Available online at www.sciencedirect.com

SCIENCE DIRECT®

Chemometrics and Intelligent Laboratory Systems 75 (2005) 163–180

Chemometrics and intelligent laboratory systems

www.elsevier.com/locate/chemolab

# PARAFAC and missing values

Giorgio Tomasi*, Rasmus Bro

*Food Technology, Royal Veterinary and Agricultural University, Rolighedsvej 30, 1958 Frederiksberg C, Denmark*

Received 1 December 2003; received in revised form 8 July 2004; accepted 9 July 2004
Available online 11 September 2004

**Abstract**

Missing values are a common occurrence in chemometrics data, and different approaches have been proposed to deal with them. In this work, two different concepts based on two algorithms are compared in their efficiency in dealing with incomplete data when fitting the PARAFAC model: single imputation (SI) combined with a standard PARAFAC-alternating least squares (ALS) algorithm, and fitting the model only to the existing elements using a computationally more expensive method (Levenberg–Marquadt) appropriately modified and optimised.

The performance of these two algorithms and the effect of the incompleteness of the data on the final model have been evaluated on the basis of a Monte Carlo study and real data sets with different amounts and patterns of missing values (randomly missing values, randomly missing spectra/vectors, and systematically missing spectra/vectors).

The evaluation is based on the quality of the solution as well as on computational aspects (time requirement and number of iterations). The results show that a PARAFAC model can be correctly determined even when a large fraction of the data is missing (up to 70%), and that the pattern matters more than the fraction of missing values. Computationally, the Levenberg–Marquadt-based approach appeared superior for the pattern of missing values typical of fluorescence measurements when the fraction of missing elements exceeded 30%.


*Keywords:* PARAFAC; Missing values; INDAFAC; Fluorescence

## 1. Introduction

In chemometrics, incomplete observations and missing values can be found in a large number of applications ranging from calibration problems to statistical process control. Recent studies have pursued the algorithmic problem in connection with missing values for two-way models [1–4], with specific focus on PCA and PLS and, to a certain extent, three-way models [1,5,6]. The aim of this paper is to study the effect of non-observed values on fitting a PARAFAC model and to compare the performances of two algorithms fitting such model in presence of missing values: PARAFAC-alternating least squares (ALS) with single imputation (ALS-SI) and the least squares approach called INcomplete DAta paraFAC (INDAFAC) based on a suitably modified Levenberg–Marquadt algorithm.

This study is based on a Monte Carlo simulation where 2400 data sets were generated varying a specific set of conditions (rank of the array, percentage of missing elements and their pattern in the array, collinearity between factors, and level of noise) and on three real data sets comprising fluorescence measurements and having known rank.

### 1.1. PARAFAC model

If one considers a three-way array $\underline{\mathbf{X}}$ of dimensions $I \times J \times K$, the PARAFAC model can be expressed as

$$x_{ijk} = \sum_{f=1}^{F} a_{if} b_{jf} c_{kf} + r_{ijk} \qquad i = 1 \ldots I,\ j = 1 \ldots J,\ k = 1 \ldots K \tag{1}$$

* Corresponding author.
*E-mail address:* gt@kvl.dk (G. Tomasi).

0169-7439/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.chemolab.2004.07.003

where $x_{ijk}$ is the measured value, $a_{if}$, $b_{jf}$, and $c_{kf}$ represent the parameters to estimate, $r_{ijk}$ are the residuals, and $F$ is the number of sought factors.

By defining the three loading matrices:

$$\mathbf{A} = \{a_{if} | i = 1...I,\ f = 1...F\}$$

$$\mathbf{B} = \{b_{jf} | j = 1...J,\ f = 1...F\}$$

$$\mathbf{C} = \{c_{kf} | k = 1...K,\ f = 1...F\} \quad (2)$$

and employing the column-wise Khatri–Rao product ($\odot$) [7], Eq. (1) can be written as

$$\mathbf{X}^{(I\times JK)} = \mathbf{A}(\mathbf{C}\odot\mathbf{B})^{\mathrm{T}} + \mathbf{R}^{(I\times JK)}, \quad (3)$$

where $\mathbf{X}^{(I\times JK)}$ is the matricised form of the data array [the superscript $(I\times JK)$ identifying the way the array is matricised; [7], and the superscript T indicates the transpose operation.

Fitting the PARAFAC model to $\underline{\mathbf{X}}$ in the least squares sense can be expressed as the minimisation problem:

$$\underset{\mathbf{A},\mathbf{B},\mathbf{C}}{\arg\min} \left\| \mathbf{X}^{(I\times JK)} - \mathbf{A}(\mathbf{C}\odot\mathbf{B})^{\mathrm{T}} \right\|_{\mathrm{F}}^{2} \quad (4)$$

where $\|\mathbf{Y}\|_{\mathrm{F}}$ is the Frobenius norm (i.e., the squared root of the sum of the squared elements of the matrix $\mathbf{Y}$).

Solving problem (4) corresponds to fitting the PARAFAC model in the maximum likelihood sense provided that the residuals $\mathbf{r}=\mathrm{vec}\mathbf{R}^{(I\times JK)}$ (where the vec operator is defined as in Ref. [8]) are normally distributed with mean 0 and variance $\sigma^2\mathbf{I}$ [9], viz. that the noise is uncorrelated and homoscedastic. Albeit for real life problems this is hardly ever the case, it has been shown in several applications that such fitting is adequate also when slight deviations occur [7].

Numerous algorithms have been proposed for solving problem (4) [10,11], two of them, namely, PARAFAC-ALS with single imputation and PARAFAC-LM (where LM stands for Levenberg–Marquadt) can be effectively employed in presence of missing values and are described in Section 2.

### 1.2. Missing values patterns

Missing values may occur in data sets for a number of reasons: glitches and malfunctions of one or more sensors, irregular measurement intervals between samples, or different sampling frequencies for the various sensors. In some cases (e.g., fluorescence Emission/Excitation Matrices—EEM), the missing values are not necessarily present originally in the data as obtained from the instrument, but are inserted as a postprocessing to yield data more suitable for being described by a multilinear model [6,7]. Depending on the cause for the missing values, their pattern within the array may change considerably, having different effects on the model fitting process.

In the simplest case to treat, but also the one that is most seldom found in practice, the missing elements are randomly scattered over the array without any specific pattern (Fig. 1a). One such situation may occur when several, distinct sensors are used to monitor one process in time and there are momentary malfunctions in the single sensor. Analogously, a survey of several variables both in time and space may not follow a particularly regular pattern, and certain sites (e.g., the least accessible ones) may be visited with lower frequency. Such a pattern is referred to as randomly missing values (RMV).

A second pattern, here denoted as randomly missing spectra (RMS), encompasses the case of entirely missing "tubes" (Fig. 1b), once again completely at random. This situation may occur when a process is monitored in time by means of a multivariate instrument (e.g., a spectrometer). If the measurement is not taken, either due to malfunctioning or caused by an irregular sampling scheme, a whole spectrum (i.e., a tube) will be missing. An analogous situation would present itself if a certain sensor or channel stops working and is not replaced until the process is terminated; only in this case the "tube" would be missing in the time mode of the array rather than in the spectral one.

