ELSEVIER

Computational Statistics & Data Analysis 31 (1999) 255–278

COMPUTATIONAL STATISTICS & DATA ANALYSIS

www.elsevier.com/locate/csda

A general algorithm for obtaining simple structure of core arrays in N-way PCA with application to fluorometric data

Claus A. Andersson[a,*], Rene Henrion[b]

[a] *Royal Veterinary and Agricultural University, Department of Food Technology, Chemometrics Group, Rolighedsvej 30, DK-1958 Frederiksberg, Denmark*
[b] *Weierstraß Institute for Applied Analysis and Stochastics, Mohrenstr. 39, D-10117 Berlin, Germany*

Received 1 February 1998; received in revised form 1 January 1999; accepted 18 March 1999

Abstract

Simplifying the structure of core arrays from N-way PCA or Tucker3 models is desirable to allow for easy interpretation of the factor estimates. In the present paper, first a general algorithm for maximizing a differentiable goal function depending on a set of orthogonal matrices is formulated and then specified to the problem of estimating orthonormal transformation matrices for rotating core arrays to simpler structure. The generality of the chosen approach allows to cope with all possible transformation criteria by just changing one command in the implementation. In particular, the classical body- and slice-wise diagonalization of core arrays as well as the recently proposed maximization of the variance of squared entries are covered. The stability of the algorithm is addressed by a simulation study using 120 three-way core arrays of dimension (4,4,4). Each core array instantiates a class of 50 equivalent cores by random orthonormal transformations. Theoretically, each core within a given class has the same optimum with respect to the chosen criterion, and the ability of the algorithm to provide that result has been investigated. The algorithm proves to work with a high degree of stability and consistency in optimizing the three discussed goal functions. In addition, theoretical convergence results of the algorithm are provided. In particular, monotonic convergence of functional values and convergence of iterates towards a stationary solution are proven. To illustrate the effect of maximizing the variance-of-squares and the functionality of the algorithm, the proposed method is applied to a three-way data array from fluorometric analysis of fractions obtained from low-pressure chromatographic separation of a preliminary sugar product, *thick juice*. A significant gain in simplicity is achieved, and in particular optimizing variance-of-squares provides a simple core structure for the data

[*] Corresponding author. Fax: +45-352-83502.
E-mail addresses: claus@andersson.dk (C.A. Andersson), henrion@wias-berlin.de (R. Henrion)

PII: S 0167-9473(99)00017-1

under investigation. The proposed algorithms for maximizing variance-of-squares, body diagonality and slice-wise diagonality have been implemented in MATLAB and are available by contact to the authors.

Keywords: N-way principal component analysis; N-mode factor analysis; Multi-linear modelling; Tucker3 model; Core array; Body diagonality; Variance-of-squares; Factor rotation; Orthogonal transformation; Exploratory modelling; PCA

1. Introduction

Having its roots in the field of psychometrics, the Tucker3 model of N-way principal component analysis (PCA), see Tucker (1966) and Kapteyn et al. (1986), is applied more and more often within chemometrics in the context of multivariate calibration or explanatory data analysis, see e.g. De Ligny et al. (1984), Zeng and Hopke (1990), Smilde (1992), Henrion et al. (1997) and Andersson et al. (1997). In both cases, a huge amount of data, arranged in higher-dimensional arrays, is produced by modern analytical devices. N-way PCA serves as one possible tool for subsequent data reduction. The corresponding model reads as (see Magnus and Neudecker, 1988 for details):

$$vec\,\mathbf{X} \approx (\mathbf{A}_1 \otimes \cdots \otimes \mathbf{A}_N) vec\,\mathbf{C}. \tag{1}$$

Here, $\mathbf{X}$ represents the N-way data array of order $(n_1, \ldots, n_N)$ and $\mathbf{A}_i$ of order (n_i, s_i) is the orthonormal component matrix belonging to the ith way. The array $\mathbf{C}$ of order $(s_1, \ldots, s_N)$ designates the core array, while vec and $\otimes$ refer to vectorization and Kronecker product, respectively.

A specific aspect of the N-way PCA model is its non-uniqueness in the sense that the factors, together with the core array, can be rotated without loss of fit: Transforming each of the component matrices $\mathbf{A}_i$ in (1) to $\mathbf{A}_i\mathbf{P}_i$ by means of orthonormal matrices $\mathbf{P}_i$ of order (s_i, s_i), the same approximation to the data array in (1) is obtained when transforming the original core array $\mathbf{C}$ to

$$vec\,\tilde{\mathbf{C}} = (\mathbf{P}_1^{\mathrm{T}} \otimes \cdots \otimes \mathbf{P}_N^{\mathrm{T}})\, vec\,\mathbf{C}. \tag{2}$$

The resulting core, designated by $\tilde{\mathbf{C}}$, is of equal order as $\mathbf{C}$. For later argumentation, it is important to note that the sum of squared elements of core arrays is invariant under the above transformation.

The core array provides a way to interpret the solutions since its squared entries represent the relative importance of the factor combinations from different (orthonormal) component matrices in terms of explained variability. Therefore, it is desirable to have a few significant entries in the core array allowing for easy identification of the significant factor combinations. Such factor combinations will reflect the latent behaviour or pattern in the data. But, often the core array does not facilitate direct interpretation because the squared entries are of equal magnitude giving no direct pointer to major trends and systematics in data. Then, the rotational degree of freedom described by (2) may be used to accommodate for this situation. A common feature of different approaches in this direction is the aim of giving the core a simple structure by optimizing a well-defined goal function that quantifies the simplicity of the core.

Much of the work devoted to increasing the interpretability of the N-way PCA model has been concerned with estimating orthogonal rotation matrices that could transform the solution to give a more unambiguous interpretation, see Kiers (1992). The present work will focus on the common algorithmic aspect of applying orthonormal core transformations (2) for optimizing any differentiable criterion of core simplicity (for the latest work on *oblique* rotations the reader is referred to Kiers, 1999). Special attention will be paid to the variance-of-squares criterion as a recently proposed goal function, see Henrion and Andersson (1999), as well as to some more classical diagonalization criteria. The potential of the presented approach lies in its generality, so for a new criterion of core simplicity, no specific algorithm has to be re-designed. The stability of the algorithm is illustrated by application to a large amount of synthesized, well-characterized, cores. Furthermore, theoretical convergence properties are studied. The discussion concludes with an application to data collected at-line in industrial production of sugar.

2. Criteria for simple-structure transformations

The squared core entries reflect the significance of the factor combinations in the model. In order to allow for easy and correct interpretation, it is desirable to obtain as simple a core structure as possible. If the core can be brought to a simple structure where only a few but very large elements are present, the analyst may focus on these respective factor combinations. The worst case is the situation where all elements in the core are equal, thereby indicating that no significant single factor combination could be found. The concept of rotating core arrays from three-way PCA originates from Tucker (1966) and the field of multidimensional scaling, e.g. De Leeuw and Pruzansky (1978) and Carroll and Wish (1974). For the moment we will leave out of discussion *how* the measures are maximized and focus on the goal functions.

Classical criteria of core simplicity refer to diagonal shapes. Understanding diagonality of a square N-way core array of order $(s,\dots,s)$ in a strict sense means that all non-zero elements should be located on the so-called body diagonal of the array, i.e. $C_{i_1,\dots,i_N}=0$ unless $i_1=\cdots=i_N$. In general, of course, core arrays cannot be transformed via (2) to exact body diagonality. All one can do is to maximize the sum of the squared entries on the body diagonal:

$$\max\sum_{i=1}^{s} C_{i,\dots,i}^2. \tag{3}$$

Since the total sum is invariant under the transformation (2), this will simultaneously minimize the off-diagonal sum of squares, hence body diagonal shape is approached as close as possible. An algorithm for maximizing the body diagonality of three-way cores has been proposed by Kiers (1992). The whole approach applies to square N-way cores of order $(s,\dots,s)$ only. An N-way PCA model with all off-diagonal core elements being zero corresponds to the N-way PARAllel FACtors (PARAFAC) model (Harshman, 1970) and the CANonical DECOMPosition (CANDECOMP) model (Carroll and Chang, 1970), with the factors being constrained to orthogonality. The term *degree of diagonality* refers to the ratio between the sum of squares of the diagonal elements and the total sum of squares of the core array. According to the statements above, this degree has values between zero and one (exact body diagonality), and it may serve to compare the diagonality structure of cores with different total sum of squares.

A weaker concept of diagonality refers to slices of the core array along one fixed, say the Nth, mode. In order to give sense to the concept of slice diagonality, the $(N-1)$-dimensional slices of the core have to be square arrays, i.e. the core has to have the order $(s,\dots,s,s_N)$. For $N=3$, the slices are square matrices then, but the entire array need not be square. For slice-wise diagonal cores, the N-way PCA model reduces to a PARAFAC model again, but now with factors that are not necessarily independent. An algorithm for slice-wise diagonalization of 3-way arrays has been proposed by Kroonenberg (1983). The goal function to be maximized now becomes

$$\max \sum_{j=1}^{s_N} \sum_{i=1}^{s} C_{i,\dots,i,j}^2. \tag{4}$$

In analogy with diagonality, the *degree of slice-wise diagonality* refers to the ratio between the sum of squared slice-wise diagonal elements and the total sum of squared core elements.

