JOURNAL OF CHEMOMETRICS
J. Chemometrics **13,** 275–294 (1999)

PARAFAC2—PART I. A DIRECT FITTING ALGORITHM FOR THE PARAFAC2 MODEL

HENK A. L. KIERS,[1]* JOS M. F. TEN BERGE[1] AND RASMUS BRO[2]
[1]*Heymans Institute (PA), University of Groningen, Groningen, The Netherlands*
[2]*Chemometrics Group, Food Technology, Department of Dairy and Food Science, The Royal Veterinary and Agricultural University, Frederiksberg, Denmark*

SUMMARY

PARAFAC is a generalization of principal component analysis (PCA) to the situation where a set of data matrices is to be analysed. If each data matrix has the same row and column units, the resulting data are three-way data and can be modelled by the PARAFAC1 model. If each data matrix has the same column units but different (numbers of) row units, the PARAFAC2 model can be used. Like the PARAFAC1 model, the PARAFAC2 model gives unique solutions under certain mild assumptions, whereas it is less severely constrained than PARAFAC1. It may therefore also be used for regular three-way data in situations where the PARAFAC1 model is too restricted. Usually the PARAFAC2 model is fitted to a set of matrices with cross-products between the column units. However, this model-fitting procedure is computationally complex and inefficient. In the present paper a procedure for fitting the PARAFAC2 model directly to the set of data matrices is proposed. It is shown that this algorithm is more efficient than the indirect fitting algorithm. Moreover, it is more easily adjusted so as to allow for constraints on the parameter matrices, to handle missing data, as well as to handle generalizations to sets of three- and higher-way data. Furthermore, with the direct fitting approach we also gain information on the row units, in the form of ‘factor scores’. As will be shown, this elaboration of the model in no way limits the feasibility of the method. Even though full information on the row units becomes available, the algorithm is based on the usually much smaller cross-product matrices only. Copyright © 1999 John Wiley & Sons, Ltd.

KEY WORDS: principal component analysis; multiple-set data; repeated measurements; three-way data; PARAFAC

1. INTRODUCTION

Principal component analysis (PCA) is a popular technique for the exploratory analysis of a set of variables. If **X** denotes an $I \times J$ data matrix, then PCA comes down to minimizing

$$\sigma(\mathbf{F},\mathbf{A}) = \|\mathbf{X} - \mathbf{F}\mathbf{A}^{\mathrm{T}}\|^2 \tag{1}$$

over the $I \times R$ factor score matrix **F** and the $J \times R$ loading matrix **A;** in PCA, factor scores are usually denoted as component scores, but for consistency with the other terminology in the present paper we denote them as ‘factor scores’ here. The solution for **F** and **A** is not unique, because replacing **F** by **FT** and **A** by $\mathbf{A}(\mathbf{T}^{\mathrm{T}})^{-1}$, where **T** is an arbitrary non-singular matrix, gives the same function value. Faced

* Correspondence to: H. A. L. Kiers, Heymans Institute (PA), University of Groningen, Groningen, The Netherlands. E-mail: h.a.l.kiers@ppsw.rug.nl

Contract/grant sponsor: Nordic Industry Foundation; Contract/grant number: P93149.
Contract/grant sponsor: FØTEK.

CCC 0886–9383/99/030275–20 $17.50

Received 25 July 1998
Accepted 4 February 1999

with the problem of choosing a particular orientation of the axes in the PCA solution, Cattell[1] proposed the principle of parallel proportional profiles as a basis for determining such orientation. According to this principle, different observations of the same variables on the same observation units are determined by a single set of factors, of which only the relative importance may differ over occasions. This principle led to the PARAFAC[2] model, which has been proposed independently by Carroll and Chang[3] under the name CANDECOMP. The PARAFAC model, here named PARAFAC1 to distinguish it from the PARAFAC2 method to be discussed later, is based on the parallel proportional profile principle as follows. Let the $I \times J$ matrix $\mathbf{X}_k$, $k = 1,\ldots,K$, denote the kth slab of a three-way data array $\underline{\mathbf{X}}$, for instance, consisting of scores of I observation units on J variables at K occasions; in a particular chemical context, e.g. chromatography, one may have K samples separated in a chromatographic system using spectral detection with J wavelengths I times during elution. Then, according to the PARAFAC1 model, we have

$$\mathbf{X}_k = \mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}} + \mathbf{R}_k \tag{2}$$

where $\mathbf{F}$ is an $I \times R$ matrix of factor scores (for the row units), $\mathbf{A}$ is a $J \times R$ matrix of weights for the column units (analogous to loadings in PCA), $\mathbf{D}_k$ is a diagonal ($R \times R$) matrix containing the weights for the kth slab of $\underline{\mathbf{X}}$, and $\mathbf{R}_k$ denotes an $I \times J$ matrix with residuals, $k = 1,\ldots,K$. If we define $\mathbf{A}_k = \mathbf{A}\mathbf{D}_k$, this model can be seen as a variant of the PCA model, with the same factor scores matrix for all slabs, and loading matrices that are columnwise proportional across slabs. The PARAFAC1 model is fitted to the data in the least squares sense, which amounts to minimizing

$$\sigma_1(\mathbf{F}, \mathbf{A}, \mathbf{D}_1, \ldots, \mathbf{D}_k) = \sum_{k=1}^{K} \quad \|\mathbf{X}_k - \mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2 \tag{3}$$

where summation is over $k = 1,\ldots,K$. In addition to the original algorithms[2,3] for this, various improvements have been proposed.[4,5]

Harshman[6] and Kruskal[7] have shown that, under certain relatively mild conditions, there is a unique solution for **F, A** and $\mathbf{D}_1,\ldots,\mathbf{D}_K$ that minimizes (3), up to scaling and permutation of the columns of the matrices. Hence the above-described PARAFAC1 method offers a solution to the problem of rotational indeterminacy of factor analytic solutions in cases where a number of observation units are measured on the same variables a number of times. However, sometimes the model is too restrictive in that it employs the same factor score matrix **F** for all data matrices, which need not always be a plausible assumption. An obvious case in point is where the observation units vary from data set to data set, but even if basically the same observation units are used, it is possible that the assumption of equality of the factor scores is too strong. In such cases where all data matrices pertain to the same set of variables, but where observation units are not comparable (and may even differ in number), one may use the parallel proportional profile principle in an adjusted way, as follows. For each occasion one can model the observations on the $n_k \times J$ matrix $\mathbf{X}_k$ as

$$\mathbf{X}_k = \mathbf{F}_k\mathbf{D}_k\mathbf{A}^{\mathrm{T}} + \mathbf{R}_k \tag{4}$$

which differs from the PARAFAC1 model only in that for each data matrix we have a different factor score matrix, which is a natural implication of having incomparable observation units in each data matrix. As is readily verified, the present model does not have the uniqueness properties of the PARAFAC1 model. This is because the factor scores can no longer be constrained to be equal for the different data matrices. The only invariance constraint imposed in (4) is that the loading matrices

$\mathbf{AD}_1,\ldots,\mathbf{AD}_K$ be proportional, thus modelling the relations of the variables to the factors (as expressed by the loadings) in essentially the same way for every data matrix. Indeed, this is not sufficient to get uniqueness, as follows from the fact that $\mathbf{F}_k\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{F}_k\mathbf{D}_k\mathrm{T}^{-1}\mathbf{E}_k^{-1}\mathbf{E}_k\mathbf{T}\mathbf{A}^{\mathrm{T}} = \mathbf{G}_k\mathbf{E}_k\mathbf{B}^{\mathrm{T}}$, with $\mathbf{G}_k \equiv \mathbf{F}_k\mathbf{D}_k\mathbf{T}^{-1}\mathbf{E}_k^{-1}$, $\mathbf{E}_k$ diagonal and $\mathbf{B} \equiv \mathbf{AT}^{\mathrm{T}}$, for any non-singular matrix $\mathbf{T}$. Hoping to find a model which does have a unique representation, Harshman[8]. imposed a particular invariance constraint on the factor scores, as follows. He has proposed to impose that the cross-product matrix $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k$ is constant over k. The model in equation (4) combined with this constraint is called the PARAFAC2 model. When $\mathbf{F}_K$ contains centred scores, this implies that the correlations between the factors are kept invariant, as can be seen upon noting that scale differences are captured by the $\mathbf{D}_k$ matrices. More generally, this constraint implies that congruence coefficients,[9] also called[10] 'uncorrected correlation coefficients', between columns of $\mathbf{F}_k$ are invariant over k. The congruence coefficient is defined as

$$\varphi(\mathbf{x},\mathbf{y}) = \frac{\mathbf{x}^{\mathrm{T}}\mathbf{y}}{\sqrt{\mathbf{x}^{\mathrm{T}}\mathbf{x}}\sqrt{\mathbf{y}^{\mathrm{T}}\mathbf{y}}} \tag{5}$$

where $\mathbf{x}$ and $\mathbf{y}$ denote vectors of the same order.

An example of a situation with incomparable cases, where the PARAFAC2 model can be useful, is the following. In typical chromatography with spectral detection the data can be arranged in a three-way array, with the first mode designating elution time, the second mode wavelength and the third mode samples. An ideal model for such data would be the PARAFAC1 model, with the first mode (here seen as factor scores) corresponding to elution profiles, the second mode to pure spectra and the third mode to concentration profiles. However, if the retention times of specific analytes shift from sample to sample, the model no longer holds, because in fact the first-mode units become incomparable. Assume, however, that the following holds for the elution profiles of each sample, i.e. for each column in the elution profile matrix. All profiles are approximately shifted the same amount, and for every profile, sufficiently many elution times are sampled such that at both extremes of the profile the same (baseline) values are found. Then it holds that, even though the elution profiles shift from sample to sample, the cross-products of the factor score matrix holding the elution profiles remain constant (see Part II[11]).

