour plot, the zero-gradient points of $\tilde{g}_2$ are indicated by means of "+" marks; the global maximum is indicated by means of a small circle. $\mathbf{A}_{(1)}$ and $\mathbf{A}_{(2)}$ still have nearly orthogonal rows, arranged in order of decreasing Frobenius norm. Truncation of the HOSVD leads to vectors that are still close to $(1\ 0)^T$; this corresponds to the "×" mark on the contour plot. Because of the symmetry of $\tilde{g}_2$ around the axis $\beta = 0$, a gradient ascent starting from the truncated HOSVD will converge not to the global maximum but to the local optimum $(0.5536, 0)$. The same holds for the higher-order power algorithm: after each iteration the intermediate result was plotted as a dot in the contour plot of Figure 3.2. Altering the ordering of the substeps in Algorithm 3.2 yields comparable results. (Starting from the HOSVD guess, the global maximum could be found, however, by alternating, as indicated in section 3.3, between simultaneous updates of the mode-1 and mode-3 vector on one hand, and on the other hand the mode-2 and mode-4 vector, beginning with the former.) It is clear that this example is not an isolated case: the conditions that ensure that the HOSVD truncate is in the wrong attraction region are still satisfied for sufficiently small perturbations of $\mathcal{A}$.

3.5. Best rank-1 approximation of supersymmetric binary tensors. The best rank-1 approximation of supersymmetric $(2 \times 2 \times \cdots \times 2)$-tensors deserves special attention, as it appears that the determination of the symmetric stationary points of the higher-order power algorithm can be reformulated as a polynomial rooting problem in this case. Despite its elementary character, the problem is very relevant: e.g., in [11] we proved that it is intimately connected with the blind separation of a linear mixture of two independent sources; separation of mixtures of a larger size can be achieved by a Jacobi iteration over such elementary cases [4].

We first prove that there can be at most three distinct symmetric solutions to the Lagrange equations for a supersymmetric $(2 \times 2 \times 2)$-tensor; the generalization to arbitrary tensor orders is straightforward. For convenience we use the notation

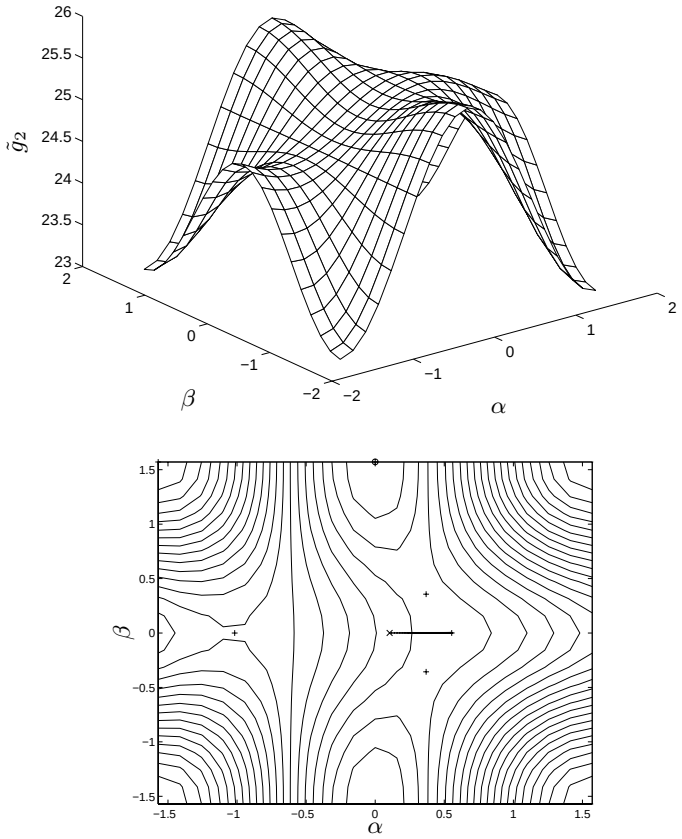

FIG. 3.2. *The quality of the best rank-*1 *approximation of the partially symmetric tensor* $\mathcal{A}$ *in Example* 3*, for different values of* α *and* β*. The "*$\times$*" mark shows the initial guess obtained by HOSVD. The global optimum is indicated by means of a small circle. The "+" marks correspond to the zero-gradient points of the function* $\tilde{g}_2$*, defined in the example. The intermediate results after each iteration cycle of Algorithm* 3.2*, initialized with the first HOSVD components, are plotted as a sequence of dots.*