Both of the diagonalization approaches focus on optimizing pre-defined elements in the core array, hence, it is implicitly assumed that the data are well described by these respective factor combinations. Possibly significant off-diagonal entries are not maximized. The variance-of-squares measure, recently introduced in Henrion and Andersson (1999), allows to detect significant factor combinations without using any *a priori* assumption on the structure like diagonality. This more flexible approach to core simplification usually leads to a smaller number of significant core entries than with diagonalization procedures. Of course, an interpretation in terms of PARAFAC, as given above, fails then, since the significant elements can be located anywhere in the core. The criterion to be maximized measures the variance of the squared core entries:

$$\max \sum_{i_1=1}^{s_1} \cdots \sum_{i_N=1}^{s_N} (C_{i_1,\dots,i_N}^2 - \bar{C})^2, \tag{5}$$

$$\bar{C} = \prod_{i=1}^{N} s_i^{-1} \sum_{i_1=1}^{s_1} \cdots \sum_{i_N=1}^{s_N} C_{i_1,\dots,i_N}^2. \tag{6}$$

In contrast to any measures of diagonality the variance-of-squares is defined for cores that are non-square. To summarize, Fig. 1 depicts what elements are used during optimization of the three goal functions. Fig. 1a illustrates two elements on a body diagonal of an array of order (2,2,2). Accordingly, Fig. 1b shows the diagonal elements taken slice-wise in the third way. In Fig. 1c the variance-of-squares expression is indicated by letting all entries in the core array contribute to the goal function.

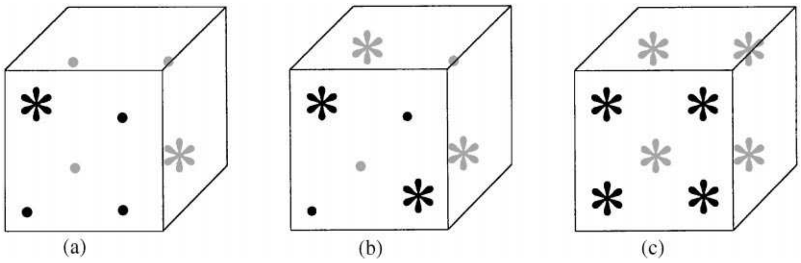

Fig. 1. The differences between the three discussed goal functions for a core array of order (2,2,2) are depicted (a) Maximizing sum of squares of the body diagonal elements, (b) maximizing sum of squares of the slice-wise diagonal elements and (c) maximizing the variance-of-squares using all elements in the core.

In accordance with the diagonality criteria, it would be desirable to define a degree for the variance-of-squares criterion in order to compare different cores. For the diagonality criteria, the maximum possible value which could be obtained within a class of cores of common order and having equal sum of squares is the total sum of squares. Since this value and, hence, the mean value in (6), are invariant under the transformation (2), it is easy to show that the theoretical maximum of the variance-of-squares criterion is attained in the situation where all core elements but one are zero. Then, the non-zero element has to account for the total sum of squares of the core, which is the constant $p\bar{C}$, where $p = \prod_{i=1}^{N} s_i$ refers to the total number of elements (cf. (6)). Therefore, the variance of squares for such a core equals $(p\bar{C} - \bar{C})^2$ (deviation from mean of the non-zero element) plus $(p-1)(0-\bar{C})^2$ (deviation from mean of the $p-1$ zero elements) which gives $p(p-1)\bar{C}^2$. In general, a given core cannot be transformed into one with a single non-zero element only, hence this situation is the theoretical limit which the actual transformation may be related to. Due to the invariance of the mean value $\bar{C}$, this limit can be calculated from any given core. Now, the *degree of variance-of-squares* is defined as the ratio between the actual variance-of-squares and the theoretical maximum $p(p-1)\bar{C}^2$.

3. An algorithm for optimal orthogonal core transformations

In this section, we develop a general algorithm for finding an optimal orthonormal N-way core transformation according to a specific criterion. In particular, the above-described variance-of-squares maximization, and also the classical body and slicewise diagonalization are included. Since all these transformations can be simultaneously realized by a procedure with common basic structure, we establish a general-purpose algorithm first, which applies to the optimization of any (differentiable) criterion of orthonormal matrices and not just to the three special cases mentioned above in the context of core transformations.

3.1. Proposal for a general-purpose algorithm

Denote by $\mathcal{O}(n)$ the manifold of orthogonal matrices of order (n,n) and consider the optimization problem

$$(P)\quad \max\{v(P_1,\dots,P_N)|P_i\in\mathcal{O}(n_i);\ i=1,\dots,N\},$$

where $v:\mathcal{M}(n_1)\times\cdots\times\mathcal{M}(n_N)\to\mathbb{R}$ is a differentiable function and $\mathcal{M}(n)$ refers to the space of matrices of order (n,n). The orthogonality constraints above may be written as $P_i^{\mathrm{T}}P_i=I_{n_i}$ $(i=1,\dots,N)$. Denoting by A_i $(i=1,\dots,N)$ any multiplier matrix, we define the Lagrangian function

$$f:\mathcal{M}(n_1)\times\cdots\times\mathcal{M}(n_N)\times\mathcal{M}(n_1)\times\cdots\times\mathcal{M}(n_N)\to\mathbb{R}$$

via

$$f(P_1,\dots,P_N,A_1,\dots,A_N)=v(P_1,\dots,P_N)-\sum_{i=1}^{N}tr\,[A_i(P_i^{\mathrm{T}}P_i-I_{n_i})].$$

Now, since the orthogonality constraints define a regular surface in $\mathcal{M}(n_1)\times\cdots\times\mathcal{M}(n_N)$, it follows that, if $(\bar{P}_1,\dots,\bar{P}_N)$ is a solution of the Problem (P), then there exist *symmetric* multiplier matrices Λ_i $(i=1,\dots,N)$ (see Magnus and Neudecker, 1988), such that $(\bar{P}_1,\dots,\bar{P}_N,\Lambda_1,\dots,\Lambda_N)$ is a stationary point of f (i.e. the derivative of f vanishes at that point).

Writing down the stationary conditions gives

$$\frac{\partial v}{\partial P_i}(\bar{P}_1,\dots,\bar{P}_N)-2\bar{P}_i\Lambda_i=0\quad(i=1,\dots,N),\tag{7}$$

$$\bar{P}_i^{\mathrm{T}}\bar{P}_i-I_{n_i}=0\quad(i=1,\dots,N).\tag{8}$$

Here, we made use of the convention that $\partial v/\partial P_i$ is a matrix of same order as P_i with general entry $(\partial v/\partial P_i)_{kl}=\partial v/\partial p_{kl}$, where the last expression refers to the usual partial derivative of v with respect to the general entry p_{kl} of P_i. This special arrangement of partial derivatives is useful in the context of matrix calculus. Later, we shall also work with the conventional definition of the partial gradient $\nabla_{P_i}v$ considered as a linear function assigning to each $Q\in\mathcal{O}(n_i)$ the scalar

$$\langle\nabla_{P_i}v,Q\rangle=\sum_{k,l}\frac{\partial v}{\partial p_{kl}}q_{kl}.$$

From here, the following relation between the two notions is obvious:

$$\langle\nabla_{P_i}v,Q\rangle=\mathrm{tr}\left[\frac{\partial v}{\partial P_i}Q^{\mathrm{T}}\right].\tag{9}$$

Of course, (8) means nothing else than the required orthogonality of $\bar{P}_1,\dots,\bar{P}_N$, so the interesting part is contained in (7). Multiplying the ith condition of this set from the left by $\bar{P}_i^{\mathrm{T}}$, provides (by orthogonality)

$$\bar{P}_i^{\mathrm{T}}\frac{\partial v}{\partial P_i}(\bar{P}_1,\dots,\bar{P}_N)=2\Lambda_i\quad(i=1,\dots,N).$$

From these equations it follows that for any stationary solution $(\bar{P}_1,\dots,\bar{P}_N)$ of the problem (P) the matrices on the left-hand side have to be symmetric. Conversely, if we find orthogonal $\bar{P}_i$, such that the mentioned matrices are symmetric, then we have obtained a stationary solution of problem (P). This follows after left-multiplication of the above relation by $\bar{P}_i$ leading back to (7) and (8) due to orthogonality of the $\bar{P}_i$. Summarizing, $(\bar{P}_1,\dots,\bar{P}_N)$ is a stationary solution of problem (P) if and only if the matrices

$$\bar{P}_i^{\mathrm{T}} \frac{\partial v}{\partial P_i}(\bar{P}_1,\dots,\bar{P}_N) \tag{10}$$

are symmetric for $i=1,\dots,N$. Therefore, it is desirable to have an algorithm iterating on orthogonal P_i, thereby ‘symmetrifying’ the above matrices. This is realized by the following algorithm:

Algorithm 1.

```
1. Set $P_i^0 := I_{n_i}$ $(i = 1,\dots,N)$ and $k := 0$
2. Set $k := k + 1$ and $i := 0$
3. Set $i := i + 1$ and compute an orthogonal matrix $P_i^k := U^{\mathrm{T}}V^{\mathrm{T}}$, such that
       $U\left[\frac{\partial v}{\partial P_i}(P_1^k,\dots,P_{i-1}^k,P_i^{k-1},\dots,P_N^{k-1})\right] V = \mathrm{diag}[d_1,\dots,d_{n_i}]$,
   where $U,V \in \mathcal{O}(n_i)$, $d_1 \geq \cdots \geq d_{n_i} \geq 0$ (i.e. $U$ and $V$ provide a singular value
   decomposition of the derivative matrix). If $i < N$, then goto 3.
4. If $v(P_1^k,\dots,P_N^k)$ significantly differs from $v(P_1^{k-1},\dots,P_N^{k-1})$, then goto 2.
5. Stop
```

The motivation behind step 3 is that it provides a symmetrification in the sense of (10). Indeed, one has

$$P_i^{k\mathrm{T}}\left[\frac{\partial v}{\partial P_i}(P_1^k,\dots,P_{i-1}^k,P_i^{k-1},\dots,P_N^{k-1})\right] = V\,\mathrm{diag}[d_1,\dots,d_{n_i}]V^{\mathrm{T}} = S$$

where S is a symmetric matrix.