Instead of fitting model (4) itself to the data subject to the constraint that $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k$ be invariant across occasions, Harshman[8] proposed to fit the cross-product version of model (4) to the observed cross-products. Specifically, if $\mathbf{C}_k$ denotes the cross-product matrix associated with $\mathbf{X}_k$, then he proposed to fit (in the least squares sense) the model

$$\mathbf{C}_k = \mathbf{AD}_k\mathbf{\Phi}\mathbf{D}_k\mathbf{A}^{\mathrm{T}} + \mathbf{R}_k \tag{6}$$

where $\mathbf{\Phi}$ denotes the invariant matrix $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k$, and $\mathbf{R}_k$ now denotes a matrix with residuals for the cross-product matrix $\mathbf{C}_k$, $k = 1,\ldots,K$ (and hence is entirely different from that in (4)).

In the present paper it is proposed to fit the model (4) itself to the data, rather than fitting the derived model (6) to the cross-products. Specifically, it is proposed here to minimize

$$\sigma_2(\mathbf{F}_k, \mathbf{A}, \mathbf{D}_1, \ldots, \mathbf{D}_K) = \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{F}_k\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2 \tag{7}$$

subject to the constraint that $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{F}_l^{\mathrm{T}}\mathbf{F}_l$ for all pairs k, $l = 1,\ldots,K$, and that $\mathbf{D}_k$, $k = 1,\ldots,K$, be diagonal. The first reason for direct fitting of the PARAFAC2 model (by minimizing (7)) instead of indirect fitting (by minimizing the sum of squared residuals in (6)) is that in this way the actual data, rather than derived entities, are fitted and, as a consequence, factor scores (for the row units) will be obtained, whereas these are not obtained by the indirect fitting procedure. An additional reason is that the present fitting procedure is easier to adjust when constraints (e.g. non-negativity constraints) are to be imposed on the parameters, as well as when missing data are to be handled by replacing them by the optimal model estimates (as in expectation maximization (EM) procedures). Also, the present procedure is easier to generalize to situations with sets of three-way or higher-way data, as will be discussed at the end of the paper. Finally, the algorithm for direct fitting turns out to be considerably more efficient.

In the present paper it will first be derived how (7) can be minimized subject to the constraints at hand. Next it will be shown that the present method, just as the original PARAFAC2 method, is based on the cross-products $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$, $k=1,\ldots,K$, only. Then a simple lower bound to the function σ_2 will be given. Finally the procedure will be tested and compared with the algorithm for the original PARAFAC2 method. For an application of the newly developed method we refer to Part II.[11]

2. AN ALGORITHM FOR DIRECT FITTING OF THE PARAFAC2 MODEL

Before dealing with the minimization of (7), we will first reformulate the constraint that $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{F}_j^{\mathrm{T}}\mathbf{F}_j$ for all pairs $j,k=1,\ldots,K$. For this constraint to be met, it is necessary and sufficient to have $\mathbf{F}_k=\mathbf{P}_k\mathbf{F}$ for a columnwise orthonormal ($n_k \times R$) matrix $\mathbf{P}_k$ and an $R \times R$ matrix $\mathbf{F}$, $k=1,\ldots,K$. Sufficiency follows from the fact that $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{F}^{\mathrm{T}}\mathbf{P}_k^{\mathrm{T}}\mathbf{P}_k\mathbf{F} = \mathbf{F}^{\mathrm{T}}\mathbf{F}$ is constant over k. To prove necessity, we start from the fact that $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{F}_j^{\mathrm{T}}\mathbf{F}_j$ for all pairs $j,k=1,\ldots,K$, hence that, taking $j=1$ as a point of reference, $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{F}_1^{\mathrm{T}}\mathbf{F}_1$ for $k=1,\ldots,K$. We express $\mathbf{F}_k$ with respect to a columnwise orthonormal basis matrix $\mathbf{Q}_k$ ($n_k \times R$) as $\mathbf{F}_k=\mathbf{Q}_k\mathbf{T}_k$, $k=1,\ldots,K$, where $\mathbf{T}_k$ is an $R \times R$ matrix. Then it follows that $\mathbf{T}_k^{\mathrm{T}}\mathbf{T}_k = \mathbf{T}_1^{\mathrm{T}}\mathbf{T}_1$ and hence $\mathbf{T}_k=\mathbf{N}_k\mathbf{T}_1$, where $\mathbf{N}_k$ is an orthonormal $R \times R$ matrix, for $k=1,\ldots,K$. As a consequence, $\mathbf{F}_k=\mathbf{Q}_k\mathbf{N}_k\mathbf{T}_1$, which, upon defining $\mathbf{P}_k \equiv \mathbf{Q}_k\mathbf{N}_k$ and $\mathbf{F} \equiv \mathbf{T}_1$, gives $\mathbf{F}_k=\mathbf{P}_k\mathbf{F}$ for a columnwise orthonormal ($n_k \times R$) matrix $\mathbf{P}_k$ and an $R \times R$ matrix $\mathbf{F}$, $k=1,\ldots,K$.

Having proven that the constraint $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{F}_j^{\mathrm{T}}\mathbf{F}_j$ for all $j,k=1,\ldots,K$ is equivalent to the constraint $\mathbf{F}_k=\mathbf{P}_k\mathbf{F}$ for some columnwise orthonormal ($n_k \times R$) matrix $\mathbf{P}_k$ and an $R \times R$ matrix $\mathbf{F}$, $k=1,\ldots,K$, we substitute $\mathbf{F}_k=\mathbf{P}_k\mathbf{F}$ for $\mathbf{F}_k$ in (7) to find

$$\sigma_3(\mathbf{P}_1,\ldots,\mathbf{P}_K,\mathbf{F},\mathbf{A},\mathbf{D}_1,\ldots,\mathbf{D}_K) = \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{P}_k\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2 \tag{8}$$

which is to be minimized over all its arguments, subject to the constraints that $\mathbf{P}_k^{\mathrm{T}}\mathbf{P}_k = \mathbf{I}_R$ and $\mathbf{D}_k$ is diagonal, $k=1,\ldots,K$. To minimize this function, we propose an alternating least squares algorithm that alternately minimizes (8) over $\mathbf{P}_k$ for fixed $\mathbf{F}$, $\mathbf{D}_k$ and $\mathbf{A}$, $k=1,\ldots,K$, and over $\mathbf{F}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$ and $\mathbf{A}$ for fixed $\mathbf{P}_1,\ldots,\mathbf{P}_K$.

Minimizing (8) over $\mathbf{P}_k$ subject to $\mathbf{P}_k^{\mathrm{T}}\mathbf{P}_k = \mathbf{I}_R$ is equivalent to maximizing

$$f(\mathbf{P}_k) = \mathrm{tr}\ \mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}\mathbf{P}_k \tag{9}$$

$k=1,\ldots,K$, the solution for which is found as follows.[12] Let $\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}} = \mathbf{U}_k\boldsymbol{\Delta}_k\mathbf{V}_k^{\mathrm{T}}$ be a singular value decomposition (SVD); then the maximum of (9) over columnwise orthonormal $\mathbf{P}_k$ is given by

$$\mathbf{P}_k = \mathbf{V}_k\mathbf{U}_k^{\mathrm{T}} \tag{10}$$

$k=1,\ldots,K$.

The problem of minimizing (8) over $\mathbf{F}$, $\mathbf{D}_1,\ldots,\mathbf{D}_k$ and $\mathbf{A}$ reduces to minimizing

$$\sigma_3(\mathbf{F},\mathbf{A},\mathbf{D}_1,\ldots,\mathbf{D}_K|\mathbf{P}_1,\ldots,\mathbf{P}_K)=\sum_{k=1}^{K}\|\mathbf{P}_k^{\mathrm{T}}\mathbf{X}_k-\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2+c \tag{11}$$

where c denotes a constant with respect to $\mathbf{F}$, $\mathbf{A}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$. Clearly, minimizing (11) is equivalent to the PARAFAC1 problem of minimizing (3) with $\mathbf{X}_k$ replaced by $\mathbf{P}_k^{\mathrm{T}}\mathbf{X}_k$. Hence for the minimization of (8) over $\mathbf{F}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$ and $\mathbf{A}$ we can use any PARAFAC1 algorithm. Rather than actually minimizing (8) over $\mathbf{F}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$ and $\mathbf{A}$, it suffices to *decrease* (8) over these parameter matrices, as is achieved by using one cycle of updates from a PARAFAC1 algorithm.

A complete alternating least squares algorithm can now be set up by alternately minimizing σ_3 over $\mathbf{P}_k$ according to (10) and applying a cycle of PARAFAC1 updates to the $R\times J\times K$ three-way array with frontal planes $\mathbf{P}_k^{\mathrm{T}}\mathbf{X}_k$, $k=1,\ldots,K$. Because both steps decrease the function value of σ_3 and because σ_3 is bounded below by zero, this procedure will converge to a stable function value.

The algorithm needs to be started at certain values for three of the four parameter sets. These can be chosen randomly, but alternatively one may choose an initialization which has particular useful properties and which it is hoped will have a relatively high chance of leading to the global (rather than a local) minimum of the function. As such a ‘rational’ start, for $\mathbf{A}$ we propose to take the loadings obtained from a PCA of the matrix containing $\mathbf{X}_1,\ldots,\mathbf{X}_K$ stacked below each other. The matrices $\mathbf{F}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$ are initialized as identity matrices. As will be demonstrated in the simulation study reported below, this start indeed has a relatively high chance of leading to the global minimum. Furthermore, this start has the advantage that in the special case where $K=1$ (in which case our method reduces to fitting the ordinary PCA model) the start itself gives the global minimum of the loss function, and hence no iterations are required whatsoever. Nevertheless, in practice it is recommended to use several randomly started runs in addition to a run started rationally, in order to decrease the chance of missing the global minimum of the function.