Finally, the missing values pattern may be completely systematic, as, for example, would happen if the sensors used for the monitoring of a process have a different sampling frequency. Indeed, many cases of systematically missing values (SMV) can be identified. One that is particularly interesting, because it is common for the kind of data to which PARAFAC is often applied, is represented by EEM fluorescence measurements. In fluorescence, the signal registered at emission wavelengths lower than the excitation wavelength is physically zero (Fig. 1c). The presence of these zeros, however, may interfere with the multilinearity of the data [6], provoking artefacts in the final solution. At the same time, Raman and Rayleigh scatter (Fig. 1c), cannot be adequately modelled by PARAFAC components as they are not low-rank trilinear [5,6,12]. Because both these parts of the recordings are not connected to chemical information, the values in this range are normally set to missing, although this is also often suboptimal and associated with other kinds of modelling problems [5,12]. The pattern of the missing values within the array in the latter case is systematic and constant over the samples: entire tubes are missing across the sample mode (Fig. 1d). In the remaining part of this work, this pattern will be referred to as systematically missing spectra (SMS).

## 2. Algorithms

### 2.1. Alternating least squares with single imputation (ALS-SI)

#### 2.1.1. PARAFAC-ALS

The most common algorithm used to fit a PARAFAC model is based on the alternating least squares idea [13]: the nonlinear problem (4) is split into smaller, linear subproblems that are solved iteratively until convergence is established. Because all the steps are optimised in the least square sense, the loss function $L(\mathbf{A},\mathbf{B},\mathbf{C}) = \mathbf{X}\,\|^{(I\times JK)} - \mathbf{A}(\mathbf{C}\odot\mathbf{B})^{\mathrm{T}}\,\|_{\mathrm{F}}^{2}$ is bound not to increase at each step and tends to a (possibly local) minimum.

Given initial estimates for **B** and **C**, Eq. (4) becomes linear with respect to the matrix **A**, and an interim optimal least squares estimate of the latter can be computed as

$$\mathbf{A}^{(s)} = \mathbf{X}^{(I\times JK)}\left(\left(\mathbf{C}^{(s-1)}\odot\mathbf{B}^{(s-1)}\right)^{\mathrm{T}}\right)^{+}, \tag{5a}$$

where $s-1=0$ indicates the initial estimates for **B** and **C**, respectively, and + indicates the Moore–Penrose generalised inverse. The computation of $\mathbf{A}^{(s)}$ on the basis of $\mathbf{B}^{(s-1)}$ and $\mathbf{C}^{(s-1)}$, where $s$ indicates the iteration number, is followed by analogous substeps determining $\mathbf{B}^{(s)}$ and $\mathbf{C}^{(s)}$:

$$\mathbf{B}^{(s)} = \mathbf{X}^{(J\times IK)}\left(\left(\mathbf{C}^{(s-1)}\odot\mathbf{A}^{(s)}\right)^{\mathrm{T}}\right)^{+} \tag{5b}$$

$$\mathbf{C}^{(s)} = \mathbf{X}^{(K\times IJ)}\left(\left(\mathbf{B}^{(s)}\odot\mathbf{A}^{(s)}\right)^{\mathrm{T}}\right)^{+} \tag{5c}$$

After Eq. (5c) the convergence is checked: if the value of $L(\mathbf{A},\mathbf{B},\mathbf{C})$ has decreased in relative terms less than a chosen small positive number (the convergence criterion), the algorithm is stopped; otherwise, it continues estimating $\mathbf{A}^{(s+1)}$ for fixed $\mathbf{B}^{(s)}$ and $\mathbf{C}^{(s)}$ (i.e., the next iteration step).

The Khatri–Rao product has a property that allows significant savings in the calculations. Specifically:

$$(\mathbf{B}\odot\mathbf{A})^{\mathrm{T}}(\mathbf{B}\odot\mathbf{A}) = \mathbf{B}^{\mathrm{T}}\mathbf{B}*\mathbf{A}^{\mathrm{T}}\mathbf{A} \tag{6}$$

where * is the Hadamard (element-wise) product. Following the fact that $\mathbf{M}^{+}=(\mathbf{M}^{\mathrm{T}}\mathbf{M})^{+}\,\mathbf{M}^{\mathrm{T}}$ [8], Eq. (5a) is solved as

$$\mathbf{A} = \mathbf{X}^{(I\times JK)}(\mathbf{C}\odot\mathbf{B})(\mathbf{B}^{\mathrm{T}}\mathbf{B}*\mathbf{C}^{\mathrm{T}}\mathbf{C})^{+}, \tag{7}$$

where the indices relative to the iterations are skipped for clarity. The ALS algorithm has only linear convergence and slows down even further when it encounters so-called swamps, i.e., regions where two or more factors grow increasingly collinear and arbitrarily large maintaining opposite sign while the loss function decreases very slowly [14,15]. In order to accelerate the convergence, several strategies have been devised [7]. One that proved efficient in many cases uses a so-called line-search procedure [7,13], which is based on the observation that the ALS algorithm, particularly when stuck in a swamp, often proceeds with increasingly shorter steps in very collinear directions for several consecutive iterations. The line-search acceleration tests, every given number of iterations, if a longer step along the latest computed update for the loading matrices leads to a larger decrease of the loss function [7].

#### 2.1.2. Handling missing data

The ALS algorithm, as described in the previous section, cannot handle missing values and requires some modifications to operate in the presence of incomplete observations. One method, which has been successfully employed with other multilinear models [1,3,4,7], is represented by single imputation.

In such procedure, Eqs. (5a)–(5c) are applied, instead of the original array $\underline{\mathbf{X}}$, to an array $\underline{\tilde{\mathbf{X}}}$ defined as

$$\underline{\tilde{\mathbf{X}}}^{(s)} = \underline{\mathbf{X}} * \underline{\mathbf{M}} + \underline{\mathbf{Y}}^{(s)} * \left(\underline{\mathbf{1}} - \underline{\mathbf{M}}\right) \tag{8}$$

where $\underline{\mathbf{Y}}^{(s)}$ is the interim model computed at the $s$-th iteration, and $\underline{\mathbf{1}}$ is an array of ones having the same dimensions of $\underline{\mathbf{X}}$. $\underline{\mathbf{M}}$ is an array whose elements are defined as

$$m_{ijk} = \begin{cases} 0 & \text{if } x_{ijk} \text{ is missing} \\ 1 & \text{if } x_{ijk} \text{ is not missing} \end{cases} \tag{9}$$

$\underline{\tilde{\mathbf{X}}}$ contains no missing values and thus allows the use of the standard PARAFAC-ALS algorithm to estimate the model parameters. $\underline{\tilde{\mathbf{X}}}^{(s)}$ is updated at every iteration on the base of Eq. (3). The zero-iteration approximation $\underline{\mathbf{Y}}^{(0)}$ is reckoned depending on the pattern of the missing values. In general, it is taken as the average of the observed values in the corresponding columns/tubes or of the whole array.