Note that the proposed method does not depend on the concrete structure of the function v to be maximized in problem (P), therefore it applies as a general-purpose algorithm for maximizing (or minimizing after passing to $-v$) a differentiable function of N orthogonal matrices of possibly differing orders.

3.2. Application to core transformations

Now, we are going to specialize the developed general algorithm to the case of core transformations. All one has to do, according to the preceding section, is to calculate the partial derivatives $\partial v/\partial P_i$ of the corresponding criteria v with respect to the transformation matrices P_i. This turns out to be rather difficult, however, when evaluating at general current iterates whereas it is quite easy to compute at identity matrices. In the following, we shall develop an appropriate modification of the algorithm described above taking into account the specific structure of core transformations.

It is important to note that, in the context of core transformation, the criteria depend on the transformation matrices in a composite way: the criterion is a function of the core array which in turn depends on the transformation matrices. Given a core array C and orthonormal matrices $P_1,\dots,P_N$, we denote the core array transformed according to (2) by

$$T(P_1,\dots,P_N;C) = (\mathbf{P}_1^{\mathrm{T}} \otimes \cdots \otimes \mathbf{P}_N^{\mathrm{T}}) vec\,\mathbf{C}. \tag{11}$$

Now, the criterion as a function of transformation matrices writes as a composition

$$v(P_1,\dots,P_N) = \tilde{v}(T(P_1,\dots,P_N;C^0)),$$

where C^0 is the original core array and $\tilde{v}$ denotes the criterion as a function of the core array. For the three transformations to be considered here, one has

$$\text{variance of squares}\quad \tilde{v}_1(C) = \sum_{i_1=1}^{s_1} \cdots \sum_{i_N=1}^{s_N} (C^2_{i_1,\dots,i_N} - \bar{C})^2, \tag{12}$$

$$\text{body diagonality}\quad \tilde{v}_2(C) = \sum_{i=1}^{s} C^2_{i,\dots,i}, \tag{13}$$

$$\text{slice diagonality}\quad \tilde{v}_3(C) = \sum_{j=1}^{s_N} \sum_{i=1}^{s} C^2_{i,\dots,i,j}. \tag{14}$$

Let us consider the very first step ($k=1, i=1$) of the algorithm above: The initial transformation matrices are identity matrices and in step 3 one has to compute the partial derivative

$$\frac{\partial v}{\partial P_1}(I_{s_1},\dots,I_{s_N}) = \frac{\partial \tilde{v}}{\partial C}(C^0)\,\frac{\partial T}{\partial P_1}(I_{s_1},\dots,I_{s_N};C^0) \tag{15}$$

according to the chain rule. The right-hand side matrices are easily calculated as will be seen later on. First note, however, that in the following iteration ($k=1, i=2$) of the algorithm, the partial derivative is no longer taken at a complete set of identity matrices but at $(P_1^1, I_{s_2},\dots,I_{s_N})$, where P_1^1 is the current iterate obtained in step 3 of the previous iteration. So, in the course of iterations, the convenient possibility of evaluating the partial derivatives at identity matrices gets lost. Yet, by a simple modification, this difficulty may be overcome. Let us illustrate this for the second iteration: Define a function

$$v^*(P_1,\dots,P_N) := v(P_1^1 P_1, P_2,\dots,P_N).$$

Obviously, the maximization of v^* is equivalent to the maximization of v, since any solution of the one criterion is immediately transformed into a solution of the other.

Therefore, instead of continuing the maximization of v as proposed in the original algorithm (with $k=1, i=2$), one may restart the whole algorithm at the beginning, but now maximizing v^* and iterating on P_2 instead. Starting again with identity matrices means to keep the current value of the old criterion, since $v^*(I_{s_1},\dots,I_{s_N}) = v(P_1^1, I_{s_2},\dots,I_{s_N})$. From the definitions, one gets

$$v^*(P_1,\dots,P_N) = \tilde{v}(T(P_1^1 P_1,\dots,P_N; C^0)) = \tilde{v}(T(P_1,\dots,P_N; C^1)),$$

where $C_1 = T(P_1^1, I_{s_2},\dots,I_{s_N}; C^0)$ is the updated core array after applying the transformation matrices $P_1^1, I_{s_2},\dots,I_{s_N}$ to the original core C^0. In order to apply step 3 of the algorithm, one has to calculate now the partial derivative $\partial v^*/\partial P_2$ at the identity matrices, so – again by the chain rule – it results

$$\frac{\partial v^*}{\partial P_2}(I_{s_1},\dots,I_{s_N}) = \frac{\partial \tilde{v}}{\partial C}(C^1)\,\frac{\partial T}{\partial P_2}(I_{s_1},\dots,I_{s_N}; C^1).$$

Now it is clear how to proceed: calculate the second transformation matrix P_1^2 as to symmetrize the matrix $P_1^{2\mathrm{T}}(\partial v^*/\partial P_2)(I_{s_1},\dots,I_{s_N})$ (compare step 3 of the algorithm), update the core array by $C^2 = T(I_{s_1}, P_1^2, I_{s_3},\dots,I_{s_N}; C^1)$, and, in the next iteration evaluate the partial derivative according to

$$\frac{\partial \tilde{v}}{\partial C}(C^2)\,\frac{\partial T}{\partial P_3}(I_{s_1},\dots,I_{s_N}; C^2)$$

(without explicit reference to a newly defined v^{**}). In this way, one gets a sequence of core arrays maximizing the considered criterion.

Summarizing, the following algorithm for optimal core transformation with respect to one of the three criteria $\tilde{v}$ introduced above is proposed:

Algorithm 2.

```
1. Set $C^{\mathrm{new}} := C^0$ (=original core array), $P_j^{\mathrm{new}} := I_{s_j}$ $(j = 1,\dots,N)$ and $k := 0$
2. Set $k := k + 1$ and $j := 0$
3. Set $j := j + 1$, $C^{\mathrm{old}} := C^{\mathrm{new}}$, $P_j^{\mathrm{old}} := P_j^{\mathrm{new}}$ and compute an orthonormal matrix $P := U^{\mathrm{T}} V^{\mathrm{T}}$ such that
       $U\left[\frac{\partial \tilde{v}}{\partial C}(C^{\mathrm{old}})\,\frac{\partial T}{\partial P_j}(I_{s_1},\dots,I_{s_N}; C^{\mathrm{old}})\right] V = \mathrm{diag}\,[d_1,\dots,d_{n_i}],$
   where $U, V \in \mathcal{O}(n_i)$, $d_1 \geq \cdots \geq d_{n_i} \geq 0$.
   Set $C^{\mathrm{new}} := T(I_{s_1},\dots,I_{s_{j-1}}, P, I_{s_{j+1}},\dots,I_{s_N}; C^{\mathrm{old}})$ and $P_j^{\mathrm{new}} := P_j^{\mathrm{old}} P$. If $j < N$, then goto 3.
4. If $\tilde{v}(C^{\mathrm{new}})$ significantly differs from $\tilde{v}(C^{\mathrm{old}})$, then goto 2.
5. Stop
```

The transformation matrices, leading from the original core array C^0 to the final core array C^{new} are given by P_j^{new}, i.e. $C^{\mathrm{new}} = T(P_1^{\mathrm{new}},\dots,P_N^{\mathrm{new}}; C^0)$. Step 3 is performed by singular-value decomposition as in Algorithm 1, so it remains to compute the matrix in brackets. The general element of the second factor in (15), which is common to all procedures, is obtained as

$$\left[\frac{\partial T_{i_1,\dots,i_N}}{\partial P_j}(I_{s_1},\dots,I_{s_N}; C^{\mathrm{old}})\right]_{k,l} = \begin{cases} C^{\mathrm{old}}_{i_1,\dots,i_{j-1},k,i_{j+1},\dots,i_N} & l = i_j, \\ 0 & l \neq i_j. \end{cases}$$

The general element of the first factor calculates for the three criteria according to

$$\begin{aligned}
\left[\frac{\partial\tilde{v}_1}{\partial C}(C^{\text{old}})\right]_{i_1,\dots,i_N} &= 4(C^{2\ \text{old}}_{i_1,\dots,i_N} - \bar{C})\,C^{\text{old}}_{i_1,\dots,i_N},\\
\left[\frac{\partial\tilde{v}_2}{\partial C}(C^{\text{old}})\right]_{i_1,\dots,i_N} &= \begin{cases} 2C^{\text{old}}_{i_1,\dots,i_1} & i_1=\cdots=i_N,\\ 0 & \text{else}\end{cases}\\
\left[\frac{\partial\tilde{v}_3}{\partial C}(C^{\text{old}})\right]_{i_1,\dots,i_N} &= \begin{cases} 2C^{\text{old}}_{i_1,\dots,i_1,i_N} & i_1=\cdots=i_{N-1},\\ 0 & \text{else.}\end{cases}
\end{aligned}$$

Now, the expressions in brackets in step 3 become (by multiplication of the corresponding factors) for the three different methods

$$[\ ]^1_{k,l} = 4\sum_{i_1=1}^{s_1}\cdots\sum_{i_{j-1}=1}^{s_{j-1}}\sum_{i_{j+1}=1}^{s_{j+1}}\sum_{i_N=1}^{s_N}(C^{2\ \text{old}}_{i_1,\dots,i_{j-1},l,i_{j+1},\dots,i_N} - \bar{C})\,C^{\text{old}}_{i_1,\dots,i_{j-1},l,i_{j+1},\dots,i_N}\,C^{\text{old}}_{i_1,\dots,i_{j-1},k,i_{j+1},\dots,i_N},$$

$$[\ ]^2_{k,l} = 2C^{\text{old}}_{l,\dots,l}\cdot C^{\text{old}}_{l,\dots,l,k,l,\dots,l}\quad (k \text{ at position } j),$$

$$[\ ]^3_{k,l} = \begin{cases} 2\sum_{i_N=1}^{s_N} C^{\text{old}}_{l,\dots,l,i_N}\cdot C^{\text{old}}_{l,\dots,l,k,l,\dots,l,i_N}\ (k \text{ at position } j) & \text{if } j<N,\\ 2\sum_{i=1}^{s} C^{\text{old}}_{i,\dots,i,l}\cdot C^{\text{old}}_{i,\dots,i,k} & \text{if } j=N.\end{cases}$$

It is interesting to note that the matrix $[\ ]^3_{k,l}$ is automatically symmetric for $j=N$. Henceforth, the slice-wise diagonality remains unaffected by rotation for $j=N$, and with regard to algorithmic efficiency this last inner iteration should be omitted from the optimization scheme.