A schematic description of the algorithm is postponed to the end of Section 3. In Section 3, first a procedure for enhancing computational efficiency is proposed.

3. SUFFICIENCY OF CROSS-PRODUCTS

If n_k (the number of rows in the kth data matrix) is considerably larger than J, it will be inefficient to work with the full data matrix $\mathbf{X}_k$. In such cases the original PARAFAC2 method, working with cross-product matrices $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$, might seem to be more efficient. However, as will be shown below, our direct PARAFAC2 method similarly uses only the cross-product matrices $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$ during the iterations. Hence our direct PARAFAC2 method can be made considerably more efficient in such cases by replacing $\mathbf{X}_k$ by a smaller matrix with the same cross-product matrix, as will be explained below.

The direct PARAFAC2 method consists of minimizing σ_3 over $\mathbf{P}_k$ for fixed $\mathbf{F}$, $\mathbf{D}_k$ and $\mathbf{A}$, $k=1,\ldots,K$, subject to certain constraints. However, we can eliminate the largest matrix $\mathbf{P}_k$ from the minimization problem by expressing the solution for $\mathbf{P}_k$ in terms of the other parameters and $\mathbf{X}_k$. That is, from (10) we have $\mathbf{P}_k=\mathbf{V}_k\mathbf{U}_k^{\mathrm{T}}=\mathbf{V}_k\boldsymbol{\Delta}_k\mathbf{U}_k^{\mathrm{T}}(\mathbf{U}_k\boldsymbol{\Delta}_k^2\mathbf{U}_k^{\mathrm{T}})^{-1/2}$, which can be written as

$$\mathbf{P}_k=\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}}(\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}})^{-1/2} \tag{12}$$

Substituting (12) for $\mathbf{P}_k$ in (8) gives

$$\sigma_3(\mathbf{F},\mathbf{A},\mathbf{D}_1,\ldots,\mathbf{D}_K)=\sum_{k=1}^{K}\|\mathbf{X}_k-\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}}(\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}})^{-1/2}\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2$$

$$=\sum_{k=1}^{K}\|\mathbf{X}_k\|^2-2\sum_{k=1}^{K}\mathrm{tr}\ \mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}}(\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}})^{-1/2}\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}+\sum_{k=1}^{K}\|\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2$$

$$=\sum_{k=1}^{K}\mathrm{tr}\ \mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k-2\sum_{k=1}^{K}\mathrm{tr}\ (\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k\mathbf{A}\mathbf{D}_k\mathbf{F}^{\mathrm{T}})^{1/2}+\sum_{k=1}^{K}\|\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2 \tag{13}$$

Clearly, in this function, $\mathbf{X}_k$ only appears in the cross-products $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$. Therefore the minimum of σ_3 depends on $\mathbf{X}_k$ only through the cross-products $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$.

Above we have seen that the minimum of the loss function (8), as well as the set $[\mathbf{F},\mathbf{A},\mathbf{D}_1,\ldots,\mathbf{D}_K]$ that minimizes the loss function, depends on $\mathbf{X}_k$ only through $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$. It follows that the problem of minimizing (8) reduces to that of minimizing (13) even if we replace the $n_k \times J$ matrix $\mathbf{X}_k$ by any other ($m_k \times J$) matrix $\mathbf{H}_k$ (with m_k considerably smaller than n_k) which has the same cross-product matrix as $\mathbf{X}_k$. Such a matrix can, for instance, be obtained by the Cholesky decomposition of $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$, in which case $\mathbf{H}_k$ will have order $m_k = J$. As a consequence, function (8) with $\mathbf{X}_k$ replaced by a (much smaller) matrix $\mathbf{H}_k$ for which $\mathbf{H}_k^{\mathrm{T}}\mathbf{H}_k=\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$ has the same minimum, reached for the same parameter matrices $\mathbf{F}$, $\mathbf{A}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$, as has the original function (8). Thus we can efficiently find the minimum of σ_3 and the minimizing $\mathbf{F}$, $\mathbf{A}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$ by first computing matrices $\mathbf{H}_k$ for which $\mathbf{H}_k^{\mathrm{T}}\mathbf{H}_k=\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$ and then applying our algorithm to the set of these matrices $\mathbf{H}_k$, $k = 1,\ldots,K$. This procedure does not give the correct solutions for $\mathbf{P}_k$, k = 1,…,K. In order to obtain the solution for the matrix $\mathbf{P}_k$, which we expressed in terms of the other parameter matrices in (12), we do need the original data matrix $\mathbf{X}_k$, and compute $\mathbf{P}_k$ according to (12). Thus, only in order to compute factor scores, given by $\mathbf{F}_k = \mathbf{P}_k\mathbf{F}$, we need to have the full matrices $\mathbf{X}_k$, $k = 1,\ldots,K$. This situation is similar to that of PCA: One can compute loading matrices using the cross-product matrix only, without having access to the raw data; however, in order to compute factor scores, the data for each individual observation have to be used.

To summarize, both the original PARAFAC2 method and our direct PARAFAC2 method use only the cross-product matrices, albeit that they optimize different functions involving these cross-products. Furthermore, the indirect PARAFAC2 method has no method for obtaining factor scores, whereas factor scores are explicitly defined in the model (4) fitted in our direct PARAFAC2 method.

The complete direct fitting PARAFAC2 algorithm can be summarized as follows.

Step 0. If $J < n_k$, replace $\mathbf{X}_k$ by $\mathbf{H}_k$, e.g. from the Cholesky decomposition $\mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k=\mathbf{H}_k^{\mathrm{T}}\mathbf{H}_k$.

Step 1. Initialize $\mathbf{A}$ as the loading matrix from PCA on $\sum_k \mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$ and initialize $\mathbf{F}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$ as $\mathbf{I}_R$.

Step 1a. Compute the SVD $\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}=\mathbf{U}_k\boldsymbol{\Delta}_k\mathbf{V}_k^{\mathrm{T}}$ and update $\mathbf{P}_k$ as $\mathbf{V}_k\mathbf{U}_k^{\mathrm{T}}$, $k = 1,\ldots,K$.

Step 1b. Update $\mathbf{F}$, $\mathbf{A}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$ by one cycle of a PARAFAC1 algorithm applied to the $R \times J \times K$ three-way array with frontal planes $\mathbf{P}_k^{\mathrm{T}}\mathbf{X}_k$, $k = 1,\ldots,K$.

Step 1c. Evaluate the function value σ_3 $(\mathbf{P}_1,\ldots,\mathbf{P}_K,\mathbf{F},\mathbf{A},\mathbf{D}_1,\ldots,\mathbf{D}_K) = \sum_k \|\ \mathbf{X}_k - \mathbf{P}_k\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2$; see (8). If $\sigma_3^{\mathrm{old}}-\sigma_3^{\mathrm{new}}>\epsilon\sigma_3^{\mathrm{old}}$ for some small value ε, repeat Step 1; else go to Step 2.

Step 2. If $\mathbf{X}_k$ has been replaced by $\mathbf{H}_k$ in Step 0, now replace $\mathbf{H}_k$ by $\mathbf{X}_k$ again and compute $\mathbf{P}_k$ according to Step 1a, $k = 1,\ldots,K$.

4. UPPER BOUND TO THE PARAFAC2 FIT VALUE

The minimum of the direct PARAFAC2 loss function can easily be related to the minimum of a loss function based on a PCA. That is, we have

$$\min \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{F}_k\mathbf{A}^{\mathrm{T}}\|^2 \leq \min \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{P}_k\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2 \tag{14}$$

where ‘min’ means that the minimum of the function over all parameter matrices at hand is taken, $k = 1,\ldots,K$. This inequality follows at once from the fact that for $\mathbf{F}_k$ constrained to be equal to $\mathbf{P}_k\mathbf{F}\mathbf{D}_k$ the minimum of the loss function is never smaller than for unconstrained $\mathbf{F}_k$. The left-hand term gives the loss function value resulting from a PCA of the supermatrix $\mathbf{X} = [\mathbf{X}_1^{\mathrm{T}} \ldots \mathbf{X}_K^{\mathrm{T}}]^{\mathrm{T}}$. To verify this, it should be noted that

$$\sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{F}_k\mathbf{A}^{\mathrm{T}}\|^2 = \| [\mathbf{X}_1^{\mathrm{T}} \ \ldots \ \mathbf{X}_K^{\mathrm{T}}] - \mathbf{A}[\mathbf{F}_1^{\mathrm{T}} \ \ldots \ \mathbf{F}_K^{\mathrm{T}}] \|^2 \tag{15}$$

the minimum of which is obtained by a PCA of $\mathbf{X}$. This minimum is equal to the sum of the $J-R$ smallest eigenvalues of $\sum_k \mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k$. Upon defining the PARAFAC2 *fit* value (often reported as fit *percentage*, after multiplication by 100) as the proportion of the total sum of squares that is explained by the R-dimensional model (i.e. the total minus the residual sum of squares), namely

$$\mathrm{Fit}_{\mathrm{PF2}}(R) = 1 - \sum_{k=1}^{K} \|\mathbf{X}_k - \mathbf{P}_k\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}\|^2 / \sum_{k=1}^{K} \|\mathbf{X}_k\|^2 \tag{16}$$

we find that inequality (14) immediately leads to

$$\mathrm{Fit}_{\mathrm{PF2}}(R) \leq \sum_{r=1}^{R} \lambda_r \left(\sum_{k=1}^{K} \mathbf{X}_k^{\mathrm{T}}\mathbf{X}_k \right) / \sum_{k=1}^{K} \|\mathbf{X}_k\|^2 \tag{17}$$

where λ_r () denotes the rth eigenvalue of the matrix in parentheses; the right-hand side gives the proportion of variance explained by a PCA of the supermatrix $\mathbf{X}$.