$U = (c\ s)^T$. Equation (3.16) corresponds to the following set of equations:

$$a_{111}c^2 + 2a_{112}cs + a_{122}s^2 = \lambda c,$$
$$a_{211}c^2 + 2a_{212}cs + a_{222}s^2 = \lambda s.$$

Eliminating λ and claiming that $\|U\|^2 = 1$ yields

$$a_{211}c^3 + (2a_{212} - a_{111})c^2 s + (a_{222} - 2a_{112})cs^2 - a_{122}s^3 = 0,$$
$$c^2 + s^2 = 1.$$

On the unit circle a solution is entirely defined by $t = s/c$, each time corresponding to the vectors $(c\ s)^T$ and $(-c\ -s)^T$, which are not essentially different. The solutions can be found by rooting the polynomial

$$a_{211} + (2a_{212} - a_{111})t + (a_{222} - 2a_{112})t^2 - a_{122}t^3 = 0,$$

which has at most three distinct real roots. The same expression shows where the derivative of $\tilde{g} = \mathcal{A} \times_1 U^T \times_2 U^T \times_3 U^T$ is equal to zero. Similarly, for an Nth-order supersymmetric $(2 \times 2 \times \cdots \times 2)$-tensor, there can be at most N distinct symmetric solutions to the Lagrange equations; these solutions can be found as the roots of an Nth-order polynomial.

On the other hand, it is also possible that a higher-order power iteration yields a stationary solution that is characterized by a symmetric state $U^{(2)} = U^{(3)} = \cdots = U$, but nevertheless produces an intermediate vector $U^{(1)}$ that is different from U; e.g., see the solution (0.3347, 0.3347) in Example 2. Therefore, let us consider solutions to the Lagrange equations of the type

$$\mathcal{A} \times_2 U^T \times_3 U^T = \lambda U^{(1)}, \tag{3.18}$$

$$\mathcal{A} \times_2 {U^{(1)}}^T \times_3 U^T = \lambda U \tag{3.19}$$

for a supersymmetric $(2 \times 2 \times 2)$-tensor $\mathcal{A}$, where $U^{(1)}$ is not necessarily equal to U. Equations (3.18) and (3.19) can be combined as

$$\mathcal{B} \times_2 U^T \times_3 U^T \times_4 U^T = \lambda U, \tag{3.20}$$

in which the tensor $\mathcal{B}$ is defined by $b_{ijkl} = \sum_p a_{ijp} a_{klp}$ for all entries. Although $\mathcal{B}$ itself is not supersymmetric, the solutions to (3.20) can be determined as above; at most four distinct real solutions are possible. Similarly, for an Nth order supersymmetric $(2 \times 2 \times \cdots \times 2)$-tensor, the solutions to the Lagrange equations

$$\mathcal{A} \times_2 U^T \times_3 U^T \cdots \times_N U^T = \lambda U^{(1)}, \tag{3.21}$$

$$\mathcal{A} \times_2 {U^{(1)}}^T \times_3 U^T \cdots \times_N U^T = \lambda U \tag{3.22}$$

can be found by rooting a polynomial of degree $2N - 2$.

The determination of arbitrary unsymmetric stationary points of a higher-order power iteration is much harder than the analysis of the two specific cases above; it generally leads to sets of polynomial equations and will therefore not be considered.

4. Higher-order orthogonal iteration. In this section we generalize the best rank-1 approximation problem of the previous section in the sense that the approximation should now have prespecified mode-1 rank, mode-2 rank, etc. The derivation of the computational procedure follows a similar scheme. In section 4.1, we give two related formal definitions of the approximation problem. Section 4.2 is devoted to the actual computation of the solution.

The derivation follows roughly the same lines as in the work by Kroonenberg [13, 14], Kroonenberg and de Leeuw [15], and ten Berghe, de Leeuw, and Kroonenberg [19], but with some modifications in the practical implementation. On the other hand, the overall presentation is intended to clarify the general linear/multilinear framework underlying the best rank-R/rank-$(R_1, R_2, \ldots, R_N)$ approximation problem. The algorithm that we present is a square-root version of the Kroonenberg algorithm, which increases the accuracy, as is well known [12]. Next, the computation scheme can be based on the calculation of dominant subspaces, rather than individual singular vectors, resulting in a higher efficiency. The algorithm will be interpreted as a multilinear generalization of the technique of orthogonal iterations [12]. An important remark concerns the initialization of the algorithm. Kroonenberg, de Leeuw, and ten Berghe initialized their algorithm with a truncated HOSVD, but it was indicated that only

local optimization was guaranteed. In section 3.4 we have demonstrated that initializing optimization routines with a truncated HOSVD does not always lead to the global optimum, though it is our experience that defective cases are rarely met.