4. Validation of the algorithm

A large quantity of well-characterized core arrays have been simulated for the purpose of assessing the robustness of the proposed algorithm with respect to finding global, rather than local, optima. The core arrays have been synthesized especially for investigating the ability of the algorithm to find the global optima of the three discussed goal functions; variance-of-squares, diagonality and slice-wise diagonality. The amount and features of cores required for such an analysis can only be provided by synthesis.

4.1. Experimental

A number of 120 core arrays of dimensions (4,4,4) with random elements in the range -100 to $+100$ were synthesized. Each of the 120 synthesized cores were used to establish a class containing 50 core arrays by random orthonormal transformations of the same synthesized core array as described by (2). This ensures that all 50 core arrays within one class can be obtained from each other by an orthonormal transformation, and they are equal in this sense. By comparing the values of the 50 optimized measures within each class, an estimate can be made towards the ability of the algorithm to locate the global optimum. Rotated cores within each class have the same optimal value with regard to the three investigated measures. However, preliminary calculations on 80 simulated cores showed that for 11 core arrays the optimal value of the goal function was not found in approx. 10% of the cases. Thus, to enhance the probability of locating the global optimum, the algorithm was restarted 5 times with each core using random initial orthonormal rotation matrices. Additional restarts were performed until the two largest values of the goal function differed less than 1%. This scheme was used throughout the calculations and appears to be a feasible approach to the problem of non-global optima.

Computations were performed on a DELL 200 MHz Pentium Pro running MATLAB 5.1.0.421 under Windows NT 4.0. The MATLAB built-in function rand() was used for the purpose of generating random numbers.

4.2. Results

The results from applying the proposed algorithm to the synthesized cores are depicted in Fig. 2a–c. For each class two groups of core arrays are available; the

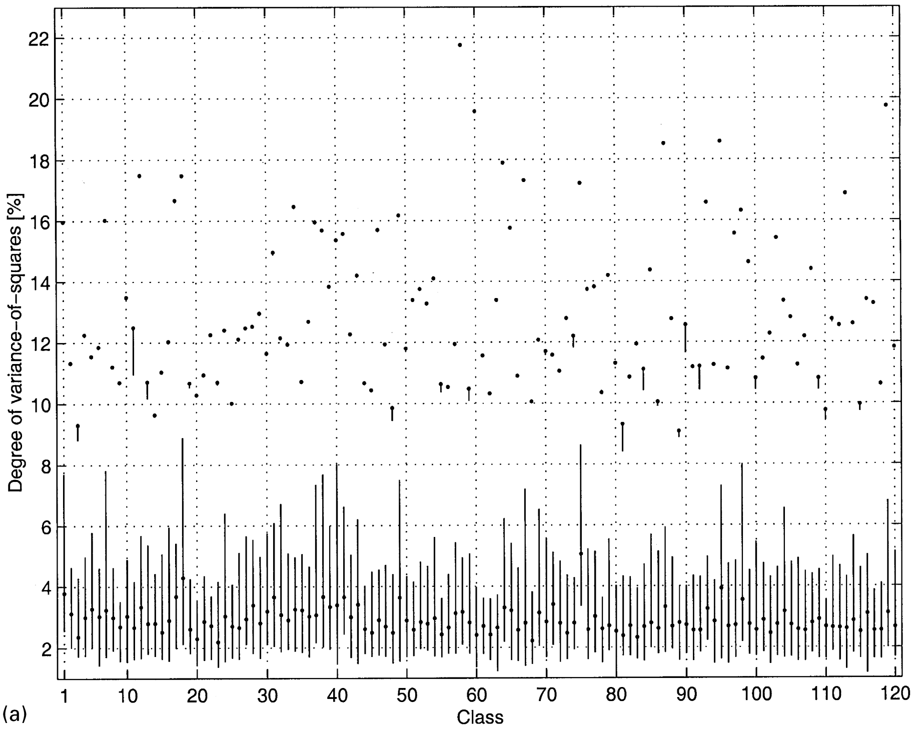

Fig. 2. Summary of the 120 classes each containing 50 cores derived from the same synthesized core array by random orthonormal transformations. The figure depicts the distribution of un-optimized and optimized goal function values for (a) variance-of-squares, (b) body diagonality and (c) slice-wise diagonality. The vertical line indicates the range from the minimum value to the highest value of the goal function. The dots indicate the medians of the two sets. See Section 4.2 for a detailed discussion.

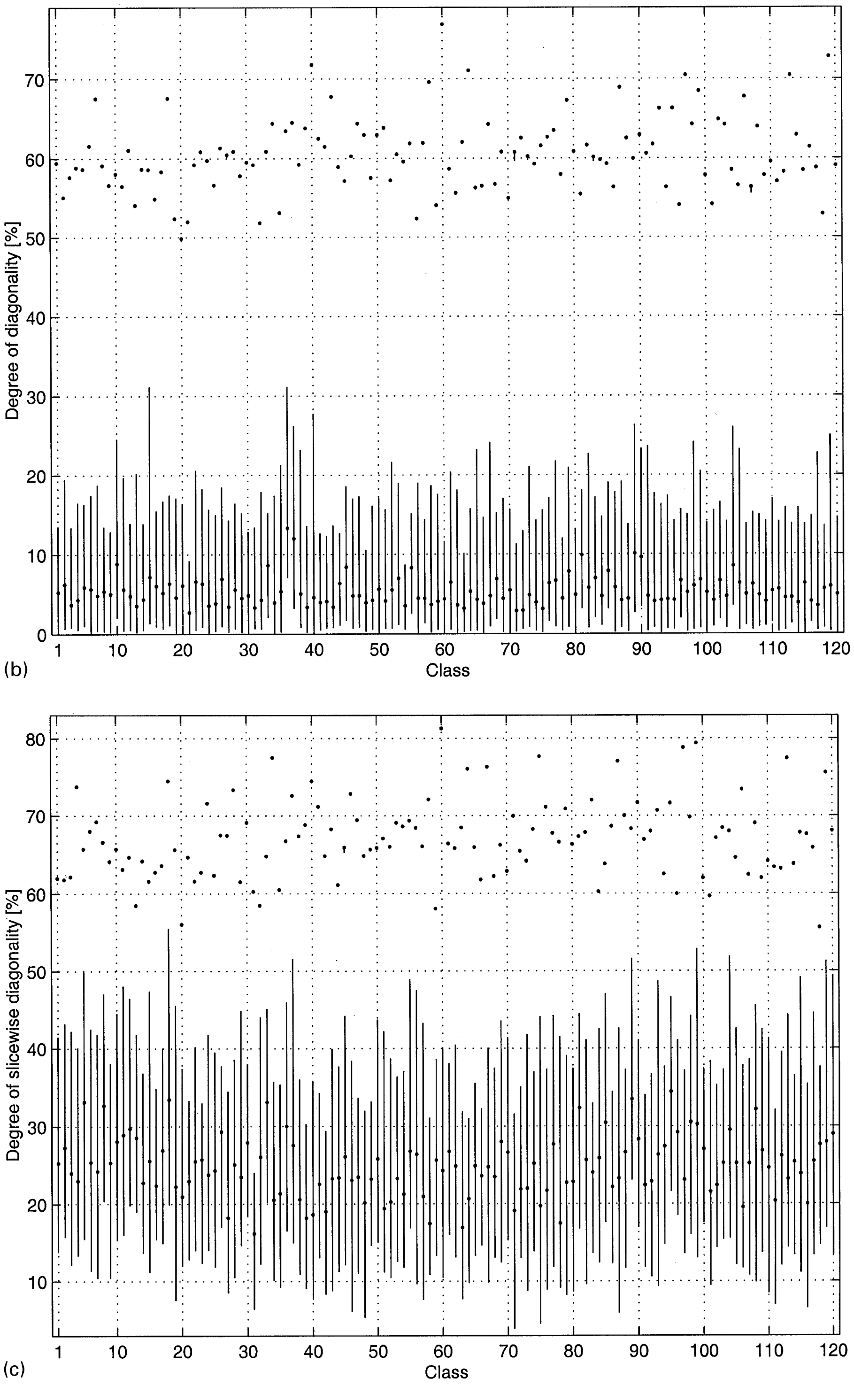

Fig. 2. Continued.