The single imputation algorithm, under the conditions of normality (with zero mean and identical variance) and independence of the residuals, falls into the category of the Expectation Maximisation (EM) approach for incomplete data sets. The EM method was devised in the maximum-likelihood framework [16] and is divided in two steps: the E-step and the M-step. In the E-step, the conditional expectation of the likelihood function is computed given the observed data and the current estimated parameters. In least squares terms, this corresponds to calculating the loss function with respect to Eq. (8), i.e.,

$$\begin{aligned} L(\mathbf{A}^{(s)},\mathbf{B}^{(s)},\mathbf{C}^{(s)}) &= \|\tilde{\mathbf{X}}^{(s)} - \mathbf{Y}^{(s)}\|_{\mathrm{F}}^{2} \\ &= \|\tilde{\mathbf{X}}^{(s)} - \mathbf{A}^{(s)}(\mathbf{C}^{(s)}\odot\mathbf{B}^{(s)})^{\mathrm{T}}\|_{\mathrm{F}}^{2} \end{aligned} \tag{10}$$

where the superscript relative to the unfolding has been skipped for clarity. The loss function (Eq. (10)) represents the expected value of the log-likelihood function (with changed sign) given the above assumptions on the residuals.

The M-step determines new estimates for the parameters maximising the likelihood function. This step is simply represented by Eqs. (5a)–(5c) and the computation of the corresponding $\underline{\mathbf{Y}}$.

It has been demonstrated that Eq. (10) is bound not to increase and that the convergence of the procedure is linear with a convergence rate that is related to the amount of missing information [17,18].

This suggests that the already relatively slow convergence rate of ALS may be further reduced by an increased amount of missing values. Furthermore, whereas upon final convergence the estimates for the missing values have no influence on the estimated parameters, a large amount of missing elements may increase the risk of convergence to a local minimum as the interim model would describe for the largest part the (erroneous) imputed values. These two aspects pose the premises for the use of the modified Levenberg–Marquadt algorithm described in the following section.

### 2.2. Incomplete data PARAFAC (INDAFAC)

#### 2.2.1. PARAFAC-LM

The Levenberg–Marquadt method is a modification of the Gauss–Newton (iterative) algorithm for solving nonlinear least squares problems [19,20] and has been proposed for solving problem (4) in several instances [11,21,22]. In order to describe this method, it is necessary to introduce a vectorised notation for the PARAFAC model:

$$\mathbf{x} = \mathrm{vec}\mathbf{X}^{(I\times JK)} = \mathrm{vec}\left[\mathbf{A}(\mathbf{C}\odot\mathbf{B})^{\mathrm{T}}\right] + \mathrm{vec}\mathbf{R}^{(I\times JK)} \tag{11}$$

If one defines a vector $\mathbf{p}=\mathrm{vec}[\mathbf{A}^{\mathrm{T}}|\mathbf{B}^{\mathrm{T}}|\mathbf{C}^{\mathrm{T}}]$ holding the model parameters, problem (4) can be expressed as

$$\arg\min_{\mathbf{p}} \| \mathbf{r}(\mathbf{p}) \|_2^2 = \arg\min_{\mathbf{p}} (\mathbf{x}-\mathbf{y}(\mathbf{p}))^{\mathrm{T}}(\mathbf{x}-\mathbf{y}(\mathbf{p})) \tag{12}$$

where $\mathbf{r}=\mathrm{vec}\mathbf{R}^{(I\times JK)}$ and $\mathbf{y}=\mathrm{vec}[\mathbf{A}(\mathbf{C}\odot\mathbf{B})^{\mathrm{T}}]$.

In the Gauss–Newton algorithm (and the Levenberg–Marquadt modification), an update $\Delta\mathbf{p}$ for all the parameters is computed at each iteration, and the new estimates for the model parameters are defined as $\mathbf{p}^{(s)}=\mathbf{p}^{(s-1)}+\Delta\mathbf{p}$. This method is based on a Taylor expansion of the residuals with respect to the interim parameters $\mathbf{p}^{(s)}$:

$$\mathbf{r}\left(\mathbf{p}^{(s)}+\Delta\mathbf{p}\right) = \mathbf{r}\left(\mathbf{p}^{(s)}\right) + \mathbf{J}\Delta\mathbf{p} + O\left(\|\Delta\mathbf{p}\|_2^2\right) \tag{13}$$

where $\mathbf{J}$ is the Jacobian matrix of $\mathbf{r}(\mathbf{p})$, i.e., an $IJK\times(I+J+K)F$ matrix whose elements are defined as

$$j_{mn} = \frac{\partial r_m}{\partial p_n} = -\frac{\partial y_m}{\partial p_n} \tag{14}$$

If one ignores the error term $O(\|\Delta\mathbf{p}\|_2^2)$ in Eq. (13), the update $\Delta\mathbf{p}$ for all the parameters can be computed as the solution to the linear least squares problem [19]:

$$\arg\min_{\Delta\mathbf{p}} \| \mathbf{r}\left(\mathbf{p}^{(s)}\right) + \mathbf{J}\left(\mathbf{p}^{(s)}\right)\Delta\mathbf{p} \|_2^2 \tag{15}$$

There are several methods for solving Eq. (15). The one employed here is based on the system of normal equations:

$$\mathbf{J}^{\mathrm{T}}\mathbf{J}\Delta\mathbf{p} = \mathbf{J}^{\mathrm{T}}\mathbf{r} \tag{16}$$

which is solved by means of a Cholesky decomposition and back-substitution. The choice is justified by the sparsity of the Jacobian and by its dimensions [11]. Because of its computational complexity (each update requires approximately $O(N^3)$ operations, where $N$ is the number of parameters), this solution is suited for small- and medium-size problems. Iterative methods, such as Preconditioned Conjugate Gradients, may be more efficient for large-scale problems [22].

The Gauss–Newton algorithm described thus far is particularly appealing, because it guarantees quadratic convergence provided that the initial estimates for the parameters are close enough to the solution and that the residuals at the solution are not too large [19,20]. On the other hand, if these conditions are not fulfilled, the algorithm may not converge at all. Furthermore, the method requires modifications if the Jacobian is rank-deficient [19], as it is the case when fitting a PARAFAC model: due to the scaling indeterminacy intrinsic to this model, $2F$ (for a three-way array) of the Jacobian singular values are zeros to machine precision [21,22].

The Levenberg–Marquadt modification (LM) of the Gauss–Newton algorithm copes with all these problems, thus yielding a globally convergent algorithm [19,20]. In the LM algorithm, the system of normal Eq. (16) is modified to

$$\left(\mathbf{J}^{\mathrm{T}}\mathbf{J} + \lambda\mathbf{I}_{(I+J+K)F}\right)\Delta\mathbf{p} = \mathbf{J}^{\mathrm{T}}\mathbf{r} \tag{17}$$

The algorithm belongs to the category of the trust region methods. In essence, a "trust region", which radius is a function of $\lambda$, is a sphere centred in the current estimate $\mathbf{p}^{(s)}$ where the linear approximation for the residuals is assumed to hold. The update $\Delta\mathbf{p}$ is computed so that it minimises the residuals inside this region. If $\Delta\mathbf{p}$ leads to an insufficient decrease of the loss function, the update is rejected, the trust region is shrunk (i.e., $\lambda$ is increased), and a new update is calculated. Various strategies exist to define whether the update should be accepted or rejected and how to update $\lambda$. The one used here is described in detail in Ref. [20] and is based on the ratio between the linearly predicted decrease of the loss function $L(\mathbf{p})-\|\mathbf{r}(\mathbf{p})+\mathbf{J}(\mathbf{p})\Delta\mathbf{p}\|_2^2$ and the actual decrease after the update $L(\mathbf{p})-L(\mathbf{p}+\Delta\mathbf{p})$.