This upper bound for the direct PARAFAC2 *fit* value is easily evaluated and gives an idea of the severeness of the constraints imposed in the direct PARAFAC2 method. That is, if the direct PARAFAC2 fit value is only slightly smaller than that for PCA of $\mathbf{X}$, then apparently the more parsimonious PARAFAC2 representation is almost just as good as the PCA representation of $\mathbf{X}$.

5. TESTING THE DIRECT PARAFAC2 ALGORITHM

5.1. Simulation study I. Comparison of direct fitting algorithm with indirect fitting algorithm

Kiers[13] has developed an algorithm for least squares fitting of the original indirect PARAFAC2 model (6), but found very slow convergence. Here we compare the presently proposed method with his method and investigate some further aspects of the performance of the new method by means of a simulation study. For this purpose we have constructed 80 data sets according to the model

$$\mathbf{X}_k = \mathbf{F}_k\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{P}_k\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}} \tag{18}$$

based on random matrices $\mathbf{A}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$ (all diagonal) and $\mathbf{P}_1,\ldots,\mathbf{P}_K$ (all columnwise orthonormal) and fixed matrices $\mathbf{F}$, thus fitting the direct PARAFAC2 model exactly. The elements of $\mathbf{A}$ were drawn from the standard normal distribution. The elements of $\mathbf{D}_1,\ldots,\mathbf{D}_K$ were drawn from the uniform [0, 1] distribution and hence are non-negative; to avoid computational problems that might be caused by multicollinearity of the columns of the matrix $\mathbf{C}$ containing the diagonal elements of $\mathbf{D}_1,\ldots,\mathbf{D}_K$ in its rows, we selected only cases where the congruences between columns of $\mathbf{C}$ were smaller than 0·8. The elements of $\mathbf{P}_k$ were obtained by first sampling from a standard normal distribution and next applying a standard orthonormalization routine to the obtained matrix, thus leading to columnwise orthonormal matrices $\mathbf{P}_1,\ldots,\mathbf{P}_K$. The matrix $\mathbf{F}$ was taken such that $\mathbf{F}^{\mathrm{T}}\mathbf{F}$ equalled a matrix with unit diagonal elements and all other elements equal to either 0·4 or 0·8; in fact, $\mathbf{F}$ was obtained from the Cholesky decomposition of such matrices.

In the present simulation study, four variables were varied: the number of columns in $\mathbf{X}_k$ was chosen to be $J = 10$ or 20; the number of data matrices per data set was chosen as $K = 3$ or 6; the number of underlying factors was chosen to be $R = 2$ or 3; and the factor scores (in the columns of $\mathbf{P}_k\mathbf{F}$) were chosen to have unit sums of squares and to be mildly related (congruences of 0·4) or strongly related (0·8); n_k was always set equal to J. These choices were fully crossed to give a 2 × 2 × 2 × 2 design; in each cell, five replications were used, thus leading to the number of 80 data sets mentioned above. The sizes of the data were chosen relatively small, because the indirect fitting program was limited to relatively small sizes.

The above 80 data sets were analysed by the direct PARAFAC2 algorithm (programmed in MATLAB (Mathworks Inc., Natick, MA)) as well as by the indirect PARAFAC2 algorithm[13] (programmed in PASCAL) using the constraint that $\mathbf{\Phi}$ be positive semidefinite. In the comparison of the two methods, for both algorithms the rational start was used. As stopping criterion for the algorithm we used that successive function values should differ by less than 0·0001%; and the number of iterations should not exceed 5000.

5.1.1. *Retrieval of perfect fit*

The first issue studied pertained to the quality of the fit. Since we are dealing with perfect data, the fit could be 100% in all cases, but this theoretical maximum can be expected to be *approximated* only by means of the two methods studied here. For both analyses we computed the fit both according to the direct criterion (as defined in (16)) and according to the indirect criterion (defined as (16) with $\mathbf{X}_k$

Table 1. Frequency of finding fit values below various percentages with direct and indirect fitting algorithm[a] applied to 80 data sets in Simulation study I

	Direct fitting algorithm		Indirect fitting algorithm	
Threshold value for fit %	Direct fit %	Indirect fit %	Direct fit %	Indirect fit %
<99·99%	5	2	51	22
<99·90%	0	0	26	9
<99·00%	0	0	6	1

[a] All analyses were based on one rationally started run, using 10^{-4}% as convergence criterion.

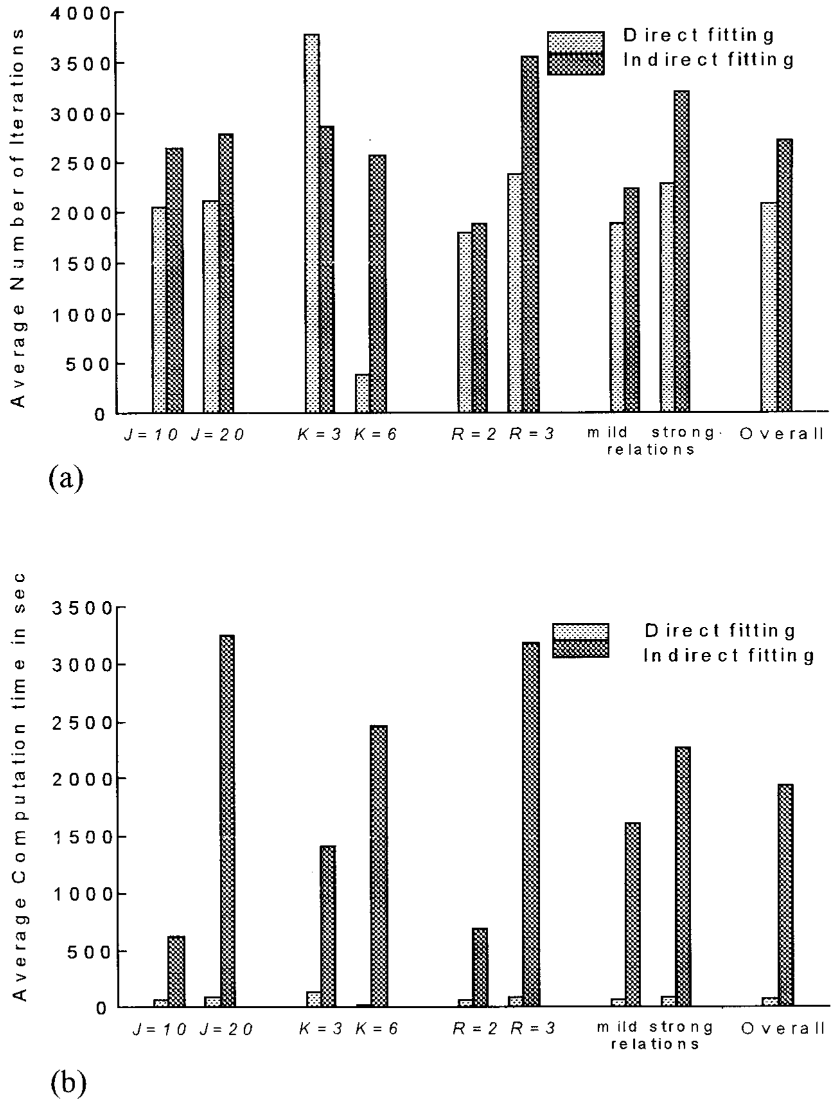

Figure 1. Numbers of iterations and (b) computation times required by direct algorithm (lightly shaded) and indirect algorithm (heavily shaded), averaged over all analyses, as well as over analyses with same value of independent variables in simulation study

replaced by $\mathbf{C}_k$ and with $\mathbf{P}_k\mathbf{F}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}$ replaced by $\mathbf{A}\mathbf{D}_k\mathbf{\Phi}\mathbf{D}_k\mathbf{A}^{\mathrm{T}}$), even though the two methods aimed to maximize only one of these. The frequencies of failures to find a perfect fit from the analyses of the 80 data sets are summarized in Table 1 for different 'definitions' of perfect fit. It can be seen at once that the indirect fitting algorithm rather frequently failed to approximate the 100% fit value very closely, whereas the direct fitting algorithm did so only rarely, and always led to fit percentages higher than 99·9%, in terms of both fit measures. These differences may have various causes. First of all, the methods aim at the minimization of different loss functions, so for that reason alone one would expect the direct fitting algorithm to perform best in terms of the direct fitting criterion, as is indeed the case, and the indirect fitting algorithm to perform best in terms of the indirect fitting criterion, which is not the case. The surprising second finding, that the direct fitting algorithm leads to better indirect fitting criterion values than does the indirect fitting algorithm, implies that the indirect fitting algorithm either led to suboptimal stationary points or was stopped prematurely. This suggests that convergence criteria for the two methods were not comparable and that a more lenient convergence criterion should be used for the indirect fitting procedure. However, this would make the indirect fitting method less efficient, whereas it is doubtful if this would really help.