50 un-optimized core arrays and their optimized equivalents. The differences in the goal function values of the two groups are illustrated in Fig. 2a–c. For each distribution a vertical line connects the lowest observed value with the highest observed value and serves to illustrate the range of observations. The dot on each vertical line depicts the median of the observations. The goal function values for the un-optimized core arrays are, as expected, lower than the values of the same optimized core arrays. This is seen as a clear-cut separation between the two groups; the goal function values of the un-optimized core arrays are clearly lower and more spread than the goal function values of the core upon maximization. The function values upon optimization are in most cases so similar that there is no difference between the lowest and the highest of the returned goal function values. The gain of optimization is illustrated by the large differences between the respective measures before and after applying the algorithm. In addition, there is no overlap of the highest values of un-optimized cores with the lowest values of optimized cores, thus, all cores have gained in goal function value. Fig. 2a illustrates the degree of variance-of-squares before and after optimization. There generally is a tri-fold gain for this measure, providing a significant gain in simplicity for all classes. Within some classes, the optimal variance-of-squares core arrays obtained by the algorithm differ significantly in function values. E.g., for class no. 11 at least one of the returned cores have a suboptimal function value at approx. 11%, whereas the median clearly shows that the large part of the estimated optima are equal in value at approx. 12.5%. An important observation is that for all classes the median is similar to the highest value, this indicating, that by applying the algorithm several times a good estimate on the global optimum is found as the highest value. Fig. 2b represents the parameters for the optimization of the body diagonality. For the body diagonality version of the algorithm, the ranges within classes of the calculated optima are quite small. This observation confirms what was apparent during iterations; the optimal degrees of body diagonality within classes were more similar than for the values for variance-of-squares. The median of the distributions typically increase 8 times by optimization. Fig. 2c depicts the parameters for the optimization of slice-wise diagonality. The calculated optima are very close within classes, hence the algorithm for slice-wise diagonalization is slightly more stable in providing the global optima. This behaviour may be explained as follows: since there is no transformation matrix for the last mode, there is one less derivative matrix to return a non-global optimum and the algorithm is less prone to obtain a suboptimal rotation.

5. A three-way PCA of fluorometric measurements of thick juice

To exemplify the principle of maximizing the variance-of-squares and the use of the algorithm described in Section 3, we will apply the method to a core array derived from fluorometric measurements. To keep the focus on the proposed method we will restrict ourselves to discuss solely the core array and leave out detailed chemical interpretation.

In northern Europe white crystalline suger is produced from sugar beets, i.e. *Beta Vulgaris*. The process is extremely complex and many of the unit processes involve recycled streams, see Larsson (1989). At different stages in the production, colour is formed due to combined effects of pH, temperature and the natural presence of colour precursors, polyphenolic oxidases, phenolic amino acids, carbonylic components and amino-N. The colour is a quality parameter which, in part, has influence on the classification of the final crystalline sugar product. From an economical standpoint it is therefore of great importance to be able to automatically control the operating conditions to give the whitest possible sugar and the most uniform product. Among the many possible intermediary products *thick juice* was chosen as a potential indicator of the degree of colouration in the final sugar. Thick juice is comparable in colour and viscosity to syrup. Spectrofluorometry has been selected for screening due to its sensitivity towards phenolic compounds and, to some extent, amino acids. See Nørgaard (1995) for a discussion of the suitability of spectrofluorometry as a screening method in the sugar process.

5.1. Experimental

From the 1994 production period, 15 thick juice samples were chosen. Each sample was separated into 28 fractions in a low-pressure liquid chromatography (LPLC) system. For each fraction, the fluorescence intensity for six combinations of excitation and emission wavelengths have been measured, thereby yielding a three-way array of order $(15, 28, 6)$ corresponding to (samples, fraction, filter combination). Since the sensitivity and noise levels are equal for the measured filter combinations, it was chosen not to scale the data prior to modelling. However, due to the significant differences in levels of the intensities measured over the filter combinations data were centred over the latter mode.

5.2. Results

To determine the correct dimensionality of the model, a number of three-way PCA models were calculated and the fit to the data was evaluated for each model. The dimensions ranged from one to four factors in all modes, thus, a total of 37 valid models were calculated. It applies that not all combinations of 1–4 factors are valid since the product of the two smallest dimensions of the core must be equal to, or greater than, the largest dimension. E.g., valid dimensions are $(1,2,2)$ and $(2,4,2)$, whereas $(1,1,2)$ and $(1,1,4)$ are not. The dimensionality of the model is found under consideration of parsimony, and the chosen model must describe data well with as little complexity as possible since this minimizes the risk of overfit. In order to identify the model that is optimal in this sense, the 37 models were arranged in 13 groups according to the number of parameters in the core. For each group of cores with equal number of elements, the model explaining the highest amount of variation in the data was identified. The number of core elements is of direct interest for the analyst, since the higher the number of core elements, the more factor combinations

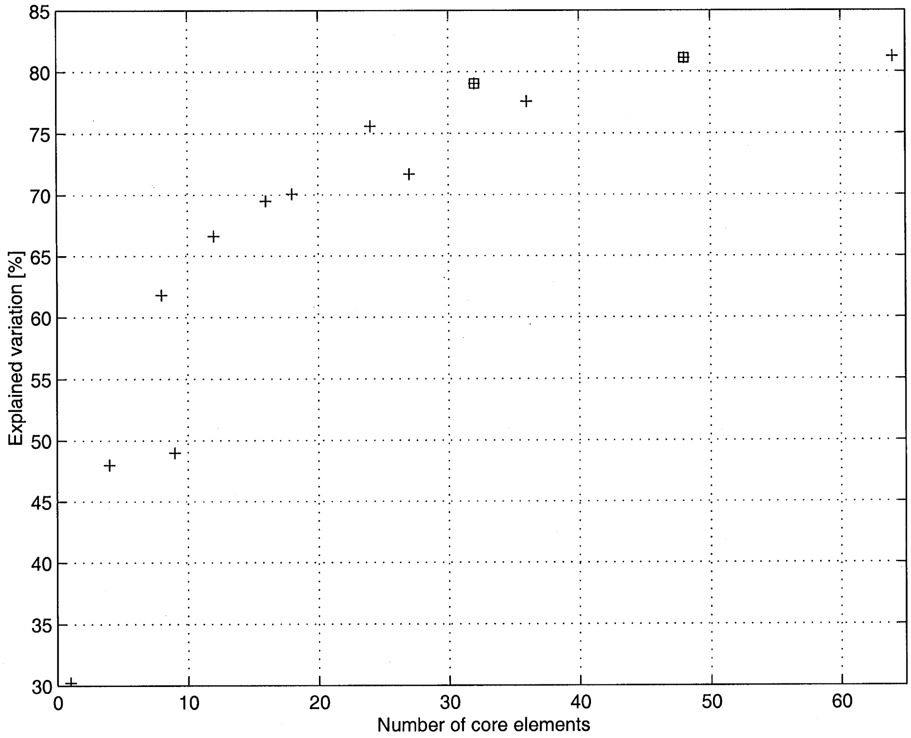

Fig. 3. A total of 37 three-way Tucker models of specified dimensions are calculated and each model is grouped according to the number of elements in the core. For each group the maximum degree of fit is plotted as a function of the number of core elements in the group. From the plot it is seen that the best model with 32 core elements explains 79.0% of the variation, see Section 5.2.

must be included in analysis and interpretation. One could undertake a view of model complexity in terms of the total number of parameters rather than just the number of parameters in the core. However, the complexity of the systematic behaviour of data is reflected by the dimensions of the core since the dimensions directly relate to the number of latent phenomenon in data. Thus, it is chosen to weigh the fit of the model against the complexity in terms of number of factors.

In Fig. 3 the highest explained variation in each of the 13 groups is plotted as a function of the number of core elements in the group. As indicated by the emphasized points, two groups are interesting; models having 32 and 48 core elements. Both these models provide a close fit to the data with a relatively low number of parameters in the core. When going from 32 core elements to 48, the explained variation increases merely from 79.0% to 81.1%. Thus, in order to make the interpretation manageable, the array with 32 elements is chosen for further analysis. The model with the highest explained variation in this group of models was found to have dimensions (4,4,2) indicating that four principal patterns prevail in the sample and fraction modes whereas two principal trends suffice to describe the variation of the filter combinations. The core array of the initial (i.e., un-rotated) model is depicted by C^{raw}:

$$C^{\mathrm{raw}} = \left(\begin{array}{cccc|cccc} 3256 & -2901 & 620 & 183 & 2702 & 2270 & -277 & -869 \\ 1986 & 1921 & 16 & 1601 & -1025 & 951 & 1632 & 152 \\ 742 & 735 & 949 & -940 & -329 & 67 & -1130 & 315 \\ -609 & 184 & 1168 & 548 & 232 & 249 & 30 & -580 \end{array}\right).$$

Bearing in mind that the squared value of any core element is proportional to the variation explained by the respective factor combination, inspection of C^{raw} reveals that there is no clear threshold allowing for a simple distinction between significant and insignificant core elements. This is a common problem when interpreting larger core arrays. Because the analyst cannot pin-point a few significant combinations of factors, interpretation may be rendered impossible. The variance-of-squares of C^{raw} is 2.22×10^{14} and the degree of variance-of-squares is 7.73%. Application of the algorithm described in Section 3 for optimizing variance-of-squares rotates C^{raw} into C^{vos} by orthonormal transformations.

$$C^{\mathrm{vos}} = \left(\begin{array}{cccc|cccc} \mathbf{4486} & 110 & -16 & -9 & 129 & \mathbf{3509} & -198 & 373 \\ 301 & \mathbf{2644} & 496 & -1215 & -1319 & -605 & 833 & -252 \\ 222 & -75 & -1249 & 662 & 37 & -537 & -3 & -609 \\ 39 & -414 & 569 & \mathbf{1649} & 274 & -45 & \mathbf{-2009} & -324 \end{array}\right).$$

The variance-of-squares of C^{vos} is found to be 5.45×10^{14} which is 2.5 times higher than before rotation. With a sum of squared residuals at $1.443820401 \times 10^{7}$ the fit of the two models is verified to remain unaffected by the orthonormal transformation. In contrast to C^{raw}, the rotated core, C^{vos}, directs the analyst to a few significant combinations of factors. This is clearly illustrated in Fig. 4 where the squared value of each of the 32 elements is plotted against the respective ranking (solid lines). The squares of the elements level out slightly below 2×10^{6} after the fifth element for the rotated core. Thus, the significant variation in data is accounted for by interpreting the factors represented by the five largest squared core elements. The sum of squares of these five squared elements is 4.62×10^{7}, whereas the sum of squares of the five largest elements of the un-rotated core amounts to 3.54×10^{7} as seen from the curves representing the cumulated values (dashed lines). The largest squared core element of the rotated core ($\approx 2 \times 10^{7}$) is approx. twice as high as the largest squared element of the unrotated core ($\approx 1 \times 10^{7}$), thus explaining twice the variation in the data. For comparison, a number of 9 core elements would have to be included in the interpretation of the un-rotated core to account for the same amount of variation.