4.1. Best rank-$(R_1, R_2, \ldots, R_N)$ approximation. In this section we generalize the best rank-1 approximation problem of the previous section to the best approximation by a tensor with prespecified mode-1 rank, mode-2 rank, etc. Formally, the problem we want to solve can be formulated as follows.

Given a real Nth-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$, find a tensor $\hat{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$, having $\text{rank}_1(\hat{\mathcal{A}}) = R_1$, $\text{rank}_2(\hat{\mathcal{A}}) = R_2, \ldots, \text{rank}_N(\hat{\mathcal{A}}) = R_N$, that minimizes the least-squares cost function

$$f(\hat{\mathcal{A}}) = \|\mathcal{A} - \hat{\mathcal{A}}\|^2. \tag{4.1}$$

The n-rank conditions imply that $\hat{\mathcal{A}}$ can be decomposed as

$$\hat{\mathcal{A}} = \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}, \tag{4.2}$$

in which $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}$, $\mathbf{U}^{(2)} \in \mathbb{R}^{I_2 \times R_2}$, $\ldots$, $\mathbf{U}^{(N)} \in \mathbb{R}^{I_N \times R_N}$ each have orthonormal columns and $\mathcal{B} \in \mathbb{R}^{R_1 \times R_2 \times \cdots \times R_N}$.

Actually it is sufficient to determine the matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$ for the optimization of f. For any estimate of these matrices, the optimal tensor $\mathcal{B}$ is given by the following theorem.

THEOREM 4.1. *Assume a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$ and consider a tensor $\hat{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$ as in* (4.2)*. For given matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$, the tensor $\mathcal{B}$ that optimizes $f(\hat{\mathcal{A}}) = \|\mathcal{A} - \hat{\mathcal{A}}\|^2$ is given by*

$$\mathcal{B} = \mathcal{A} \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \ldots \times_N \mathbf{U}^{(N)^T}. \tag{4.3}$$

Proof. The optimization of

$$f = \|\mathcal{A} - \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}\|^2$$

for the unknown elements of $\mathcal{B}$ is merely a classical linear least-squares problem. It corresponds to the least-squares estimation of the solution of the following set of linear equations, which is possibly overdetermined:

$$\mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)} = \mathcal{A}.$$

Taking into account that $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$ have orthonormal columns, these matrices can be brought to the other side of the equation by (n-mode) multiplication with the transposed matrices. ∎

Remark 4. Theorem 4.1 is the multidimensional equivalent of the optimal choice of λ in (3.10).

Analogous with Theorem 3.1 we state that the best rank-$(R_1, R_2, \ldots, R_N)$ approximation problem can also be formulated as follows.

THEOREM 4.2. *Assume a real Nth-order tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \ldots I_N}$; then the minimization of the cost function of* (4.1) *is equivalent to the maximization, over the matrices $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$ having orthonormal columns, of the function*

$$g(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}) = \left\| \mathcal{A} \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \cdots \times_N \mathbf{U}^{(N)^T} \right\|^2. \tag{4.4}$$

If the tensor $\mathcal{B}$, the Frobenius norm of which equals the Frobenius norm of $\hat{\mathcal{A}}$ in (4.2)*, is chosen in accordance with* (4.3)*, then the functions of* (4.1) *and* (4.4) *are related by*

$$f = \|\mathcal{A}\|^2 - g. \tag{4.5}$$

Proof. We look for the relations between the definition of g and the terms in the following expression for f:

$$f(\hat{\mathcal{A}}) = \|\mathcal{A} - \hat{\mathcal{A}}\|^2 = \|\mathcal{A}\|^2 - 2\langle \mathcal{A}, \hat{\mathcal{A}}\rangle + \|\hat{\mathcal{A}}\|^2.$$

First, the definition of the inner product allows us to write

$$\begin{aligned}\langle \mathcal{A}, \hat{\mathcal{A}}\rangle &= \langle \mathcal{A}, \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}\rangle \\ &= \langle \mathcal{A} \times_1 \mathbf{U}^{(1)^T} \times_2 \mathbf{U}^{(2)^T} \cdots \times_N \mathbf{U}^{(N)^T}, \mathcal{B}\rangle \\ &= \|\mathcal{B}\|^2.\end{aligned}$$

Next, since $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$ have orthonormal columns, they do not affect the Frobenius norm

$$\|\hat{\mathcal{A}}\|^2 = \|\mathcal{B}\|^2.$$

Substitution of the preceding expressions in the equation for f yields

$$f(\hat{\mathcal{A}}) = \|\mathcal{A}\|^2 - \|\mathcal{B}\|^2.$$

Combination with the definition of g proves the theorem. ∎

Remark 5. Basically, the best rank-$(R_1, R_2, \ldots, R_N)$ approximation problem consists of the determination of a reduced n-rank tensor $\hat{\mathcal{A}}$ that explains as much of the "energy" (sum of the squared entries) of a given tensor $\mathcal{A}$ as possible under the given n-rank constraints. Theorem 4.2 implies that this problem is equivalent to the explicit maximization of the energy of the approximation.