5.1.2. *Efficiency of the algorithms*

To study the efficiency of the algorithms, we compared numbers of iterations and computation times (on the same Pentium 100 MHz computer) needed by the two methods for the analyses of the 80 data sets. The numbers of iterations for both methods, averaged within all conditions, are displayed in Figure 1(a); the computation times are similarly given in Figure 1(b). It can be seen that both the number of iterations and the computation time are usually considerably larger for the indirect fitting algorithm than for the direct fitting algorithm (the only exception being the number of iterations for the $K = 3$ condition, which was larger for the direct fitting algorithm than for the indirect fitting algorithm). By means of an ANOVA it was verified that the overall difference in numbers of iterations required could be distinguished from chance fluctuations ($p < 0{\cdot}01$), and so could the effect of the number of slabs ($p < 0{\cdot}001$) and the effect of strength of relations between factors ($p < 0{\cdot}05$) on this difference in numbers of iterations required; the number of variables did not have a significant effect on the difference in numbers of iterations required as could be expected upon inspection of Figure 1. Differences in computation times can be seen to increase rapidly with increasing J, K and R and with increasing strength of relations between factors. By means of an ANOVA it was verified that the overall difference, as well as the effects of the design variables on these differences, could be distinguished from chance fluctuations ($p < 0{\cdot}03$ for the effect of strength of relations between factors; for all other effects, $p < 0{\cdot}001$). Given that the currently studied data sets are relatively small compared with what can be found in actual practice, we may expect differences to become even more pronounced in real practice.

It can be concluded that in the present comparison on relatively small data sets the direct fitting algorithm not only fitted better than the indirect fitting algorithm, but also did so much more efficiently. For larger data sets, differences in efficiency can only be expected to grow.

5.1.3. *Sensitivity to hitting suboptimal solutions*

Because the above results clearly point out that the direct fitting algorithm is to be preferred over the indirect one, we stop our comparison of the two algorithms at this point. We do, however, continue our study of the performance of the direct fitting algorithm. As can be inferred from Table 1, the direct fitting algorithm led to fit values over 99·99% in 75 cases. In the other five cases (all associated with $K = 6$, $R = 3$ and strongly related factors), fit values were all over 99·94%, which is still close to perfect, but not as high as desired. In none of these analyses was the maximum of 5000 iterations reached, so apparently in these cases the algorithm stopped because it had reached a suboptimal (e.g. locally optimal) solution or entered a region of very small changes. A standard procedure for avoiding such situations is to use several differently started runs of the algorithm. To see to what extent this is useful and, more generally, how sensitive the algorithm is to hitting suboptimal solutions, we reanalysed all data sets by nine additional randomly started runs of the algorithm. Furthermore, in each run we used 10^{-7}% of the function value as a convergence criterion. The run leading to the highest fit value was taken as the globally optimal solution. Now in *all* 80 cases the best run led to a fit

Table 2. Frequency of hitting suboptimal solutions (fit <99·99%) with direct fitting algorithm[a] in Simulation study I, broken down over conditions $K = 3$ and 6

	Best run (80 in total)	Rationally started run (80 in total)	All ten runs (800 in total)
$K = 3$	0	0	25
$K = 6$	0	5	195
Overall	0	5	220

[a] The 80 data sets were analysed using ten runs (one started rationally and nine randomly), with 10^{-7}% as convergence criterion.

exceeding 99·999%. As can be seen from Table 2, the rationally started runs rarely led to suboptimal solutions, and randomly started runs led to suboptimal solutions more frequently. This happened mainly when $K = 6$. The frequency of finding suboptimal solutions was also related to the other dependent variables (being highest for the highest values of J and R and for strongly related factors), but not as strongly as to the value of K. Thus it can be recommended to use more randomly started runs as K increases. Fortunately, as K increases, the computation time of the algorithm decreases (at least in our simulation study; see Figure 1(b)), so the use of many restarts is not problematic in such situations.

5.2. Simulation study II. Testing the direct fitting algorithm on large data and on data with noise

Simulation study I was limited to relatively small data sets so that the indirect fitting PARAFAC2 algorithm could be used on it. The PASCAL program for that algorithm did not allow for considerably larger data sets. Moreover, extrapolating the results from the simulation study to larger values of J or K clearly indicates that computation times for the indirect fitting program will soon become prohibitive. Furthermore, Simulation study I was limited to noise-free data so that it could be checked if the global optimum was indeed attained. In typical practical applications, n_k and J may be quite large (> 100) and K of intermediate or small size. To study the performance of the direct fitting algorithm for such data sets, 20 data sets were constructed according to (18), with $n_k = 100$, $k = 1, \ldots, K$, $J = 100$, $K = 6$ or 24, $R = 3$ or 2, and using strongly related factors (cross-products of columns of $\mathbf{F}$ equal to 0·8); the conditions for K and R were fully crossed, and for each combination, five data sets were constructed, leading to the total of 20 data sets. The choices for $\mathbf{A}$ and $\mathbf{P}_k$, $k = 1, \ldots, K$, were made as in Simulation study I; the elements of $\mathbf{D}_1,\ldots,\mathbf{D}_K$ were now drawn from the uniform [0·1,1·1] distribution so as to avoid near-zero values in these matrices, but no further precautions were taken. To overcome the second limitation of Simulation study I, in addition to the 20 data sets constructed above, 20 data sets of the same sizes were constructed, but with 25% noise added. This was done as follows: to each matrix $\mathbf{X}_k$ a matrix $\mathbf{N}_k$ of the same size as $\mathbf{X}_k$ was added, which contained elements drawn randomly from the standard normal distribution, multiplied by $0{\cdot}5\|\mathbf{X}_k\|/(Jn_k)^{1/2}$. As a consequence, the expected sum of squares of $\mathbf{N}_k$ equals $0{\cdot}25\|\mathbf{X}_k\|^2$, which clarifies why this is denoted as 25% noise. All data sets were analysed by the direct fitting algorithm with one rationally started run and 19 randomly started runs; the convergence criterion was again taken as 10^{-7}% of the function value.

5.2.1. *Noise-free data*

The results for the *noise-free* data sets are summarized in Table 3. First of all, it can be seen that the fit

Table 3. Results of Simulation study II: analyses of noise-free data sets with direct fitting algorithm[a]

	Overall	$K = 6$, $R = 2$	$K = 6$, $R = 3$	$K = 24$, $R = 2$	$K = 24$, $R = 3$
Frequency of fit >99·9%	20	5	5	5	5
Frequency of fit >99·99%	19	5	5	5	3
Frequency of fit >99·9999%	18	5	5	5	3
Mean number of iterations per run	775	1188	1521	205	188
Mean computation time (s) per run	428	425	664	299	325
Overall occurrence of suboptimal solutions	70%	32%	70%	85%	96%
Frequency of suboptimal solutions with rational start	9	0	3	3	3
Frequency of suboptimal solutions with best run	2	0	0	0	2

[a] The 20 noise-free data sets were analysed using 20 runs (one started rationally and 19 randomly), with 10^{-7}% as convergence criterion.

values were all very high, the lowest two being found when $K = 24$ and $R = 3$. Furthermore, it can be seen that the computation times for the present large data sets are by no means prohibitive, being only a few times as high as for the much smaller data sets in Simulation study I. Specifically, it can be seen that the number of iterations, and to a smaller extent also the computation time, *decreased* as K increased, whereas it *increased* as R increased. Finally, it can be seen that for the present large data sets with strongly related factors the algorithm is quite prone to hitting suboptimal solutions, as defined by the rather strong criterion employed here. On the other hand, relatively few of these pertained to rationally started runs. Furthermore, out of the 20 runs used in each analysis, the best run was suboptimal in only two cases, while the fit values obtained were still high enough for all practical purposes. Thus we conclude that, using 20 differently started runs, among which is at least the rationally started run, the algorithm performs reasonably well for large data sets.

5.2.2. Noisy data

For the *noisy* data sets we do not expect fit values close to 100%. In fact, it was found that the best out of 20 runs all gave fit percentages between 80% and 81%, which corresponds very well to the fact that 25% noise was *added* to the noise-free data: the noise-free part has $(100/125) \times 100\% = 80\%$ of the total sum of squares. Results on the frequency of finding suboptimal solutions (defined as values below 0·9999 times the highest fit value found in 20 runs) can be found in Table 4, where it can be seen that the algorithm is even more prone to suboptimal solutions in the case of the analysis of noisy data. As far as efficiency is concerned, it can be seen from Table 4 that the analyses of noisy data are a

Table 4. Results of Simulation study II: analyses of noisy data sets with direct fitting algorithm[a]

	Overall	$K = 6$, $R = 2$	$K = 6$, $R = 3$	$K = 24$, $R = 2$	$K = 24$, $R = 3$
Overall occurrence of suboptimal solutions	77%	45%	80%	87%	95%
Mean number of iterations per run	358	225	977	93	138
Mean computation time (s) per run	223	80	430	137	244

[a] The 20 noisy data sets were analysed using 20 runs (one started rationally and 19 randomly), with 10^{-7}% as convergence criterion.

bit less time-consuming than those of the noise-free data. The differences between the conditions were comparable with those for the noise-free data. The main topic of interest in the analysis of noisy data is, of course, whether the noise-free parts of the data are accurately recovered. In other words, the question is to what extent the underlying parameters are recovered. Adequate recovery, however, presupposes uniqueness of the model estimates, which will be discussed in Section 6.

6. UNIQUENESS OF THE DIRECT PARAFAC2 SOLUTION

6.1. Proving uniqueness under certain conditions

For the indirect PARAFAC2 model, some results have been obtained concerning uniqueness of the parameter estimates.[14,15] Basically, these results imply that, under certain conditions (here denoted as uniqueness conditions), PARAFAC2 estimates are 'essentially unique'. An indirect PARAFAC2 solution (see (6)) is considered 'essentially unique' if $\mathbf{A}$, $\mathbf{D}_1, \ldots, \mathbf{D}_K$ and $\mathbf{\Phi}$ are determined up to:

(1) joint permutations of the columns of $\mathbf{A}$ and the columns and rows of $\mathbf{D}_1, \ldots, \mathbf{D}_K$ and $\mathbf{\Phi}$
(2) arbitrary scalings/reflections of the columns of $\mathbf{A}$ and of the supermatrix $(\mathbf{D}_1,\ldots,\mathbf{D}_K)^{\mathrm{T}}$, combined with the inverse scalings/reflections of the columns and rows of $\mathbf{\Phi}$
(3) reflections of any subset of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$.