As no body diagonal is defined for the (4,4,2) core array under investigation, the core cannot be optimized with respect to diagonality. For the sake of comparison and for proving the functionality of the algorithm, the core array has been optimized with respect to slice-wise diagonality over the last mode. The resulting core, C^{swdia}, is found to have a sum of squared slice diagonals of 3.80×10^{7} corresponding to

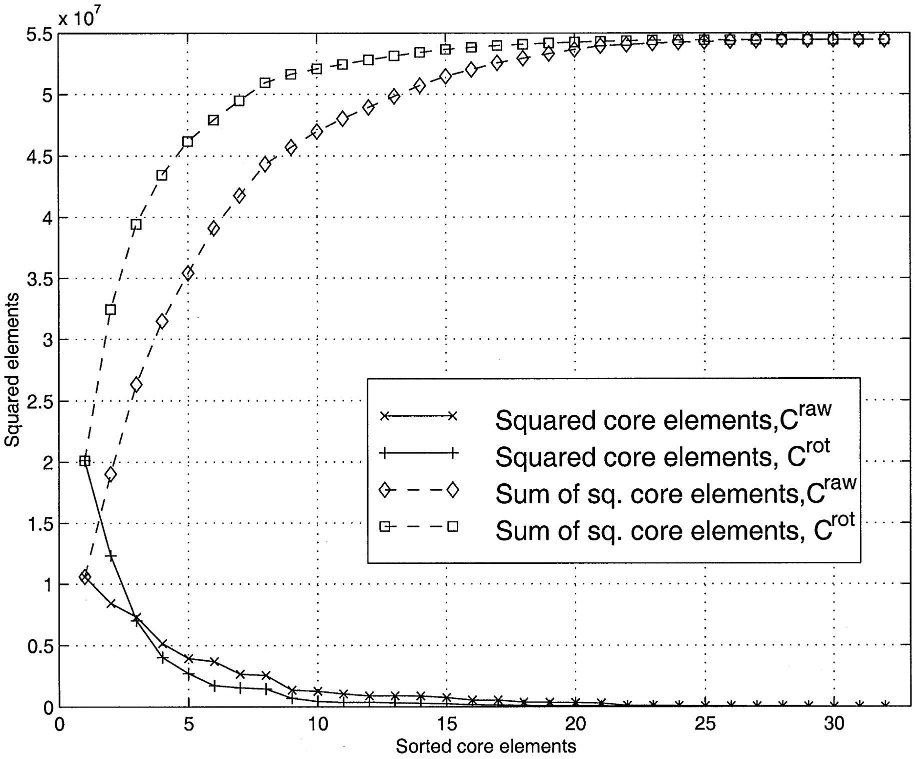

Fig. 4. Squared values of the 32 elements in the unrotated core and the rotated core are plotted against their ranking (solid lines). As expected, the rotated core has fewer and more significant core elements than does the unrotated core array. By following the course of the cummulated curves (dashed lines) it is concluded that for any given number of factor combinations the rotated core captures a significantly higher amount of variation of the data.

a degree of 69.8%:

$$C^{\mathrm{swdia}} = \left(\begin{array}{rrrr|rrrr} 4252 & -762 & -8 & 554 & 1362 & 2213 & -356 & -1751 \\ 59 & 3179 & -191 & -224 & -1618 & 132 & 1217 & 773 \\ -42 & -628 & 1619 & -561 & 321 & -303 & -1533 & -489 \\ 283 & -54 & 363 & 1365 & 697 & 622 & 671 & -1049 \end{array}\right).$$

According to C^{swdia} the core can be diagonalized to some extent, albeit, not yielding few significant elements, although the diagonalization has provided the analyst with a core that is a little simpler than the initial core array, but not as simple as the core array that is optimal in a variance-of-squares sense.

6. Convergence properties of the algorithm

In this section, we study convergence properties of Algorithm 1 presented in Section 3.1. First, we are going to show that the sequence of iterates generated by this algorithm has monotonically nondecreasing values of the criterion to be maximized. As a preparatory step, we need the following lemma:

Lemma 1. *With a matrix A of order (n,n) associate the optimization problem*

$$\max\{\operatorname{tr} PA \mid P \in \mathcal{O}(n)\}. \tag{16}$$

Then, the set of (global) solutions to (16) *is given by*

$$GS = \{P \in \mathcal{O}(n) \mid P = VU,\ (U,V) \in SV(A)\},$$

where

$$SV(A) = \{(U,V) \in \mathcal{O}(n) \times \mathcal{O}(n) | UAV = D,\ D \text{ is diagonal and nonnegative}\}$$

is the set of pairs of orthogonal matrices yielding an 'unordered' *singular-value decomposition of A.*

Proof. First note that $SV(A)$ consists of all pairs of orthogonal matrices providing a singular-value decomposition (in arbitrary order of singular values) of A. Now, writing down the first-order optimality conditions of (16), one verifies (similar to Section 3.1) the set of stationary solutions of this problem as being

$$SS = \{P \in \mathcal{O}(n) \mid PA \text{ is symmetric}\}.$$

Denote by $SEV(B)$ and $SSV(B)$ the sum of eigenvalues and singular-values, respectively, of a symmetric matrix B. Then,

$$\operatorname{tr} PA = SEV(PA) \leqslant SSV(PA) = SSV(A) = \operatorname{tr} QA \quad \forall (P,Q) \in SS \times GS \tag{17}$$

holds. Here, the first and second equality are obvious (recall the orthogonality of P), the inequality follows from the fact that the singular values of a symmetric matrix coincide with their absolute eigenvalues, and the last equality comes from the definition of GS:

$$\operatorname{tr} QA = \operatorname{tr} VUAVV^{\mathrm{T}} = \operatorname{tr} VDV^{\mathrm{T}} = \operatorname{tr} D = SSV(A),$$

due to the orthogonality of V and to D being a diagonal matrix of all singular values of A. Hence, the elements of GS realize a value of goal function which is not less than the value of the goal function of any element in SS, which in turn, being the set of stationary solutions to (16), contains the global solutions to (16). In conclusion, all elements of GS are global solutions. If, on the other hand, P is a global solution to (16), then $\operatorname{tr} PA \geqslant \operatorname{tr} QA$, where $Q \in GS$ is selected arbitrarily (a singular value decomposition of A always exists). As a global solution, P is a stationary solution as well, hence, $P \in SS$ and $SEV(PA) = SSV(PA)$ due to (17). Now, for the symmetric (due to $P \in SS$) matrix PA, there exists some $V \in \mathcal{O}(n)$ such that $V^{\mathrm{T}}PAV = D$, where D is a nonnegative (by $SEV(PA) = SSV(PA)$) diagonal matrix. Consequently, D contains the singular values of PA and, hence, those of A. Defining $U := V^{\mathrm{T}}P$, it follows that $P = VU$ and $(U,V) \in SV(A)$. This means $P \in GS$, hence the set of global solutions to (16) coincides with GS as was to be shown. □

Corollary 1. *The choice of P_i^k in step* 3 *of Algorithm* 1 *corresponds to a selection*

$$P_i^k \in \operatorname{argmax}\{\langle \nabla_{P_i} v(P_1^k, \ldots, P_{i-1}^k, P_i^{k-1}, \ldots, P_N^{k-1}), Q\rangle \,|Q \in \mathcal{O}(n_i)\}.$$

Proof. By definition of step 3 of Algorithm 1 and according to Lemma 1, one has

$$P_i^k \in \operatorname{argmax}\left\{\operatorname{tr} Q^{\mathrm{T}}\left[\frac{\partial v}{\partial P_i}(P_1^k,\dots,P_{i-1}^k,P_i^{k-1},\dots,P_N^{k-1})\right] |Q \in \mathcal{O}(n_i)\right\}.$$

Now the assertion follows from (9). □

Now, we are able to prove our first result on monotone convergence of functional values in Algorithm 1. To this aim, we refer to the criterion v as being partially convex, if it is convex in each variable P_i while the remaining ones are kept fixed. Of course, each convex v is partially convex, but the converse is not true. For instance, the function $f(x,y)=xy$ is partially convex (actually linear in both variables separately) but fails to be convex. We also recall that convexity of a differentiable function f implies the relation $f(y)-f(x)\geqslant\langle\nabla f(x),y-x\rangle$ for all x and y.