In other words, we are looking for a basis of rank-1 tensors $\{U_{r_1}^{(1)} \circ U_{r_2}^{(2)} \circ \cdots \circ U_{r_N}^{(N)}\}$, in which $U_1^{(n)}, U_2^{(n)}, \ldots, U_{R_n}^{(n)}$ are mutually orthonormal $(1 \leqslant n \leqslant N)$, such that the norm of the projection of $\mathcal{A}$ onto this basis is maximal.

4.2. An orthogonal iteration. For the actual computation of the best rank-$(R_1, R_2, \ldots, R_N)$ approximation of $\mathcal{A}$, let us interpret the function g in the following way. Imagine that the matrices $\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(n-1)}, \mathbf{U}^{(n+1)}, \ldots, \mathbf{U}^{(N)}$ are fixed and that g in (4.4) is merely a quadratic expression in the components of the unknown matrix $\mathbf{U}^{(n)}$, consisting of orthonormal columns. We have

$$g = \|\tilde{\mathcal{U}}^{(n)} \times_n \mathbf{U}^{(n)^T}\|^2, \tag{4.6}$$

in which

$$\tilde{\mathcal{U}}^{(n)} \stackrel{\text{def}}{=} \mathcal{A} \times_1 \mathbf{U}^{(1)^T} \cdots \times_{n-1} \mathbf{U}^{(n-1)^T} \times_{n+1} \mathbf{U}^{(n+1)^T} \cdots \times_N \mathbf{U}^{(N)^T}. \tag{4.7}$$

Hence the columns of $\mathbf{U}^{(n)}$ can be found as an orthonormal basis for the dominant subspace of the n-mode space of $\tilde{\mathcal{U}}^{(n)}$.

Repeating this procedure for different mode numbers leads to an ALS algorithm for the (local) minimization of $f(\hat{\mathcal{A}})$: in each step the estimate of one of the matrices $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$ is optimized, while the other matrix estimates are kept constant.

ALGORITHM 4.2.
HIGHER-ORDER ORTHOGONAL ITERATION.
In: $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$.
Out: $\hat{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$: estimator of best rank-$(R_1, R_2, \ldots, R_N)$ approximation of $\mathcal{A}$.

1. Initial values: $\mathbf{U}_0^{(n)} \in \mathbb{R}^{I_n \times R_n}$, of which the columns form an orthonormal basis for the dominant R_n-dimensional left singular subspace of $\mathbf{A}_{(n)}$ $(2 \leqslant n \leqslant N)$ and/or repeat the algorithm for several initial values.
2. Iterate until convergence:
   - $\tilde{\mathcal{U}}_{k+1}^{(1)} = \mathcal{A} \times_2 \mathbf{U}_k^{(2)^T} \times_3 \mathbf{U}_k^{(3)^T} \cdots \times_N \mathbf{U}_k^{(N)^T}$;
     Maximize over $\mathbf{U}^{(1)} \in \mathbb{R}^{I_1 \times R_1}$ with $\mathbf{U}^{(1)^T}\mathbf{U}^{(1)} = \mathbf{I}$:
     $h(\mathbf{U}^{(1)}) = \|\tilde{\mathcal{U}}_{k+1}^{(1)} \times_1 \mathbf{U}^{(1)^T}\|$;
     $\max(h(\mathbf{U}^{(1)})) = h(\mathbf{U}_{\max}^{(1)}) = \lambda_{k+1}^{(1)}$;
     $\mathbf{U}_{k+1}^{(1)} = \mathbf{U}_{\max}^{(1)}$;
   - $\tilde{\mathcal{U}}_{k+1}^{(2)} = \mathcal{A} \times_1 \mathbf{U}_{k+1}^{(1)^T} \times_3 \mathbf{U}_k^{(3)^T} \cdots \times_N \mathbf{U}_k^{(N)^T}$;
     Maximize over $\mathbf{U}^{(2)} \in \mathbb{R}^{I_2 \times R_2}$ with $\mathbf{U}^{(2)^T}\mathbf{U}^{(2)} = \mathbf{I}$:
     $h(\mathbf{U}^{(2)}) = \|\tilde{\mathcal{U}}_{k+1}^{(2)} \times_2 \mathbf{U}^{(2)^T}\|$;
     $\max(h(\mathbf{U}^{(2)})) = h(\mathbf{U}_{\max}^{(2)}) = \lambda_{k+1}^{(2)}$;
     $\mathbf{U}_{k+1}^{(2)} = \mathbf{U}_{\max}^{(2)}$;
   - $\ldots$
   - $\tilde{\mathcal{U}}_{k+1}^{(N)} = \mathcal{A} \times_1 \mathbf{U}_{k+1}^{(1)^T} \times_2 \mathbf{U}_{k+1}^{(2)^T} \cdots \times_{N-1} \mathbf{U}_{k+1}^{(N-1)^T}$;
     Maximize over $\mathbf{U}^{(N)} \in \mathbb{R}^{I_N \times R_N}$ with $\mathbf{U}^{(N)^T}\mathbf{U}^{(N)} = \mathbf{I}$:
     $h(\mathbf{U}^{(N)}) = \|\tilde{\mathcal{U}}_{k+1}^{(N)} \times_N \mathbf{U}^{(N)^T}\|$;
     $\max(h(\mathbf{U}^{(N)})) = h(\mathbf{U}_{\max}^{(N)}) = \lambda_{k+1}^{(N)}$;
     $\mathbf{U}_{k+1}^{(N)} = \mathbf{U}_{\max}^{(N)}$.