For the case where $R = 2$, it has been proven[15] that, almost surely (i.e. with probability one in practice), the indirect PARAFAC2 estimates are essentially unique under the following uniqueness conditions.

1. $K \geq 4$.
2. $\mathbf{\Phi}$ is positive definite.
3. A has full column rank.
4. At least four of the matrices $\mathbf{D}_1, \ldots, \mathbf{D}_K$ are non-singular and mutually not proportional.

For the variant of PARAFAC2 where $\mathbf{D}_1, \ldots, \mathbf{D}_K$ are constrained to be non-negative, the first condition should be replaced by $K \geq 3$. These conditions are 'sharp'. As soon as $K < 4$, (unconstrained) PARAFAC2 with $R = 2$ is no longer unique; likewise, as soon as $K < 3$, constrained PARAFAC2 with $R = 2$ is not unique. For the case where $R > 2$, it has been proven[14] that essential uniqueness for a general model incorporating PARAFAC2 holds under a set of uniqueness conditions, among which is the condition that $K \geq R(R+1)(R+2)(R+3)/24$ (see Reference 14, pp. 145–146). This condition is sufficient for uniqueness in practice, but not necessary (see Reference 14, p. 152), as follows at once from the fact that for $R = 2$ it specifies $K \geq 5$ (rather than $K \geq 4$). For $R \geq 3$ it can be expected that the condition for K can be relaxed even more. For instance, for $R = 3$ it specifies $K \geq 15$, whereas in Reference 15 (p. 130) it is reported, on the basis of computer simulations, that $K \geq 5$ seems to suffice for uniqueness.

The results mentioned above have been derived specifically for the indirect PARAFAC2 model and pertain only to the parameters therein. It will be shown now that these results are directly related to those for the direct PARAFAC2 model.

Definition

The parameters of the direct PARAFAC2 model are called 'essentially unique' if $\mathbf{A}$, $\mathbf{D}_1, \ldots, \mathbf{D}_K$ and $\mathbf{F}_1, \ldots, \mathbf{F}_K$ (except for those columns of $\mathbf{F}_k$ that correspond to zero diagonal values in the associated matrix $\mathbf{D}_k$) are uniquely determined up to

(1) joint permutations of the columns of $\mathbf{A}$ and $\mathbf{F}_1, \ldots, \mathbf{F}_K$ and of the rows and columns of $\mathbf{D}_1,\ldots,\mathbf{D}_K$

(2) arbitrary scalings/reflections of the columns of $\mathbf{A}$ and of the supermatrix $(\mathbf{D}_1,\ldots,\mathbf{D}_K)^{\mathrm{T}}$, provided that the inverse scalings/reflections of these are applied to the supermatrix $(\mathbf{F}_1^{\mathrm{T}},\ldots,\mathbf{F}_K^{\mathrm{T}})^{\mathrm{T}}$

(3) reflections of any subset of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$, provided that these are accompanied by reflections of the corresponding subset of the matrices $\mathbf{F}_1,\ldots,\mathbf{F}_K$.

Solutions satisfying the above definition are called ‘essentially unique’, because, as far as they are not uniquely determined (e.g. in terms of ordering or scaling of factors), this has no implications for the interpretation of the solution; even the possibility of total indeterminacy of certain columns of $\mathbf{F}_k$ is inconsequential, because it pertains only to columns that, in the model, are weighted by zero (as they occur in the product $\mathbf{F}_k\mathbf{D}_k$).

Theorem

The parameters of the direct PARAFAC2 model are essentially unique if and only if the parameters of the indirect PARAFAC2 model are essentially unique, provided that the ‘loading’ matrix ($\mathbf{A}$) has full column rank, the ‘factor cross-product matrix’ ($\mathbf{\Phi}$) is positive semidefinite and at least one of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$ is non-singular.

Proof

See Appendix.

As a consequence of the above Theorem, we know that the uniqueness conditions (which include $\mathbf{A}$ having full column rank and at least four of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$ being non-singular) specified for the indirect PARAFAC2 model[14,15] also pertain to the direct PARAFAC2 model. Thus, for example, for $R = 2$ the direct PARAFAC2 model can be expected to give unique solutions in practice as soon as $K \geq 4$. Furthermore, for $R = 3$, uniqueness can be expected to hold if $K \geq 5$. In fact, the latter conjecture can now be tested by using the direct fitting PARAFAC2 algorithm, which has been seen to be more well behaved than the indirect fitting PARAFAC2 algorithm. If such analyses applied to simulated data suggest uniqueness of the direct PARAFAC2 model under certain conditions, our Theorem implies that these should also lead to uniqueness in the indirect PARAFAC2 model. In the next subsection the results of such simulations will be reported.

6.2. Checking uniqueness by means of simulation studies

The uniqueness results derived here imply that, in the simulation studies described in Section 5, we should recover the original matrices $\mathbf{A}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$ and $\mathbf{F}_1,\ldots,\mathbf{F}_K$, up to columnwise permutations, scalings and reflections, in the cases where $K = 6$ and $R = 2$. For the other cases, where $K = 3$ and $R = 2$, recovery would depend on the signs of the elements of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$. If these remain non-negative, then full recovery of the parameter matrices is expected as well.[15] For the cases with $R = 3$, uniqueness is known to hold if $K \geq 15$, but is expected to be obtained for much smaller values of K as well. For all data sets in the simulation studies we checked to what extent the original parameter matrices were recovered, by computing congruence coefficients (φ; see equation (5)) among corresponding columns of the original and the obtained parameter matrices (after adequate permutation and reflection). The parameter matrices were $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{F}_{\text{sup}}$, where $\mathbf{C}$ contains the diagonals of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$ as its rows, and $\mathbf{F}_{\text{sup}} = [\mathbf{F}_1^{\mathrm{T}}\ldots\mathbf{F}_K^{\mathrm{T}}]^{\mathrm{T}}$. Recovery up to scaling implies that these φ values should be unity. In practice, however, recovery will not be perfect owing to finite convergence, but also owing to the possibility that the uniqueness conditions are almost violated (e.g. having some elements of $\mathbf{D}_1,\ldots,\mathbf{D}_K$ very close to zero). Here we consider a matrix recovered if the average φ value exceeds 0·99.

The results for the two simulation studies in Section 5 are reported in the first two panels of Table 5.

Table 5. Frequency of recovery ($\varphi > 0{\cdot}99$) of **A**, **C** and $\mathbf{F}_{\text{sup}}$ in each simulation study

	A	**C**	$\mathbf{F}_{\text{sup}}$
Study I (ten runs, convergence at 10^{-7}%)			
$K = 3$ ($R = 2,3$; $n_k = J = 10,20$); 40 sets	15	13	8
$K = 6$ ($R = 2,3$; $n_k = J = 10,20$; 40 sets	40	40	38[a]
Study II (20 runs, convergence at 10^{-7}%)			
Noise-free ($K = 6,24$; $R = 2,3$; $n_k = J = 100$); 20 sets	20	20	18[b]
Noisy ($K = 6,24$; $R = 2,3$; $n_k = J = 100$); 20 sets	20	20	15
Study III (20 runs, convergence at 10^{-7}%)			
$K = 4$ ($R = 2,3,4,5,6$; $n_k = J = 10$); 40 sets	40	40	40
$K = 3$ ($R = 2,3,4,5,6$; $n_k = J = 10$); 40 sets	9	15	10

[a] In the two failing cases, values in $\mathbf{D}_k$ were close to zero.
[b] In the two failing cases, φ values were still large ($> 0{\cdot}96$).

Here recovery always pertains to the solution from the run that gave the highest fit value. It can be seen that recovery occurred surprisingly often. Uniqueness, and hence recovery, is only known for the 20 cases with $K = 6$ and $R = 2$, whereas now recovery was found in *all* 40 cases with $K = 6$ (both $R = 2$ and 3), disregarding the two cases with near-zero values in the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$ that caused $\mathbf{F}_{\text{sup}}$ not to be recovered. Thus the proven uniqueness was confirmed in all cases with $K = 6$ and $R = 2$, and, more interestingly, uniqueness was also indicated for cases with $K = 6$ and $R = 3$, for which no proven uniqueness results are available as yet; on the other hand, the results for the cases with $K = 3$ (both $R = 2$ and 3) demonstrate clearly that, although sometimes the original parameter values are recovered rather closely, the many exceptions imply that, for these cases, uniqueness certainly does not hold.

The results from Simulation study II for the *noise-free data* show uniqueness not only in the cases with $R = 2$ (as expected), but also in those with $R = 3$, disregarding the two cases for which $\mathbf{F}_{\text{sup}}$ was not recovered as accurately as desired (which, moreover, corresponded to the cases with suboptimal fit percentages). For the *noisy* data, perfect recovery could not be expected, but it can be seen from Table 5 that **A** and **C** *were* recovered very well, and in the five cases where $\mathbf{F}_{\text{sup}}$ was not recovered up to $\varphi > 0{\cdot}99$, it did give $\varphi > 0{\cdot}95$. Thus, even with noisy data, the direct fitting PARAFAC2 algorithm is able to adequately recover the underlying parameters.