Lemma 2. *If the criterion v is partially convex, then the sequence $v(P_1^k,\dots,P_N^k)$ is nondecreasing with k.*

Proof. One has

$$\begin{aligned}
&v(P_1^k,\dots,P_N^k)-v(P_1^{k-1},\dots,P_N^{k-1})\\
&\quad=\sum_{i=1}^N v(P_1^k,\dots,P_{i-1}^k,P_i^k,P_{i+1}^{k-1},\dots,P_N^{k-1})\\
&\qquad-v(P_1^k,\dots,P_{i-1}^k,P_i^{k-1},P_{i+1}^{k-1},\dots,P_N^{k-1})\\
&\quad\geq\sum_{i=1}^N\langle\nabla_{P_i}v(P_1^k,\dots,P_{i-1}^k,P_i^{k-1},P_{i+1}^{k-1},\dots,P_N^{k-1}),P_i^k-P_i^{k-1}\rangle\geq 0.
\end{aligned}$$

Here, the first inequality relies on v being differentiable and partially convex, while the second inequality results from Corollary 1 due to $P_i^{k-1}\in\mathcal{O}(n_i)$. □

For the three criteria v_1,v_2,v_3 of core simplicity, introduced in Section 2, one has $v_i=\tilde{v}_i\circ T$, where T and the $\tilde{v}_i$ are defined by (11) and (12)–(14), respectively. Obviously, the $\tilde{v}_i$ are convex functions (for $\tilde{v}_1$, this follows from the invariance of the mean $\bar{C}$ in (6) under arbitrary orthogonal transformation T). On the other hand, the transformation T is multilinear, i.e. linear in each variable while the remaining ones are kept fixed. Consequently, the v_i are partially convex as compositions of a convex with a multilinear function. Furthermore, they are, of course, differentiable. Then, Lemma 2 allows to formulate the following result:

Corollary 2. *For the three criteria of core simplicity defined in Section 2, Algorithm 1 generates a sequence of iterates with monotonically nondecreasing values.*

Now, we turn to the convergence of iterates themselves. For the purpose of abbreviation, we put bold face characters for N-tuples of matrices, i.e., $\boldsymbol{P}=(P_1,\dots,P_N)$. As a first immediate result, we have:

Lemma 3. *If the sequence $\boldsymbol{P}^k$ of iterates generated by Algorithm* 1 *converges towards some $\boldsymbol{P}^*$, and if the criterion v is continuously differentiable, then $\boldsymbol{P}^*$ is a stationary solution of Problem* (P) *introduced in Section* 3.1.

Proof. Let $i \le N$ be arbitrarily given. By the remarks following the definition of Algorithm 1, one has that

$$P_i^{kT}\left[\frac{\partial v}{\partial P_i}(P_1^k,\dots,P_{i-1}^k,P_i^{k-1},\dots,P_N^{k-1})\right]$$

is a symmetric matrix. Passing to the limit $k \to \infty$, the above expression converges by the assumed continuous differentiability of v towards

$$P_i^{*\mathrm{T}}\left[\frac{\partial v}{\partial P_i}(\boldsymbol{P}^*)\right],$$

which, as a limit of symmetric matrices, is symmetric itself and, according to (10) implies $\boldsymbol{P}^*$ to be a stationary solution of Problem (P). □

Hence, if the iterates converge, then their limit is a stationary point, as desired. However, there is no guarantee for the sequence $\boldsymbol{P}^k$ to converge at all. On the other hand, since the $\boldsymbol{P}^k$ belong to the compact set $S{:=}\mathcal{O}(n_1)\times\cdots\times\mathcal{O}(n_N)$, there must exist some convergent subsequence $\boldsymbol{P}^{k_l}\to_l \boldsymbol{P}^*\in S$. Unfortunately, Lemma 3 does not apply to this subsequence and one may not derive the usual convergence result, stating that all accumulation points of the sequence of iterates are stationary solutions. This will be possible after excluding some degeneracy: we shall call $\boldsymbol{P}\in S$ a nondegenerate point of v, if the singular values of $(\partial v/\partial P_i)(\boldsymbol{P})$ are pairwise distinct and strictly positive for all $i \le N$. Then, we have:

Theorem 1. *Let v be continuously differentiable and partially convex* (*as it holds true for the three criteria of core simplicity defined in Section* 2). *Then each nondegenerate accumulation point of the sequence $\boldsymbol{P}^k$ generated by Algorithm* 1 *is a stationary solution of problem* (P) *introduced in Section* 3.1.

Proof. Denote by $\boldsymbol{P}^*\in S$ any nondegenerate accumulation point of $\boldsymbol{P}^k$. The realization of step 3 in Algorithm 1 means that $\boldsymbol{P}^{k+1}$ is defined by $P_i^{k+1}=(U_i^{k+1})^{\mathrm{T}}(V_i^{k+1})^{\mathrm{T}}$, where $U_i^{k+1},V_i^{k+1}\in\mathcal{O}(n_i)$ provide a singular-value decomposition

$$U_i^{k+1}\left[\frac{\partial v}{\partial P_i}(P_1^{k+1},\dots,P_{i-1}^{k+1},P_i^k,\dots,P_N^k)\right]V_i^{k+1}=\mathrm{diag}\,[d_{i,1}^{k+1},\dots,d_{i,n_i}^{k+1}],$$

with $d_{i,1}^{k+1}\ge\cdots\ge d_{i,n_i}^{k+1}\ge 0$ for $i=1,\dots,N$. Since v was assumed to be continuously differentiable, the derivative $\partial v/\partial P_i$ is bounded on the compact set S for all i, hence so are its singular values. As a consequence, there exists a subsequence with

$$\mathbf{P}^{k_l} \to_l \mathbf{P}^*; \quad (U_i^{k_l+1}, V_i^{k_l+1}, \mathrm{diag}[d_{i,1}^{k_l+1}, \ldots, d_{i,n_i}^{k_l+1}])$$
$$\to_l (U_i^{**}, V_i^{**}, \mathrm{diag}[d_{i,1}^{**}, \ldots, d_{i,n_i}^{**}]).$$

By definition of P_i^{k+1} and by continuity of $\partial v/\partial P_i$, it follows that $\mathbf{P}^{k_l+1} \to_l \mathbf{P}^{**}$, where $\mathbf{P}_i^{**} = (U_i^{**})^{\mathrm{T}}(V_i^{**})^{\mathrm{T}}$ and

$$U_i^{**} \frac{\partial v}{\partial P_i}(P_1^{**}, \ldots, P_{i-1}^{**}, P_i^*, \ldots, P_N^*) V_i^{**} = \mathrm{diag}[d_{i,1}^{**}, \ldots, d_{i,n_i}^{**}] \tag{18}$$

with $d_{i,1}^{**} \geq \cdots \geq d_{i,n_i}^{**} \geq 0$ for $i = 1, \ldots, N$. Furthermore, Lemma 2 along with the fact that $k_{l+1} \geq k_l + 1$ provide $v(\mathbf{P}^{k_{l+1}}) \geq v(\mathbf{P}^{k_l+1}) \geq v(\mathbf{P}^{k_l})$ and $v(\mathbf{P}^*) \geq v(\mathbf{P}^{**}) \geq v(\mathbf{P}^*)$, after passing to the limit $l \to \infty$. It results that $v(\mathbf{P}^*) = v(\mathbf{P}^{**})$.

Next we define the index set I to consist of those $i \leq N$ such that $P_i^* = U_i^{\mathrm{T}} V_i^{\mathrm{T}}$, where $U_i, V_i \in \mathcal{O}(n_i)$ provide any 'unordered' singular value decomposition

$$U_i \frac{\partial v}{\partial P_i}(P_1^{**}, \ldots, P_{i-1}^{**}, P_i^*, \ldots, P_N^*) V_i = \mathrm{diag}[d_{i,1}, \ldots, d_{i,n_i}],$$

with the $d_{i,j} \geq 0$ in *arbitrary* order. Suppose that $\{1, \ldots, i'\} \subseteq I$ for some $i' \leq N$. Then, by definition of I, one gets $P_1^* = U_1^{\mathrm{T}} V_1^{\mathrm{T}}$, where $U_1, V_1 \in \mathcal{O}(n_1)$ provide an unordered singular value decomposition

$$U_1 \frac{\partial v}{\partial P_1}(P_1^*, \ldots, P_N^*) V_1 = \mathrm{diag}[d_{1,1}, \ldots, d_{1,n_1}],$$

which after using some permutation matrix $\Pi \in \mathcal{O}(n_1)$ turns into a conventional singular value decomposition

$$\Pi U_1 \frac{\partial v}{\partial P_1}(P_1^*, \ldots, P_N^*) V_1 \Pi^{\mathrm{T}} = \mathrm{diag}[d_{1,1}, \ldots, d_{1,n_1}],$$

with $d_{1,1} \geq \cdots \geq d_{1,n_i}$. From (18) it follows that $P_1^{**} = (U_1^{**})^{\mathrm{T}}(V_1^{**})^{\mathrm{T}}$, where

$$U_1^{**} \frac{\partial v}{\partial P_1}(P_1^*, \ldots, P_N^*) V_1^{**} = \mathrm{diag}[d_{1,1}^{**}, \ldots, d_{1,n_1}^{**}]$$

provides another singular-value decomposition of the same derivative matrix. Now, the assumption of nondegeneracy of the accumulation point $\mathbf{P}^*$ yields the uniqueness of the singular-value decomposition of $(\partial v/\partial P_1)(\mathbf{P}^*)$ (cf. Horn and Johnson, 1991, pp. 147–148). In particular, $U_1^{**} = \Pi U_1$ and $V_1^{**} = V_1 \Pi^{\mathrm{T}}$ and, hence, $P_1^{**} = U_1^{\mathrm{T}} \Pi^{\mathrm{T}} \Pi V_1^{\mathrm{T}} = U_1^{\mathrm{T}} V_1^{\mathrm{T}} = P_1^*$. In case that $i' \geq 2$, we proceed with the index 2 as before with the index 1 in order to see that $P_2^* = U_2^{\mathrm{T}} V_2^{\mathrm{T}}$ with some $U_2, V_2 \in \mathcal{O}(n_2)$ which provide a singular-value decomposition