   Converged values: $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}, \mathcal{B} = \tilde{\mathcal{U}}^{(N)} \times_N \mathbf{U}^{(N)^T}$.
3. $\hat{\mathcal{A}} = \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \cdots \times_N \mathbf{U}^{(N)}$.

The result is shown in step 2 of Algorithm 4.2. Note that (4.7) can be written in a matrix format as follows:

$$\tilde{\mathbf{U}}_{(n)}^{(n)} = \mathbf{A}_{(n)} \cdot (\mathbf{U}^{(n+1)} \otimes \cdots \otimes \mathbf{U}^{(N)} \otimes \mathbf{U}^{(1)} \otimes \cdots \otimes \mathbf{U}^{(n-1)}).$$

Clearly this technique is a higher-order extension of the orthogonal iteration for matrices [12]. A major difference, in some sense also with the higher-order power method, is that each iteration step involves not only the computation of a multilinear transformation but also the estimation of a dominant subspace.

As far as supersymmetric higher-order tensors are concerned, higher-order orthogonal iterations—like higher-order power iterations—yield unsymmetric intermediate results.

As with higher-order power iterations, it makes sense to initialize the higher-order orthogonal iteration with column-wise orthogonal matrices of which the columns span the space of the dominant left singular vectors of the matrix unfoldings $\mathbf{A}_{(n)}$ $(1 \leqslant n \leqslant N)$; this corresponds to truncation of the HOSVD [9]. We refer to section 3.4.

The complete higher-order orthogonal iteration algorithm, for a tensor $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, is presented in Algorithm 4.2.

With respect to Algorithm 4.2, we remark as follows:

(1) It is not required that the columns of $\mathbf{U}_k^{(n)}$ ($k \geqslant 0$ and $1 \leqslant n \leqslant N$) coincide with the left singular vectors of the matrix from which they are derived. It is sufficient to compute an arbitrary orthonormal basis for the R_n-dimensional dominant subspace.

(2) The projection of $\mathbf{U}_k^{(n)}$ on the n-mode space of $\tilde{\mathcal{U}}_{k+1}^{(n)}$ gives a first estimate of the dominant R_n-dimensional subspace in the calculation of $\mathbf{U}_{k+1}^{(n)}$. For a discussion of fast subspace computation methods, we refer to [12, 6].

(3) The stop criterion used in the algorithm for the computation of a dominant subspace, can be based on the value of $h(\mathbf{U}^{(n)})$, which corresponds to the square root of the function g that has to be optimized (see (4.6)). In terms of the accuracy of the components, the termination criterion can take the form of, e.g., $\|{\mathbf{U}_{k+1}^{(n)}}^T \cdot \mathbf{U}_k^{(n)}\|^2 > (1-\epsilon)R_n$ $(1 \leqslant n \leqslant N)$.