The scaling indeterminacy poses another problem related to the numerical stability of the algorithm. If the standard scaling convention for the loading matrices is used (i.e., $\|\mathbf{b}_f\|_2=\|\mathbf{c}_f\|_2=1$), the "practical" condition number of the Jacobian (i.e., computed disregarding the scaling indeterminacy—see Section 3.2) may become exceedingly large (typically because $a_{if}b_{jf}\simeq a_{if}c_{kf}\gg b_{jf}c_{kf}$). This can be avoided by setting the norm of the three loading vectors of the same component to be the same and equal to $q_f=(\|\mathbf{a}_f\|_2\|\mathbf{b}_f\|_2\|\mathbf{c}_f\|_2)^{1/3}$ [11].

*2.2.2. Handling missing values (INDAFAC)*

This PARAFAC-LM algorithm is significantly more memory demanding than ALS and more expensive per iteration, both in number of operations and computational time [11,21]. Apart from the greater stability with respect to the collinearity and overfactoring, the LM algorithm can be readily modified to treat the case of incomplete data without any imputation. If the loss function is transformed into

$$L(\mathbf{p}) = \| \mathbf{r}^* \text{ vec}\mathbf{M}^{(I\times JK)} \|_2^2, \tag{18}$$

where $\underline{\mathbf{M}}$ is defined as in Eq. (9), the rows in the Jacobian corresponding to the missing observations can be eliminated as the residuals (and thus the loss function) do not change with respect to these elements. This has several advantages: the number of non-zero elements in the Jacobian drops from $3FIJK$ to $3pFIJK$, where $0<p\leq 1$ is the fraction of non-missing values in the array, thus reducing the memory consumption; furthermore, as $\mathbf{J}$ is extremely sparse, the computation of the products $\mathbf{J}^T\mathbf{J}$ and $\mathbf{J}^T\mathbf{r}$ becomes less and less expensive with the increase of the fraction of missing values.

Under the assumptions of normality with mean zero and identical variance of the residuals, this method of handling the missing values is an analogue to the modified Newton method using the empirical information matrix for maximum likelihood estimation [18].

## 3. Experimental

Numerous aspects can affect the quality of the estimated PARAFAC model [7]. The aim of this work is to study the behaviour of the two proposed algorithms in the presence of large amounts of missing values with different patterns. Only a few additional properties of the data have been considered in the simulations in order to simplify the setup of the experiments.

The experimental part was conducted in two different stages. The first comprised a Monte Carlo study on synthetic data sets. In the second, the presence of missing values was simulated in three fluorescence data sets of different compositions and degrees of collinearity.

The correct rank of the model was assumed known in all cases, the study of the effect of overfactoring in combination with the presence of missing values is left for future research.

Both algorithms were initialised using the same best fitting of 10 runs of ALS-SI limited to 10 iterations and started with loading matrices of random values. Both algorithms were stopped when the relative decrease in the value of the loss function $(L^{(s)}-L^{(s-1)})/L^{(s-1)}$ was less than $10^{-6}$ or a predetermined number of iterations was reached (10 000 for ALS-SI and 1000 for INDAFAC). For INDAFAC, a second convergence criterion was set at $10^{-8}$ for the infinite norm of the gradient vector [20].

All the tests were run on a Pentium IV® 2.6-GHz computer with 512 MB memory, working under Windows XP. All the computations were run in MATLAB 6.5 (The Mathworks, Natick, MA, USA). Data sets I and II, the functions for generating the simulated sets and for the PARAFAC-LM algorithms, are available for download at the authors' group webpage (http:www.models.kvl.dk, June. 2004). The functions for PARAFAC-ALS with single imputation are part of the N-way toolbox (downloadable at http:www.models.kvl.dk, June. 2004).

### 3.1. Simulated data sets

The Monte Carlo study has been carried out on the basis of 2400 arrays generated considering the following aspects: rank of the array, degree of collinearity of the underlying components, amount of noise, percentage and pattern of missing values. The different conditions are summarised in Table 1. For each setup, 20 replicates were computed to account for minor statistical fluctuations. The dimension of the data sets was 30×30×30.

The data sets were generated on the basis of Eq. (3). In order to control the collinearity between the underlying components, the loading matrices were generated using the following equation ([11]; here reported with respect to the first mode):

$$\mathbf{A} = \mathbf{VL} \tag{19}$$

where $\mathbf{V}$ is a column-wise orthonormal $I\times F$ matrix, and $\mathbf{L}$ is the Cholesky factor of a square $F\times F$ matrix $\mathbf{U}$ holding ones on the diagonal and the required cosine of the angle between the loading vectors[1] in the off-diagonal elements [23]. Consequently, $\mathbf{A}^T\mathbf{A}=\mathbf{L}^T\mathbf{V}^T\mathbf{VL}=\mathbf{L}^T\mathbf{L}=\mathbf{U}$. All the factors were given the same magnitude.

The independent and homoscedastic noise was normally distributed with mean 0. Two desired levels of noise were attained using the following formula [11]:

$$\mathbf{R}^{(I\times JK)} = \frac{\text{Noise\%}}{100-\text{Noise\%}} \|\mathbf{X}^{(I\times JK)}\|_F \tilde{\mathbf{R}}^{(I\times JK)} \tag{20}$$

where $\tilde{\mathbf{R}}^{(I\times JK)}$ is a matrix of normally distributed (mean 0) random values having a Frobenius norm of 1. Noise% indicates the percentage of noise over the total variation in the array $\underline{\mathbf{X}}+\underline{\mathbf{R}}$.

The pattern for the missing values in the array in the RMV case was determined using the first $pIJK$ elements of a random permutation of the integers on the interval $[1,IJK]$. In the RMS case, the tubes (i.e., spectra) were removed in the third mode (Fig. 1b), and the position of the missing tubes was determined using the first $pIJ$ elements of a random permutation of the integers on the interval $[1,IJ]$. In both cases, it was checked that no slab contained only missing

[1] The cosine of the angle between two vectors $\mathbf{a}$ and $\mathbf{b}$ (also referred to as congruence) is defined as: $u(\mathbf{a},\mathbf{b}) = \cos(\widehat{\mathbf{a},\mathbf{b}}) = \mathbf{a}^T\mathbf{b}/(\|\mathbf{a}\|\|\mathbf{b}\|)$.

Table 1
Design factors and levels in the Monte Carlo study

| Factors | Levels |
|---|---|
| Percentage of missing values | 30, 40, 50, 60, 70 |
| Pattern of missing values | RMV[a], RMS[b], SMS[c] |
| Congruence[d] | 0.5, 0.9 |
| Noise[e] | 0.5, 2 |
| Model rank | 3, 4 |

[a] Randomly missing values.
[b] Randomly missing spectra.
[c] Systematically missing spectra.
[d] Cosine of the angle between the components in the $IJK\times1$ space.
[e] Expressed as a percentage of the total variation (i.e., of $\| \mathrm{vec}\mathbf{X}^{(I\times JK)} \|_2$).

values. In the SMS case, the missing values for each sample were set in two identical triangles at two opposite vertices of each horizontal slab (as in Fig. 1d), their size was defined so that the required fraction of missing was best approximated.