The above surprising findings of recovery with $R = 3$ stimulated us to study possible uniqueness in various other situations. It is known[15] that, for $R = 2$, uniqueness is always found in practice if $K \geq 4$. The results reported above suggest that, for cases with $R = 3$, uniqueness holds at least if $K = 6$. However, it is possible that, for $R \geq 3$, uniqueness holds even for smaller K. On the basis of counting equations and unknowns, and the assumption that uniqueness holds if the former exceeds or equals the latter, we conjecture that uniqueness holds in practice for any value of R, provided that $K \geq 4$. We tested this conjecture by means of Simulation study III, set up analogously to the earlier simulation studies. Specifically, data sets were constructed according to (18), with $n_k = J = 10$, and with parameter matrices constructed in largely the same way as earlier, except that now cases where the congruence coefficient between the diagonals of a pair of $\mathbf{D}_k$ matrices exceeded 0·9 were excluded so as to avoid non-uniqueness due to near duplication of $\mathbf{D}_k$ matrices. Forty such data sets were constructed with $K = 4$ and $R = 3$–6 (10 for each case). After analysing these data sets, it was checked whether the original parameter matrices were recovered, which would be a confirmation of our uniqueness conjecture. In addition, 40 data sets were constructed for $K = 3$ and $R = 3$–6 (again 10 for each case); for these cases, uniqueness was expected not to hold, as would be confirmed by clear failures of recovery of the original parameter matrices. All data sets were analysed by the direct fitting algorithm, with $10^{-7}\%$ as convergence criterion, and using 20 differently started runs. The results, in the bottom panel of Table 5, clearly confirmed our conjecture. In the 40 cases with $K = 4$, **A**, **C** and $\mathbf{F}_{\text{sup}}$ were recovered *always*. For the 40 cases with $K = 3$, on the other hand, the original parameters were recovered infrequently. In all cases the fit value exceeded 99·99%, so the failures of recovery cannot be attributed to failure in fit. In fact, in cases where the parameter matrices were not recovered, the average φ values frequently fell below 0·8. Such low values were encountered even in cases where the fit value exceeded 99·999999%, which takes away all doubt about whether the results really pertain to non-uniqueness rather than to inaccuracies. Thus we conclude that our simulations sustain the conjecture that, in practice, the direct PARAFAC2 model gives unique solutions if $K \geq 4$ (irrespective of R), whereas uniqueness is not guaranteed if $K = 3$ (irrespective of R). Furthermore, if the direct PARAFAC2 model is unique under such conditions, then so is the indirect PARAFAC2 model, so our conjecture supplements proven results[15] with the empirically sustained conjecture that $K \geq 4$ suffices for uniqueness in practice.

7. MODIFICATION OF DIRECT PARAFAC2 ALGORITHM

In some applications it is useful to constrain some of the parameter sets to have, for instance, non-negative values only. Because, as far as the parameter matrices **F**, $\mathbf{D}_k$ and **A** are concerned, the direct PARAFAC2 algorithm uses steps from a PARAFAC1 algorithm, these parameter matrices can be constrained in exactly the same way as they can be constrained in PARAFAC1 algorithms. Thus it is straightforward to constrain, for instance, the matrix **A** to non-negativity, which is not at all straightforward in the indirect PARAFAC2 algorithm. Non-negativity constraints on the matrices $\mathbf{D}_k$ can be implemented straightforwardly as well. It should be noted, however, that the use of these constraints may frequently lead to zero values in some of the matrices $\mathbf{D}_1,\ldots,\mathbf{D}_K$ and hence violate one of the uniqueness assumptions. For $R = 2$ this has been shown not to violate uniqueness (Reference 15, pp. 130–131) as long as at least three mutually non-proportional matrices $\mathbf{D}_k$ are obtained. For $R > 2$ no such results are available as yet, so it must be kept in mind that uniqueness may no longer hold if too few non-singular matrices $\mathbf{D}_k$ remain. Unfortunately, non-negativity constraints on the matrices $\mathbf{F}_k$ cannot be imposed in a straightforward manner, because it involves constraining the product of two matrices, one of which is itself constrained to be columnwise orthonormal.

Another modification of the PARAFAC2 algorithm is needed for the situation where data are missing. With the current direct algorithm this may be handled by straightforward iterative imputation of missing data after each full cycle of updates.

As a second extension, complex data structures consisting of sets of three- or higher-way arrays can easily be handled by the present PARAFAC2 approach (also see Reference 16). The main modification consists of replacing the PARAFAC1 updates by more general N-way PARAFAC updates. To explain this in more detail, below we consider the case with a *set* of *three-way* arrays.

Let a set of K three-way arrays $\underline{\mathbf{X}}_k$ of orders $n_k \times J_1 \times J_2$, $k = 1,\ldots,K$, be given and let each of these be modelled by a PARAFAC1 model with parameter matrices $\mathbf{G}_k$, **A** and **B**. Defining $\mathbf{X}_k$ as the $n_k \times J_1J_2$ supermatrix with frontal planes of $\underline{\mathbf{X}}_k$ next to each other, then we can describe the PARAFAC1 model for $\underline{\mathbf{X}}_k$ as[3]

$$\mathbf{X}_k = \mathbf{G}_k(\mathbf{B} \times \mathbf{A})^{\mathrm{T}} + \mathbf{E}_k \tag{19}$$

$k = 1,\ldots,K$, where × denotes the columnwise Kronecker product (also denoted as the Khatri–Rao product[17]). Furthermore, let the models be related to each other by the constraint that $\mathbf{G}_k = \mathbf{F}_k\mathbf{D}_k$, with $\mathbf{D}_k$ diagonal and $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k$ invariant over k, $k = 1,\ldots,K$. Then we have a direct four-way generalization of the original PARAFAC2 model that can be written as

$$\mathbf{X}_k = \mathbf{P}_k\mathbf{F}\mathbf{D}_k(\mathbf{B}\times\mathbf{A})^{\mathrm{T}} + \mathbf{E}_k \tag{20}$$

This model can be fitted in the least squares sense by using an alternating least squares algorithm, where $\mathbf{P}_k$ is updated by maximizing

$$f^*(\mathbf{P}_k) = \mathrm{tr}\ \mathbf{F}\mathbf{D}_k(\mathbf{B}\times\mathbf{A})^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}}\mathbf{P}_k \tag{21}$$

$k = 1,\ldots,K$, in analogy to (9), i.e. by using the SVD $\mathbf{F}\mathbf{D}_k(\mathbf{B}\times\mathbf{A})^{\mathrm{T}}\mathbf{X}_k^{\mathrm{T}} = \mathbf{U}_k\mathbf{\Delta}_k\mathbf{V}_k^{\mathrm{T}}$ and updating $\mathbf{P}_k$ as $\mathbf{V}_k\mathbf{U}_k^{\mathrm{T}}$. To update the other parameter matrices, we have, in analogy to (11), to minimize (or at least decrease) the function

$$g^*(\mathbf{F},\mathbf{A},\mathbf{B},\mathbf{D}_1,\ldots,\mathbf{D}_K) = \sum_{k=1}^{K} \|\mathbf{P}_k^{\mathrm{T}}\mathbf{X}_k - \mathbf{F}\mathbf{D}_k(\mathbf{B}\times\mathbf{A})^{\mathrm{T}}\|^2 + c \tag{22}$$

over $\mathbf{F}$, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{D}_1,\ldots,\mathbf{D}_K$. This minimization problem can be recognized as a four-way PARAFAC problem for the four-way array consisting of the elements of the supermatrices $\mathbf{P}_k^{\mathrm{T}}\mathbf{X}_k$, $k = 1,\ldots,K$. Algorithms for the four-way PARAFAC problem can be and have been devised in complete analogy to the three-way PARAFAC algorithm.[3] Thus a first generalization of the PARAFAC2 model has been described. Further generalizations involving sets of N-way arrays can be made straightforwardly along the same lines. The algorithms will consist of iteratively updating $\mathbf{P}_k$ on the one hand and using a full cycle of an (N+1)-way PARAFAC1 algorithm on the other hand.

8. DISCUSSION

The direct fitting algorithm for the PARAFAC2 model has been demonstrated to have various advantages over the indirect fitting algorithm. The direct algorithm is more efficient, yields parameters for the first-mode (row) units, offers a possibility for handling missing data by iterative data imputation, allows for easy implementation of various constraints on the parameter matrices, fits the data according to a more sensible loss function and is straightforwardly generalized to the analysis of higher-way data structures. At first sight a disadvantage seems to be that the direct fitting approach has to deal with an often large first mode, but it has been demonstrated that this can be circumvented effectively.

In the simulation study for comparing direct PARAFAC2 and indirect PARAFAC2, mainly small data arrays were used. The differences in efficiency have been seen to increase as the data size increases. Furthermore, it has been seen that the direct PARAFAC2 algorithm performs reasonably well on large data sets with strongly related factors, and that computation time increases only slowly with increasing data size. The performance of the direct PARAFAC2 algorithm on large data arrays is further illustrated in the application in Part II.[11] Furthermore, the direct PARAFAC2 solutions have been found to recover underlying structures adequately in the cases where uniqueness was proven to hold, provided that the global optimum is indeed approximated sufficiently closely. The algorithm also recovered underlying structures in many cases where uniqueness has not been proven; in fact, simulations suggest that the model gives unique solutions as soon as $K \geq 4$, for arbitrary values of R.

In the simulation study we also inspected the sensitivity of the direct PARAFAC2 algorithm to hitting suboptimal solutions. It turned out that for the small data sets the rational start led to suboptimal solutions only rarely, but suboptimal solutions were found more frequently as the data size increased. To be on the safe side, it can be recommended to use several randomly started additional runs of the algorithm, especially in conditions where data sets consist of large numbers of (large) data matrices with strongly related factors.