(4) In comparison with the situation in section 3.3, jointly updating the estimates of two of the matrices $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \ldots, \mathbf{U}^{(N)}$ can be quite involved. Let us assume, for example, that the values of $\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(n-1)}, \mathbf{U}_k^{(n+2)}, \ldots, \mathbf{U}^{(N)}$ are fixed and that $\mathbf{U}^{(n)}$ and $\mathbf{U}^{(n+1)}$ have to be updated. Theorem 4.2 then implies that the optimal matrices $\mathbf{U}^{(n)}$ and $\mathbf{U}^{(n+1)}$ can be found as the column-wise orthonormal maximizers of $\|(\mathcal{A} \times_1 {\mathbf{U}^{(1)}}^T \cdots \times_{n-1} {\mathbf{U}^{(n-1)}}^T \times_{n+2} {\mathbf{U}^{(n+2)}}^T \cdots \times_N {\mathbf{U}^{(N)}}^T) \times_n {\mathbf{U}^{(n)}}^T \times_{n+1} {\mathbf{U}^{(n+1)}}^T\|^2$. This involves the estimation of some kind of "simultaneous" dominant subspaces of $R_1 R_2 \cdots R_{n-1} R_{n+2} \cdots R_N$ matrix slices of $\mathcal{A} \times_1 {\mathbf{U}^{(1)}}^T \cdots \times_{n-1} {\mathbf{U}^{(n-1)}}^T \times_{n+2} {\mathbf{U}^{(n+2)}}^T \cdots \times_N {\mathbf{U}^{(N)}}^T$, which is a nontrivial problem if $R_1 R_2 \cdots R_{n-1} R_{n+2} \cdots R_N > 1$.

Example 4. Consider a tensor $\mathcal{A} \in \mathbb{R}^{3 \times 2 \times 2}$, defined by the following matrix unfolding:

$$\mathbf{A}_{(1)} = \left(\begin{array}{cc|cc} 0 & -1 & 1 & 4 \\ 2 & -2 & 3 & -5 \\ 4 & 3 & 5 & -6 \end{array}\right).$$

The left singular matrices of $\mathbf{A}_{(1)}$, $\mathbf{A}_{(2)}$, and $\mathbf{A}_{(3)}$ are given by

$$\mathbf{U}^{(1)} = \begin{pmatrix} -0.2465 & -0.4993 & 0.8306 \\ 0.5217 & 0.6539 & 0.5479 \\ 0.8167 & -0.5684 & -0.0994 \end{pmatrix}, \quad \mathbf{U}^{(2)} = \begin{pmatrix} 0.1715 & 0.9852 \\ 0.9852 & -0.1715 \end{pmatrix},$$

$$\mathbf{U}^{(3)} = \begin{pmatrix} 0.5105 & 0.8599 \\ -0.8599 & 0.5105 \end{pmatrix}.$$

The singular values equal (a) 11.0753, 3.5498, and 3.2768, (b) 10.6975 and 5.6181, and (c) 10.5162, 5.9506, and 0, respectively.

Truncation of the HOSVD after the first term in each mode gives a rank-1 approximation of $\mathcal{A}$ of which the Frobenius norm equals 10.0470. This value is increased to $\lambda = 10.1693$ by a higher-order power iteration (Algorithm 3.2). The evolution of the norm during the iteration is given by the solid line in Figure 4.1. The best rank-1 approximation $\lambda V^{(1)} \circ V^{(2)} \circ V^{(3)}$ is given by

$$V^{(1)} = \begin{pmatrix} -0.2515 \\ 0.6035 \\ 0.7567 \end{pmatrix}, \qquad V^{(2)} = \begin{pmatrix} 0.1344 \\ 0.9909 \end{pmatrix}, \qquad V^{(3)} = \begin{pmatrix} 0.5765 \\ -0.8171 \end{pmatrix}.$$

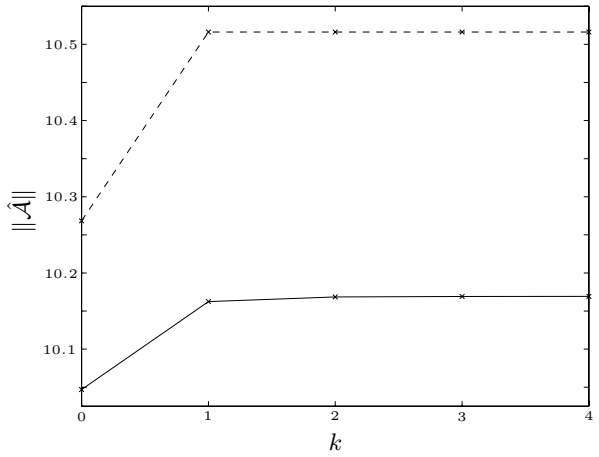

FIG. 4.1. *Evolution of the norm of the approximation $\hat{\mathcal{A}}$ of the tensor $\mathcal{A}$ in Example* 4*, during the higher-order power iteration or higher-order orthogonal iteration, as a function of the iteration step k. Solid line*: *computation of the best rank-*1 *approximation. Dashed line*: *computation of the best rank-*$(2, 2, 1)$ *approximation.*