### 3.2. Real data sets

The two algorithms were tested on three different data sets of fluorescence measurements:

(1) Twenty-two solutions of four substances (DOPA, hydroquinone, tryptophan, and phenylalanine) were analysed on a Perking-Elmer LS50 spectrofluorometer [24]. The 13 excitation wavelengths ranged between 245 and 305 nm with steps of 5 nm, whereas in emission the range comprised 131 wavelengths measured between 260 and 390 nm with a step of 1 nm. Three "artificial" data sets were generated out of the single measured one by selecting every third variable in the emission mode in each replicate set [11]. Thus, replicate set one used emission variable number 1, 4, 7, etc., and replicate set two used variable 2, 5, 8, etc. The procedure thus yielded two arrays of size $22\times87\times13$ and one of size $22\times88\times13$. The Rayleigh scatter was removed by subtracting from each sample a "model" of the scatter. The Raman scatter was not treated.

(2) Fifteen solutions of DOPA, hydroquinone, tyrosine, and tryptophan were analysed by means of a Cary EClipse spectrofluorometer. The excitation mode comprised wavelengths between 230 and 300 nm measured at intervals of 5 nm (15 variables). In emission, the wavelengths varied between 282 and 412 nm with 2-nm steps (66 variables). Full factorial design with two concentration levels per constituent was employed, and six instrumental replicates were measured, thus generating six arrays of size $15\times66\times15$. The influence of Rayleigh and Raman scatter was minimized by subtracting a blank.

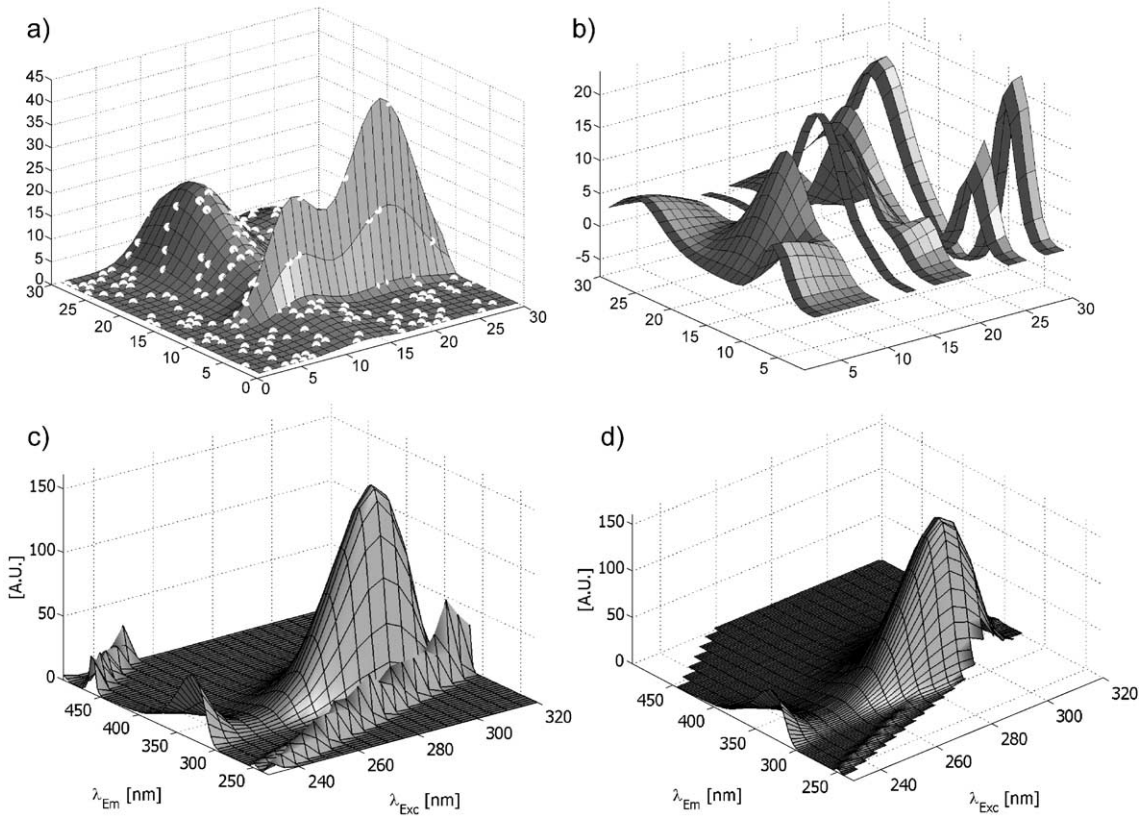


Fig. 1. Patterns of missing values on a single horizontal slab: (a) randomly missing values (RMV); (b) randomly missing spectra (RMS); (c and d) EEM fluorescence landscape (c) and the corresponding systematically missing spectra (SMS) pattern after the Rayleigh scatter removal (d).

(3) Forty-seven solutions of five compounds (cathecol, hydroquinone, indole, tryptophan ,and tyrosine) were measured with a Varian Cary EClipse spectrofluorometer. The emission ranged from 230 to 500 nm with intervals of approximately 2 nm, while the excitation varied between 230 and 305 nm with 5-nm steps [25]. The Rayleigh scatter on the original data set was removed by setting the corresponding elements in each sample to missing [6,7]. The Raman scatter was not treated. The data set was further reduced to a size of 47×80×16 by selecting the emission wavelengths between 276 and 434 nm in order to remove the largest part of the missing values. A small part (5%), though, remained at the low-emission/high-excitation wavelengths in a region that did not interfere with the resolution of the constituents.

Five replicates with random patterns of missing values (i.e., for RMV and RMS) were generated to account for minor statistical fluctuations. For SMS, five runs were also tested using different starting values but the same pattern.

Preliminary tests showed that for some of the constituents of the real data sets, the predictions worsened already at 30% of missing values in the SMS pattern. Therefore, two more levels were added, and the real data sets were analysed with fractions of missing elements varying from 10% to 70% with increments of 10%.

Table 2 shows the degree of collinearity, the explained variation, and the core consistency [26] of the three data sets of the underlying components obtained with the complete data set. According to Kiers [27], on the basis of the condition number of the loading matrices, both data set I and II can be classified as mildly collinear and data set III as severely collinear. None of them however entirely falls into any category defined according to this criterion. The Jacobian matrix associated with the PARAFAC model contains more information on the numerical difficulty and collinearity of the problem. Due to its rank deficiency, the true condition number cannot be employed for diagnostic purposes. However, the number of numerically zero singular values related to the scaling indeterminacy is known beforehand to be $2F$; consequently, one can consider the "practical" condition number $\gamma_{\mathbf{J}}$ instead:

$$\gamma_{\mathbf{J}} = \frac{\sigma_1}{\sigma_{(I+J+K-2)F}}, \tag{21}$$

where $\sigma_1$ is the largest singular value of $\mathbf{J}$, and $\sigma_{(I+J+K-2)F}$ is the last non-zero singular value after having taken into account the scaling indeterminacy. As mentioned in Section 2.2.1, the scaling convention affects the value of $\gamma_{\mathbf{J}}$, therefore the loading vectors were scaled so that $\| \mathbf{a}_f \|_2 = \| \mathbf{b}_f \|_2 = \| \mathbf{c}_f \|_2$.