$$\begin{aligned} \Pi U_2 \frac{\partial v}{\partial P_2}(P_1^{**}, P_2^*, \ldots, P_N^*) V_2 \Pi^{\mathrm{T}} &= \mathrm{diag}[d_{2,1}, \ldots, d_{2,n_2}] \\ &= \Pi U_2 \frac{\partial v}{\partial P_2}(P_1^*, \ldots, P_N^*) V_2 \Pi^{\mathrm{T}}, \end{aligned}$$

where again Π is some permutation matrix, and the last equation comes from the first one by using the identity $P_1^{**} = P_1^*$ proved before. Noting that, by (18),

$$U_2^{**} \frac{\partial v}{\partial P_2}(P_1^{**}, P_2^*, \ldots, P_N^*) V_2^{**} = \mathrm{diag}[d_{2,1}^{**}, \ldots, d_{2,n_2}^{**}] = U_2^{**} \frac{\partial v}{\partial P_2}(P_1^*, \ldots, P_N^*) V_2^{**}$$

yields another singular value decomposition of the same derivative matrix on the right-hand side, the nondegeneracy of $\mathbf{P}^*$ implies $P_2^{**}=P_2^*$ with the same argumentation as given before for the index 1. Proceeding like that for all indices $i \le i'$, thereby consecutively exploiting the previously obtained relations $P_j^*=P_j^{**}$ for $j<i$, one ends up at the following statement:

$$\{1,\dots,i'\}\subseteq I \Rightarrow P_i^{**}=P_i^* \quad \forall i \le i'. \tag{19}$$

Now suppose that $I \ne \{1,\dots,N\}$. Denote by $i^* \le N$ the smallest index such that $i^* \notin I$. By definition of I, one has that $P_{i^*}^* \ne U^{\mathrm{T}}V^{\mathrm{T}}$ where U, V are arbitrary orthogonal matrices providing an 'unordered' singular-value decomposition of

$$\frac{\partial v}{\partial P_{i^*}}(P_1^{**},\dots,P_{i^*-1}^{**},P_{i^*}^*,\dots,P_N^*).$$

Then, Lemma 1 and (9) give (similar to the proof of Corollary 1)

$$P_{i^*}^* \notin \operatorname{argmax}\{\langle \nabla_{P_{i^*}} v(P_1^{**},\dots,P_{i^*-1}^{**},P_{i^*}^*,\dots,P_N^*),Q\rangle \mid Q \in \mathcal{O}(n_{i^*})\}. \tag{20}$$

On the other hand, a combination of (18), Lemma 1 and (9), implies for all $i \le N$,

$$P_i^{**} \in \operatorname{argmax}\{\langle \nabla_{P_i} v(P_1^{**},\dots,P_{i-1}^{**},P_i^*,\dots,P_N^*),Q\rangle \mid Q \in \mathcal{O}(n_i)\}. \tag{21}$$

Now, (20) together with (21) applied to the index i^* leads to

$$\langle \nabla_{P_{i^*}} v(P_1^{**},\dots,P_{i^*-1}^{**},P_{i^*}^*,\dots,P_N^*),P_{i^*}^{**}-P_{i^*}^*\rangle > 0 \tag{22}$$

and

$$\begin{aligned} & v(\mathbf{P}^{**})-v(\mathbf{P}^*) \\ &\quad =\sum_{i=1}^{N} v(P_1^{**},\dots,P_{i-1}^{**},P_i^{**},P_{i+1}^*,\dots,P_N^*)-v(P_1^{**},\dots,P_{i-1}^{**},P_i^*,P_{i+1}^*,\dots,P_N^*) \\ &\quad \ge \sum_{i=1}^{N} \langle \nabla_{P_i} v(P_1^{**},\dots,P_{i-1}^{**},P_i^*,P_{i+1}^*,\dots,P_N^*),P_i^{**}-P_i^*\rangle > 0, \end{aligned}$$

where the first inequality relies on v being differentiable and partially convex as in the proof of Lemma 2. All terms in the last sum are nonnegative in view of (21), but at least the term with index i^* is strictly positive according to (22), whence the strict inequality. The last derivation, however, is in contradiction to the fact that $v(\mathbf{P}^{**})=v(\mathbf{P}^*)$ which was proved above. Consequently, the assumption (following (19)) was false and it holds that $I=\{1,\dots,N\}$. As a result, for all $i \le N$ the P_i^* may be written as products $U_i^{\mathrm{T}}V_i^{\mathrm{T}}$ where $U_i, V_i \in \mathcal{O}(n_i)$ and

$$U_i\frac{\partial v}{\partial P_i}(P_1^{**},\dots,P_{i-1}^{**},P_i^*,\dots,P_N^*)V_i = D = U_i\frac{\partial v}{\partial P_i}(\mathbf{P}^*)V_i$$

with some diagonal matrix D, where the second equality relies on (19). Then,

$$P_i^{*\mathrm{T}}\frac{\partial v}{\partial P_i}(\mathbf{P}^*) = V_iDV_i^{\mathrm{T}}$$

are symmetric matrices for all $i \leq N$ as required in the stationarity condition (10). Hence, we have shown, that $\mathbf{P}^*$ is a stationary solution of problem (P). □

We note that the proof of Theorem 1 follows the typical patterns of convergence proofs for algorithms in nonlinear optimization as developed, for instance, in Zangwill (1969). The nondegeneracy condition in Theorem 1 may be supposed to be satisfied in 'almost all' problems since it expresses the typical situation of all singular values of some matrix being distinct and strictly positive. Indeed, in all examples we considered so far, the algorithm asymptotically reached a stationary solution (characterized by (10)).

Acknowledgements

The authors wish to thank two anonymous referees for their constructive suggestions which led to an improved presentation of this paper. The first author is grateful to EU Affluence and FØTEK funds for financial support. The support provided by Prof. Lars Munck is greatly appreciated. Finally we thank Danisco Sugar for providing the week-samples of thick juice.

References

Andersson, C.A., Munck, L., Henrion, R., Henrion, G., 1997. Analysis of N-dimensional data arrays from fluorescence spectroscopy of an intermediary sugar product. Fresenius J. Anal. Chem. 359, 138–142.

Carroll, J.D., Chang, J., 1970. Analysis of individual differences in multidimensional scaling via an N-way generalization of the "Eckardt-Young" decomposition. Psychometrika 35, 283–319.

Carroll, J.D., Wish, M., 1974. Models and methods for three-way multidimensional scaling. In: Krantz, D.H. (Ed.), Contemporary Developments in Mathematical Psychology, vol. 2, Freeman, San Francisco, pp. 57–105.

De Leeuw, J., Pruzansky, S., 1978. A new computational method to fit the weighted Euclidean distance model. Psychometrika 43, 479–490.

De Ligny, C., Spanjer, M., van Houwelingen, J., Weesie, H., 1984. J. Chromatography 301, 311–324.

Harshman, R.A., 1970. Foundations of the PARAFAC procedure: model and conditions for an 'explanatory' multi mode factor analysis. UCLA Working Papers in Phonetics, vol. 16, pp. 1–84.

Henrion, R., Andersson, C.A., 1999. A new criterion for simple-structure transformations of core arrays in N-way PCA. Chemom. Intell. Lab. Syst. 47, 189–204.

Henrion, R., Henrion, G., Böhme, M., Behrendt, H., 1997. Three-way principal components analysis for fluorescence spectroscopic classification of algea species. Fresenius J. Anal. Chem. 352, 431–436.

Horn, R.A., Johnson, C.R., 1991. Topics in Matrix Analysis. Cambridge University Press, Cambridge.

Kapteyn, A., Neudecker, H., Wansbeek, T., 1986. An approach to n-mode components analysis. Psychometrika 51, 269–275.

Kiers, H.A.L., 1992. TUCKALS3 core rotations and constrained TUCKALS modelling. Statist. Appl. 4, 659–667.

Kiers, H.A.L., 1999. Three-way simplimax for oblique rotation of the three-mode factor analysis core to simple structure. Comput. Statist. Appl. Data Anal. 28, 307–324.

Kroonenberg, P.M., 1983. Three Mode Principal Component Analysis: Theory and Applications. DSWO Press, Leiden.

Larsson, H., 1989. Svenskt Socker, Sockerbolaget, Vejbystelle, Sweden (in Swedish).

Magnus, J.R., Neudecker, H., 1988. Matrix Differential Calculus with Applications in Statistics and Econometrics. Wiley, Chichester.

Nørgaard, L., 1995. Classification and prediction of quality and process parameters of thick juice and beet sugar by fluorescence spectroscopy and chemometrics. Zuckerindustrie 120, 970–981.

Smilde, A.K., 1992. Three-way analyses. Problems and prospects. Chemom. Intell. Lab. Syst. 15, 143–157.

Tucker, L.R., 1966. Some mathematical notes on three-mode factor analysis. Psychometrika 31, 279–311.

Zangwill, W.I., 1969. Nonlinear Programming. Prentice-Hall, Englewood Cliffs, NJ.

Zeng, Y., Hopke, P.K., 1990. Methodological study applying three-mode factor analysis to three-way chemical data sets. Chemom. Intell. Lab. Syst. 7, 237–250.