The best rank-$(2, 2, 1)$ approximation of $\mathcal{A}$, decomposed as $\mathcal{B} \times_1 \mathbf{W}^{(1)} \times_2 \mathbf{W}^{(2)} \times_3 W^{(3)}$ as in (4.2), is given by

$$\mathbf{W}^{(1)} = \begin{pmatrix} -0.2789 & -0.4141 \\ 0.5984 & -0.7806 \\ 0.7511 & 0.4681 \end{pmatrix}, \qquad \mathbf{B}_{(1)} = \left(\begin{array}{cc|c} 10.1473 & 0.0000 \\ 0.0000 & 2.7607 \end{array}\right),$$

$$\mathbf{W}^{(2)} = \begin{pmatrix} 0.0982 & -0.9952 \\ 0.9952 & 0.0982 \end{pmatrix}, \qquad W^{(3)} = \begin{pmatrix} 0.5105 \\ -0.8599 \end{pmatrix}.$$

The Frobenius norm of this approximation equals 10.5162. The evolution of the norm during the higher-order orthogonal iteration (Algorithm 4.2) is given by the dashed line in Figure 4.1.

Example 5. As an alternative for section 4.2, one could think of a generalized deflation approach for the computation of the best rank-$(R_1, R_2, \ldots, R_N)$ approximation, i.e., one could wonder whether the result could not easily be obtained by means of successive rank-1 approximations. In this example we will show that such a procedure is not straightforward.

Consider the supersymmetric $(2 \times 2 \times 2)$-tensor $\mathcal{A}$ of which all the entries are equal to 1 except for $a_{111} = 2$. Obviously, the best rank-$(2, 2, 2)$ approximation is $\mathcal{A}$ itself. The best rank-1 approximation $\lambda_1 U_1 \circ U_1 \circ U_1$ can be computed as explained in section 3.5. Next, we consider the best rank-1 approximation $\lambda_2 U_2 \circ U_2 \circ U_2$ of the residue, and so on. We obtain

$$\begin{array}{lllll} \lambda_1 = 3.2560, & \lambda_2 = 0.5235, & \lambda_3 = -0.3213, & \lambda_4 = -0.1287, & \lambda_5 = 0.0597, \\ \lambda_6 = 0.0264, & \lambda_7 = -0.0118, & \lambda_8 = 0.0053, & \lambda_9 = 0.0024, & \end{array}$$

and so on, and

$$U_1 = \begin{pmatrix} 0.7981 \\ 0.6025 \end{pmatrix}, \quad U_2 = \begin{pmatrix} 0.9186 \\ -0.3952 \end{pmatrix}, \quad U_3 = \begin{pmatrix} 0.0392 \\ -0.9992 \end{pmatrix},$$
$$U_4 = \begin{pmatrix} 0.8733 \\ 0.4872 \end{pmatrix}, \quad U_5 = \begin{pmatrix} 0.8252 \\ -0.5649 \end{pmatrix}, \quad U_6 = \begin{pmatrix} 0.1355 \\ 0.9908 \end{pmatrix},$$
$$U_7 = \begin{pmatrix} 0.9469 \\ 0.3217 \end{pmatrix}, \quad U_8 = \begin{pmatrix} 0.7112 \\ -0.7030 \end{pmatrix}, \quad U_9 = \begin{pmatrix} 0.3107 \\ 0.9505 \end{pmatrix}, \quad \text{etc.}$$

We observe that, in contrast to the matrix case, $\mathcal{A}$ cannot immediately be derived from these rank-1 approximations. Neither is there a straightforward link between the series of rank-1 approximations and the decomposition of $\mathcal{A}$ in a minimal number of rank-1 terms; the latter decomposition is given by $\mathcal{A} = X_1 \circ X_1 \circ X_1 + X_2 \circ X_2 \circ X_2$, in which

$$X_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad X_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

5. Conclusion. There are some remarkable relations, but there are also some striking differences, between the best rank-R approximation of a matrix and the best rank-$(R_1, R_2, \ldots, R_N)$ approximation of an Nth-order tensor:

(1) Unlike the matrix case, the least-squares cost function can show several local optima for higher-order tensors.

(2) For matrices, the best rank-R approximation is obtained by setting the smallest singular values equal to zero while keeping the R largest ones. Truncation of the HOSVD may yield a good tensor approximation, but this approximation is generically suboptimal. However, our simulations suggest that there is still a weak link: it is observed that for well-conditioned problems the HOSVD estimate usually belongs to the valley of the least-squares cost function, corresponding to the global optimum.