$\gamma_{\mathbf{J}}$ appears as more univocal than the condition numbers of the separate loading matrices when it comes to describing the degree of collinearity. Although no actual threshold can be given nor suggested for $\gamma_{\mathbf{J}}$, a ranking can be clearly observed between the three problems, were data set I is the least collinear, data set III is the most collinear, and data set II is in the middle. Furthermore, $\gamma_{\mathbf{J}}$ also helps in describing the effect of the missing values on the fitting procedure: $\gamma_{\mathbf{J}}$ increases systematically with the percentage of missing values. In particular, for the real data sets, with 70% missing elements, values of $\gamma_{\mathbf{J}}$ in the order of $10^6$ were observed upon final convergence.

It should be noted that the factors were extracted from the complete data sets and are thus affected by small non-linearities in the recorded signal that may show up as small interaction terms between the factors. Such phenomena can be theoretically described in terms of model error and might be "quantified" by the core consistency diagnostic [26]. It can clearly be seen that data set III is particularly problematic in this respect: the lower value of the core consistency reflects the presence of relatively unstable components (or of deviations from low-rank trilinearity) and thus may imply a more difficult problem than for data set I and II, where the PARAFAC model is clearly more adequate. The problem of model error is further complicated by the effect of certain patterns of missing values (see Appendix A).

Finally, it can be seen in Table 2 that, in spite of the fact that in one mode the angle between two factors may be small (with cosines up to 0.98), the whole components are in fact rather well separated (mostly as a

Table 2
Diagnostic parameters for the three real data sets: condition numbers for the Jacobian **J** and the three loading matrices **A**, **B**, and **C**; minimum and maximum congruence between factors for the three loading matrices and their Khatri–Rao products; core consistency.

| Data set | Condition number | | | | Congruence (min-max) | | | | | | | Core consistency (%)[b] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **J**[a] | **A** | **B** | **C** | **A** | **B** | **C** | **A⊙B** | **B⊙C** | **A⊙C** | **A⊙B⊙C** | |
| I | 18.43 | 2.83 | 20.4 | 5.96 | 0.31–0.49 | 0.002–0.88 | 0.15–0.94 | 0.00–0.38 | 0.05–0.37 | 0.00–0.32 | $4 \cdot 10^{-4}$–0.32 | 98.9 |
| II | 35.85 | 8.99 | 12.24 | 8.07 | 0.46–0.50 | 0.15–0.86 | 0.55–0.94 | 0.07–0.42 | 0.26–0.46 | 0.06–0.35 | 0.06–0.35 | 99.3 |
| III | 115.06 | 5.96 | 41.93 | 40.6 | 0.36–0.55 | 0.22–0.98 | 0.52–0.97 | 0.11–0.52 | 0.22–0.45 | 0.08–0.43 | 0.08–0.43 | 69.6 |

[a] Computed according to Eq. (21).
[b] Computed according to Ref. [26].

consequence of the design in the concentration levels). The same holds for the Khatri–Rao products $\mathbf{A}\odot\mathbf{B}$, $\mathbf{B}\odot\mathbf{C}$, and $\mathbf{A}\odot\mathbf{C}$, whose pseudoinverse is necessary for the computation of the interim solutions (Eqs. (5a)–(5c)). This suggests that the three real data problems in themselves are not particularly difficult provided that all the information is available.

For the real data sets, the position of the missing values was selected on the same basis as for the artificial ones. With specific reference to data set III, the missing values resulting from the removal of the Rayleigh scatter are completely covered by the artificially imposed ones in the SMS case. This was not entirely the case for the other two patterns, where the "naturally" missing values summed up with artificially set ones leading to a fraction of missing values slightly larger than the required value, but still reasonably close to it.

### 3.3. Criteria of interest

Two main aspects were considered in this work: the statistical quality of the retrieved solutions and the computational aspects of the two algorithms. Specifically, the diagnostics discussed in the following subsections were used.

#### 3.3.1. Recovery capability

One true factor ($\mathbf{z}_f=\mathbf{c}_f\otimes\mathbf{b}_f\otimes\mathbf{a}_f$) is considered as recovered if there is one component ($\hat{\mathbf{z}}_g=\hat{\mathbf{c}}_g\otimes\hat{\mathbf{b}}_g\otimes\hat{\mathbf{a}}_g$) in the fitted solution having a congruence with it greater than a certain threshold. If there are no extreme baselines, a threshold for the single loading vector that guarantees recovery is 0.99; correspondingly, the criterion for the component of three loading vectors may be set at $0.99^3 \simeq 0.97$. The two criteria are not entirely the same; in fact, with the latter, it is possible that a component is considered as recovered also when one of the corresponding loading vectors has a congruence lower than 0.99. In practice, the results using a threshold for the whole component of 0.97 give slightly more optimistic results, but in general the interpretation does not change much. While for the Monte Carlo simulation, the true components are known, for the real data sets the factors found from fitting the model to the original array (i.e., without artificially set missing values) were taken as the correct underlying ones, although this is clearly only an approximation.

The underlying model can be considered as fully recovered (retrieved) if all its factors have been recovered according to a threshold of 0.97. The recovery capability is the percentage full retrievals over the total number of computed models [11].

Because the factor order (permutation) in the solution is not uniquely defined [7,13], all possible permutations of the extracted factors need to be compared with the underlying components to establish full recovery. The correct permutation is defined as the one yielding the highest sum of the cosines with the "real" one [15,28]. In other words, the "winning" permutation $\mathbf{P}_{\text{win}}$ is found as a solution to

$$\mathbf{P}_{\text{win}} = \arg\max_{\mathbf{P}} \operatorname{tr}\left(\left(\mathbf{A}^{\mathrm{T}}\hat{\mathbf{A}}\mathbf{P}\right)*\left(\mathbf{B}^{\mathrm{T}}\hat{\mathbf{B}}\mathbf{P}\right)*\left(\mathbf{C}^{\mathrm{T}}\hat{\mathbf{C}}\mathbf{P}\right)\right) \tag{22}$$

where $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are the real (column-wise normalised) loading matrices, $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, and $\hat{\mathbf{C}}$ are the estimated (column-wise normalised) loading matrices, $\mathbf{P}$ are all the possible permutation matrices for $F$ columns and tr($\mathbf{M}$) indicates the trace of the square matrix $\mathbf{M}$.