MATLAB M-files for direct fitting of the PARAFAC2 model are available from http://newton.foodsci.kvl.dk/srccode.html.

ACKNOWLEDGEMENTS

The authors are obliged to Richard A. Harshman for stimulating comments on this research. For Rasmus Bro, financial support was obtained from Professor Lars Munck, Food Technology, Department of Dairy and Food Science, Royal Veterinary and Agricultural University, Denmark, through the Nordic Industry Foundation project P93149 and FØTEK.

APPENDIX

This appendix gives the proof of the Theorem on equivalence of essential uniqueness for the direct and indirect PARAFAC2 models. The 'if and only if' parts will be proven subsequently under (a) and (b).

(a) Proof that essential uniqueness of the indirect PARAFAC2 model implies essential uniqueness of the direct PARAFAC2 model

We assume that the indirect PARAFAC2 model gives essentially unique parameter estimates. Suppose the two direct PARAFAC2 solutions [$\mathbf{A}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$, $\mathbf{F}_1,\ldots,\mathbf{F}_K$] and [$\mathbf{B}$, $\mathbf{E}_1,\ldots,\mathbf{E}_K$, $\mathbf{G}_1,\ldots,\mathbf{G}_K$], with $\mathbf{F}_k^{\mathrm{T}}\mathbf{F}_k = \mathbf{\Phi}$ and $\mathbf{G}_k^{\mathrm{T}}\mathbf{G}_k = \mathbf{\Gamma}$, $k = 1,\ldots, K$, give the same estimates for $\mathbf{X}_1,\ldots,\mathbf{X}_K$. Then we have

$$\mathbf{F}_k\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{G}_k\mathbf{E}_k\mathbf{B}^{\mathrm{T}} \tag{23}$$

$k = 1,\ldots, K$. It follows from (23) that

$$\mathbf{A}\mathbf{D}_k\mathbf{\Phi}\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{B}\mathbf{E}_k\mathbf{\Gamma}\mathbf{E}_k\mathbf{B}^{\mathrm{T}} \tag{24}$$

which describes the indirect PARAFAC2 model on both sides of the equality sign. Essential uniqueness of the parameters of the indirect PARAFAC2 model implies that

$$\mathbf{B} = \mathbf{A}\mathbf{S}_1\mathbf{\Pi} \tag{25a}$$

$$\mathbf{E}_k = \sigma_k\mathbf{\Pi}^{\mathrm{T}}\mathbf{D}_k\mathbf{S}_2\mathbf{\Pi}, \quad k = 1,\ldots,K \tag{25b}$$

and

$$\mathbf{\Gamma} = \mathbf{\Pi}^{\mathrm{T}}\mathbf{S}_2^{-1}\mathbf{S}_1^{-1}\mathbf{\Phi}\mathbf{S}_1^{-1}\mathbf{S}_2^{-1}\mathbf{\Pi} \tag{25c}$$

where $\mathbf{\Pi}$ is a permutation matrix, $\mathbf{S}_1$ and $\mathbf{S}_2$ are diagonal (scaling/reflection) matrices and σ_k is a variable with value 1 or -1, $k = 1,\ldots,K$. Combining (23) with (25a),(b)), we obtain

$$\mathbf{F}_k\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{G}_k\sigma_k\mathbf{\Pi}^{\mathrm{T}}\mathbf{D}_k\mathbf{S}_2\mathbf{\Pi}\mathbf{\Pi}^{\mathrm{T}}\mathbf{S}_1\mathbf{A}^{\mathrm{T}} = \sigma_k\mathbf{G}_k\mathbf{\Pi}^{\mathrm{T}}\mathbf{S}_2\mathbf{S}_1\mathbf{D}_k\mathbf{A}^{\mathrm{T}} \tag{26}$$

Using the fact that **A** has full column rank (see the third condition of the Theorem) and assuming that $\mathbf{D}_k$ is non-singular, we find, upon postmultiplying both sides by the expression $\mathbf{A}(\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{D}_k^{-1}$, that

$$\mathbf{G}_k = \sigma_k \mathbf{F}_k \mathbf{S}_1^{-1}\mathbf{S}_2^{-1}\boldsymbol{\Pi} \tag{27}$$

$k = 1,\ldots, K$, which demonstrates essential uniqueness of the direct PARAFAC2 solution under the assumption that $\mathbf{D}_k$ is non-singular. When $\mathbf{D}_k$ is singular, (27) still holds for the columns of $\mathbf{F}_k$ corresponding to the non-zero diagonal elements of $\mathbf{D}_k$, and thus essential uniqueness is again obtained.

(b) Proof that essential uniqueness of the direct PARAFAC2 model implies essential uniqueness of the indirect PARAFAC2 model

We assume that the direct PARAFAC2 model gives essentially unique parameter estimates. Suppose the two indirect PARAFAC2 solutions [$\mathbf{A}$, $\mathbf{D}_1,\ldots,\mathbf{D}_K$, $\boldsymbol{\Phi}$] and [$\mathbf{B}$, $\mathbf{E}_1,\ldots,\mathbf{E}_K$, $\boldsymbol{\Gamma}$] give the same estimates for $\mathbf{X}_1^{\mathrm{T}}\mathbf{X}_1, \ldots, \mathbf{X}_K^{\mathrm{T}}\mathbf{X}_K$. Then we have

$$\mathbf{A}\mathbf{D}_k\boldsymbol{\Phi}\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{B}\mathbf{E}_k\boldsymbol{\Gamma}\mathbf{E}_k\mathbf{B}^{\mathrm{T}} \tag{28}$$

$k = 1,\ldots, K$, from which it follows that

$$\boldsymbol{\Phi}^{1/2}\mathbf{D}_k\mathbf{A}^{\mathrm{T}} = \mathbf{N}_k\boldsymbol{\Gamma}^{1/2}\mathbf{E}_k\mathbf{B}^{\mathrm{T}} \tag{29}$$

for certain columnwise orthonormal matrices $\mathbf{N}_k$, $k = 1,\ldots,K$. Essential uniqueness of the parameters of the direct PARAFAC2 model implies that

$$\mathbf{B} = \mathbf{A}\mathbf{S}_1\boldsymbol{\Pi} \tag{30a}$$

$$\mathbf{E}_k = \sigma_k\boldsymbol{\Pi}^{\mathrm{T}}\mathbf{D}_k\mathbf{S}_2\boldsymbol{\Pi}, \quad k = 1, \ldots, K \tag{30b}$$

and

$$\mathbf{N}_k\boldsymbol{\Gamma}^{1/2} = \sigma_k\boldsymbol{\Phi}^{1/2}\mathbf{S}_1^{-1}\mathbf{S}_2^{-1}\boldsymbol{\Pi} \quad \text{for each value of } k \text{ for which } \mathbf{D}_k \text{ is non-singular} \tag{30c}$$

where $\boldsymbol{\Pi}$,$\mathbf{S}_1$, $\mathbf{S}_2$ and σ_k are defined as above. Taking cross-products of the columns of both sides of (30c), we find

$$\boldsymbol{\Gamma} = \sigma_k{}^2\boldsymbol{\Pi}^{\mathrm{T}}\mathbf{S}_2^{-1}\mathbf{S}_1^{-1}\boldsymbol{\Phi}\mathbf{S}_1^{-1}\mathbf{S}_2^{-1}\boldsymbol{\Pi} = \boldsymbol{\Pi}^{\mathrm{T}}\mathbf{S}_2^{-1}\mathbf{S}_1^{-1}\boldsymbol{\Phi}\mathbf{S}_1^{-1}\mathbf{S}_2^{-1}\boldsymbol{\Pi} \tag{31}$$

which, together with (30a),(b), establishes essential uniqueness of the parameters of the indirect PARAFAC solution.

REFERENCES

1. R. B. Cattell, *Psychometrika*, **9**, 267 (1944).
2. R. A. Harshman, *UCLA Working Papers Phonet.* **16**, 1 (1970).
3. J. D. Carroll and J. -J. Chang, *Psychometrika*, **35**, 283–319 (1970).
4. P. Paatero, *Chemometrics Intell. Lab. Syst.* **38**, 223 (1997).
5. H. A. L. Kiers, *J. Chemometrics*, **12**, 155 (1998).
6. R. A. Harshman, *UCLA Working Papers Phonet.* **22**, 111 (1972).
7. J. B. Kruskal, *Linear Algebra Appl.* **18**, 95 (1977).
8. R. A. Harshman, *UCLA Working Papers Phonet.* **22**, 33 (1972).
9. L. R. Tucker, 'A method for synthesis of factor analysis studies', *Personnel Research Section Report 984,* Department of the Army, Washington, DC (1951).
10. B. C. Mitchell and D. S. Burdick, *Chemometrics Intell. Lab. Syst.* **20**, 149 (1993).
11. R. Bro, C. A. Andersson and H. A. L. Kiers, 'PARAFAC2— Part II. Modelling chromatographic data with retention time shifts', *J. Chemometrics*, **13**, (1999).
12. B. F. Green, *Psychometrika*, **17**, 429 (1952).
13. H. A. L. Kiers, *Comput. Statist. Data Anal.* **16**, 103 (1993).
14. R. A. Harshman and M. E. Lundy, *Psychometrika*, **61**, 133 (1996).
15. J. M. F. ten Berge and H. A. L. Kiers, *Psychometrika*, **61**, 123 (1996).
16. R. Bro, 'Multi-way analysis in the food industry: models, algorithms, & applications', *Doctoral Dissertation*, University of Amsterdam (1998).
17. C. R. Rao and S. K. Mitra, *Generalized Inverses of Matrices and Its Applications*, Wiley, New York (1971).