(3) One way to obtain the dominant R-dimensional subspaces of the row and column space of a matrix is the power iteration ($R = 1$) or orthogonal iteration ($R > 1$). Higher-order generalizations of these techniques take the form of ALS algorithms, which can be used to enhance the fit between a tensor and a rank-1 or rank-$(R_1, R_2, \ldots, R_N)$ approximation of it.

(4) A higher-order power iteration step basically consists of a multilinear mapping instead of merely a linear transformation. In addition, higher-order orthogonal iterations can involve the estimation of a dominant subspace.

In the literature of psychometrics, the least-squares approximation of a multiway dataset by a dataset with reduced n-mode rank values is known as multimode factor analysis. In this paper, we have further contributed to a linear/multilinear algebraic framework for the best rank-R/rank-$(R_1, R_2, \ldots, R_N)$ approximation problem. We have presented a square-root algorithm, in which an iteration step is based on a multilinear mapping, followed by the estimation of a dominant subspace; this technique was interpreted as a higher-order generalization of the method of orthogonal iterations. We have demonstrated that starting the iteration from a truncated HOSVD does not lead to the global optimum in all possible cases. We have paid special attention to the case of supersymmetric tensors, in view of applications in, for example, HOS. We have extensively discussed the fundamental best rank-1 approximation problem. For supersymmetric $(2 \times 2 \times \cdots \times 2)$-tensors, the determination of the best rank-1

approximation and the other supersymmetric stationary points of the higher-order power algorithm has been reformulated as polynomial rooting problem.

Finally we remark that, as with matrices, the efficiency of higher-order power and orthogonal iterations could further be improved by means of a preprocessing consisting of a finite number of steps in which some tensor entries are set equal to zero. To this end, a higher-order equivalent of the Hessenberg decomposition is proposed in [10]. However this technique is less interesting for tensors than it is for matrices: the relative speed-up becomes smaller for larger tensor order N, as the matrices in which zeros are obtained (e.g., the matrices for which at least one index is equal to one, in the case of square tensors) form a relatively decreasing part of the tensor as N is increased.

REFERENCES

[1] R. Bro, *PARAFAC. Tutorial and applications*, Chemom. Intell. Lab. Systems, 38 (1997), pp. 149–171.

[2] J.-F. Cardoso and P. Comon, *Independent component analysis, a survey of some algebraic methods*, in Proceedings of ISCAS-96, Atlanta, GA, 1996, pp. 93–96.

[3] J. D. Carroll and S. Pruzansky, *The CANDECOMP-CANDELINC family of models and methods for multidimensional data analysis*, 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. 372–402.

[4] P. Comon, *Independent component analysis, a new concept?*, Signal Process., 36 (1994), pp. 287–314.

[5] P. Comon and B. Mourrain, *Decomposition of quantics in sums of powers of linear forms*, Signal Process., 53 (1996), pp. 93–108.

[6] P. Comon and G. Golub, *Tracking a few extreme singular values and vectors in signal processing*, Proc. IEEE, 78 (1990), pp. 1327–1343.

[7] L. De Lathauwer, *Signal Processing Based on Multilinear Algebra*, Ph.D. thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 1997.

[8] L. De Lathauwer, B. De Moor, and J. Vandewalle, *Dimensionality reduction in higher-order-only ICA*, in Proceedings of IEEE Signal Processing Workshop on HOS, Banff, Alberta, Canada, IEEE Computer Society, Los Alamitos, CA, 1997, pp. 316–320.

[9] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278.

[10] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A Multilinear Hessenberg Decomposition*, Tech. Report 99-24, SISTA, ESAT, Katholieke Universiteit Leuven, 1999.

[11] L. De Lathauwer, P. Comon, B. De Moor, and J. Vandewalle, *Higher-order power method—Application in independent component analysis*, in Proceedings of NOLTA'95, Las Vegas, NV, 1995, pp. 91–96.

[12] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[13] P. M. Kroonenberg, *Three-Mode Principal Component Analysis*, DSWO Press, Leiden, The Netherlands, 1983.

[14] 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.

[15] 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.

[16] 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.

[17] C. L. Nikias and J. M. Mendel, *Signal processing with higher-order spectra*, IEEE Signal Process. Mag., 10 (1993), pp. 10–37.

[18] A. Swami and G. Giannakis, *Bibliography on HOS*, Signal Process., 60 (1997), pp. 65–126.

[19] J. ten Berghe, J. de Leeuw, and P. M. Kroonenberg, *Some additional results on principal components analysis of three-mode data by means of alternating least squares algorithms*, Psychometrika, 52 (1987), pp. 183–191.