#### 3.3.2. Congruence product

The quality of the solution was also assessed on the basis of the product of the congruences for the whole components or the single loading matrices:

$$\phi_{\mathrm{z}} = \prod_{f=1\ldots F} u\left(\hat{\mathbf{z}}_f, \mathbf{z}_f\right), \tag{23}$$

#### 3.3.3. Mean-squared error

The value of the Mean-Squared Error (MSE) for the model parameters. With respect to $\mathbf{A}$, the MSE is computed as

$$\mathrm{MSE}(\mathbf{A}, \hat{\mathbf{A}}, \mathbf{P}_{\text{win}}, \mathbf{S}_{\mathbf{A}}) = \frac{\| \mathbf{A} - \hat{\mathbf{A}}\mathbf{P}_{\text{win}}\mathbf{S}_{\mathbf{A}} \|_{\mathrm{F}}}{IF} \tag{24}$$

where $\mathbf{S}_{\mathbf{A}}$ is a scaling matrix found as the solution to

$$\arg\min_{\mathbf{S}_{\mathbf{A}},\mathbf{S}_{\mathbf{B}},\mathbf{S}_{\mathbf{C}}} \left( \| \mathbf{A} - \hat{\mathbf{A}}\mathbf{P}_{\text{win}}\mathbf{S}_{\mathbf{A}} \|_{\mathrm{F}} + \| \mathbf{B} - \hat{\mathbf{B}}\mathbf{P}_{\text{win}}\mathbf{S}_{\mathbf{B}} \|_{\mathrm{F}} + \| \mathbf{C} - \hat{\mathbf{C}}\mathbf{P}_{\text{win}}\mathbf{S}_{\mathbf{C}} \|_{\mathrm{F}} \right) \text{ subject to } \mathbf{S}_{\mathbf{A}}\mathbf{S}_{\mathbf{B}}\mathbf{S}_{\mathbf{C}} = \mathbf{I}_F \tag{25}$$

Such a procedure is necessary because trivial differences in scaling may yield unnecessarily high values for the MSE [11,29].

#### 3.3.4. Loss function value

The value of the loss function is important to establish the capability of the two different algorithms to reach a (global) minimum.

#### 3.3.5. Error in calibration

The presence of the concentration matrices for the three real data sets allows the use of one additional quality diagnostic: the Root-Mean-Squared Error in Calibration (RMSEC) in a linear regression model based on the loadings in the first mode. Only the scores of the component associated to the sought constituent are used in addition to an intercept. Thus, for the $f$-th constituent:

$$\mathrm{RMSEC}_f = \sqrt{\frac{\| \mathbf{y}_f - \hat{\mathbf{y}}_f \|_2^2}{I}} \tag{26}$$

where $\hat{\mathbf{y}}_f = [\mathbf{a}_f \quad 1]\left([\mathbf{a}_f \quad 1]\right)^{+}\mathbf{y}_f$.

### 3.3.6. Numerical assessments

The efficiency of the two algorithms is assessed in terms of time consumption and number of iterations necessary to reach convergence. Especially with respect to the latter, it is well known that ALS methods require many more (less expensive) iterations [11,21], thus a direct comparison is not feasible. On the other hand, the number of iterations may be of value, in relative terms, to assess the effect of a certain feature on the convergence of the same algorithm.

## 4. Results and discussion

### 4.1. Simulated data sets

The first aspect that was considered was the capability of full recovery for the two tested algorithms. In this respect, INDAFAC performed slightly better, managing to retrieve the true underlying components for 77.8% of all the synthetic arrays compared to the 77.3% of ALS-SI.

The feature in the designed sets that affected the most the number of complete recoveries is the pattern of the missing values Table 3. As expected, the RMV case is the easiest to be dealt with, and, apart from a small number of cases, the correct factors are recovered in all the replicates by both algorithms. The RMS proved to be somewhat more difficult to solve, and in a minor fraction of cases full recovery was not attained. The SMS pattern appeared to be much more problematic, with an occurrence of full recoveries that is not comparable with the other two.

The reasons for the lower recoveries of RMS and SMS must be sought both in the convergence to local minima and the presence of artefacts related to slabs with a large fraction of missing values (Fig. 2). In the former case, it was sufficient to restart the algorithm a number of times to yield the correct solutions, whereas in the latter, restarting did not accelerate convergence and yielded solutions with artefacts in the same positions. In these cases, not even initialising the algorithms using the real underlying factors prevented the emergence of artefacts in the solution, which was associated to a lower value of the loss function. In fact, the artefacts of the type shown in Fig. 2 are only indirectly a function of the fraction of missing values in the corresponding slab; they are determined by the fact that the few values that remain in a slab do not contain enough information with respect to the sought components. Occurrence and magnitude of the artefacts are not easy to predict as they are affected by the different sources of variation (including the non-trilinear ones such as scatter or noise [6]), as well as interactions between factors during convergence allowed for by the

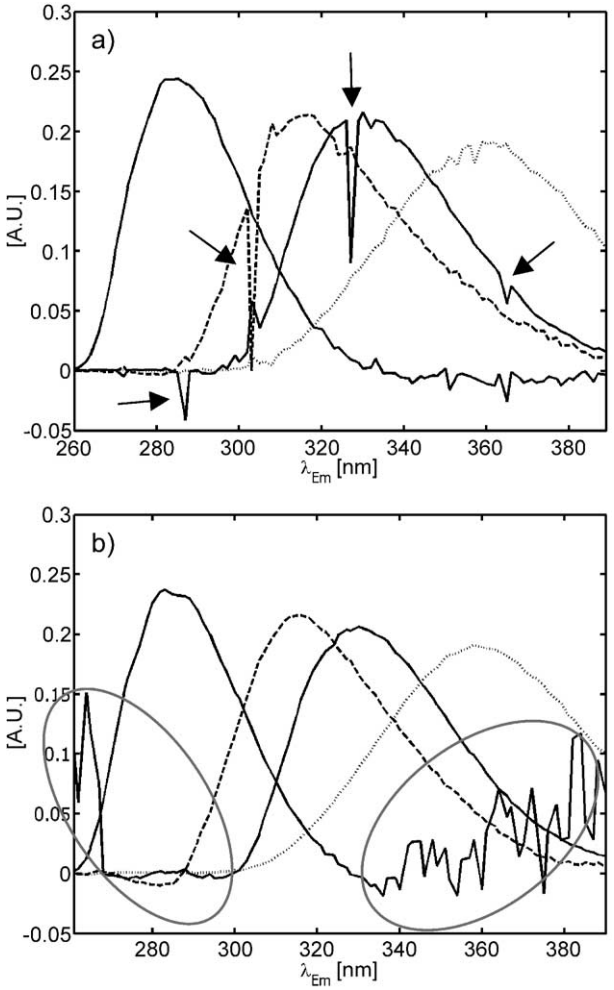


Fig. 2. Different kinds of artefacts in the emission loadings of data set I associated to the RMS (a) and SMS (b) missing values patterns.

Table 3
Percentage of full recoveries according to a threshold of 0.97 for the simulated data sets with respect to the separate design factors

| Pattern | Algorithm | Rank | | Congruence | | Missing values (%) | | | | | Noise (%) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 3 | 4 | 0.5 | 0.9 | 30 | 40 | 50 | 60 | 70 | 0.5 | 2.0 |
| RMV[a] | ALS-SI | 100.0 | 99.3 | 100.0 | 99.3 | 100.0 | 100.0 | 100.0 | 99.4 | 98.8 | 100.0 | 99.3 |
| | INDAFAC | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| RMS[b] | ALS-SI | 90.5 | 88.5 | 97.3 | 81.8 | 96.9 | 94.4 | 93.8 | 88.8 | 73.8 | 90.8 | 88.3 |
| | INDAFAC | 91.0 | 89.0 | 97.3 | 82.8 | 97.5 | 91.9 | 94.4 | 90.0 | 76.3 | 91.0 | 89.0 |
| SMS[c] | ALS-SI | 41.5 | 44.0 | 59.5 | 26.0 | 78.8 | 67.5 | 38.1 | 25.0 | 4.4 | 46.5 | 39.0 |
| | INDAFAC | 46.5 | 46.3 | 59.3 | 33.5 | 82.5 | 71.3 | 46.3 | 25.0 | 6.9 | 50.3 | 42.5 |

[a] Randomly missing values.
[b] Randomly missing spectra.
[c] Systematically missing spectra.

pattern of missing values (see Appendix A). It is beneficial to both speed of convergence and quality of the model to remove whenever possible the problematic slabs, but at the current stage there are few, if any, tools that allow their identification [4,30]. Their development exceeds the purposes of the work, and further studies will be necessary in this direction.

With respect to the other factors in the design, congruence has the most visible effects, along with the fraction of missing values. The Levenberg–Marquadt algorithm performs in general better for highly collinear factors [31], and this property is retained in the presence of missing values (Table 3); the INDAFAC algorithm yields (apart from one single setting) the correct solution more often than ALS-SI. It is once again evident how the RMV case creates very few problems to either of the algorithms, while the other two patterns yield a consistently decreasing number of fully retrieved models. Nevertheless, in the RMS case, even with 70% missing values, the loadings are correctly estimated in more than three cases out of four (for INDAFAC). In the SMS pattern, these percentages decrease to the point that in less than 50% of the cases overall the correct factors are recovered and full recovery hardly ever occurs for 70% missing values. Although, a slight worsening in the quality of the solution could be observed for both MSE (Fig. 3) and $\phi$ (not shown) as a result of an increase of rank from 3 to 4, this was too small to significantly affect the recovery capability (Table 3). Analogous observations could be made in the RMS case for the noise level, although this factor affects the quality of the solution more than the rank (Fig. 3). For the SMS pattern, the recovery capability is affected also by noise, which most likely influences magnitude and occurrence of artefacts [6].

All these results were confirmed by ANOVA models applied separately to the missing values patterns and having the number of full recoveries over the 20 replicates as response variable. As a result of such ANOVA models, also the interaction between congruence of the underlying factors and fraction of missing values appeared to be significant for ALS-SI for both RMS ($p$<0.012) and SMS ($p$<0.005) patterns and for INDAFAC only in the RMS case ($p$<0.023). Such interaction is not unexpected and means that missing values affect more critically data sets with more collinear components.

If one looks at full recovery in the three modes separately (Table 4), it is apparent how, in the SMS case, the recovery of **A**, **B**, and **C** differs, and that **A** is correctly estimated more often than the other two loading matrices. This is particularly important for calibration purposes, where **A** is used to determine the concentrations. Also in the RMS case, there is an asymmetry in the retrieval of the various modes. Although this is not apparent in the recovery capability relative to this mode, it shows in the quality of the estimations: $\phi_{\mathbf{C}}$ is larger than $\phi_{\mathbf{A}}$ and $\phi_{\mathbf{B}}$ in approximately

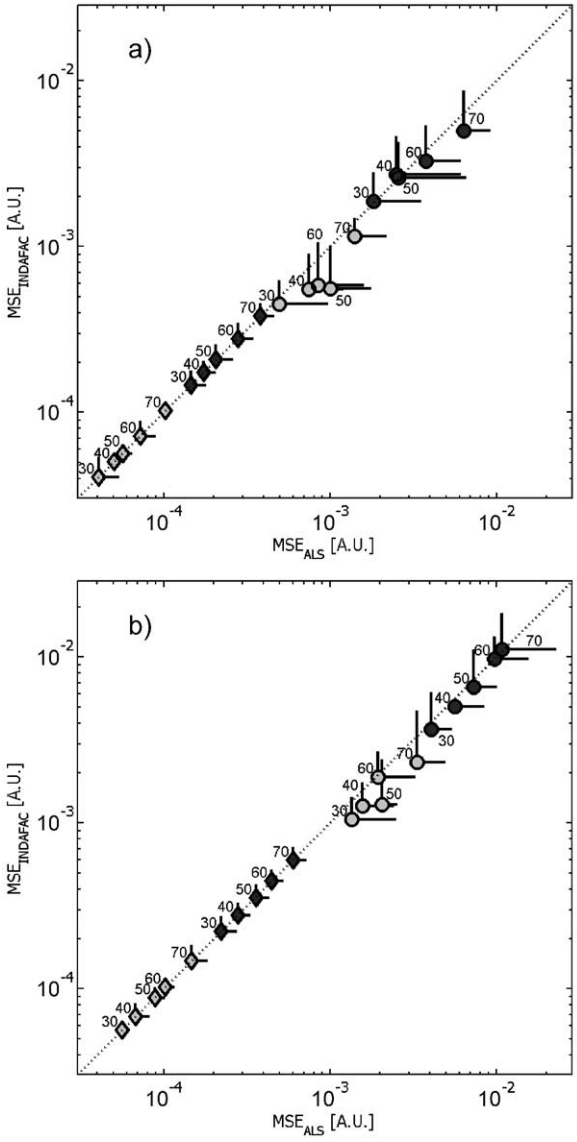


Fig. 3. Effect of various design factors on the median MSE for both algorithms for the RMV pattern: (a) rank 3, (b) rank 4; (◆) low congruence, (○) high congruence, (open symbol) low noise, (closed symbol) high noise. The numbers are the percentage of missing values. The lines departing from each symbol are the standard deviations for the MSE and the algorithm corresponding to the direction of the line.

60% of the cases, whereas in case of symmetry between modes (e.g., in the RMV pattern), this percentage is should be around 33%.

The different outcomes related to the RMS and SMS patterns can be in part explained if one considers the matricised form of the array $\underline{\mathbf{X}}$. In the RMS case, **C** spans the column space of $\mathbf{X}^{(K\times IJ)}$, which columns are formed of either completely missing elements or all real ones. Thus, with respect to **C**, complete information is always available, and the difficulties in retrieving the correct solution in this mode are associated only with how collinear are the columns of $\mathbf{B}\odot\mathbf{A}$ after the removal of the rows corresponding to the columns with missing values. Equivalently, for

Table 4
Mode recovery for the different patterns of missing values RMV, RMS, and SMS. All the factors and levels are considered. FR (fully recovered) indicates the percentage of full recoveries for the specified loading matrix according to a congruence threshold of 0.99. BR (best recovered) for loadings matrix **A** is the occurrence of $\phi_{\mathbf{A}}>\max(\phi_{\mathbf{B}},\phi_{\mathbf{C}})$ (the values for **B** and **C** are found mutatis mutandis). Symmetry between modes yields identical BR for the three loading matrices, i.e., approximately 33%

| Pattern | Algorithm | **A** | | **B** | | **C** | |
|---|---|---|---|---|---|---|---|
| | | FR (%) | BR (%) | FR (%) | BR (%) | FR (%) | BR (%) |
| RMV[a] | ALS-SI | 99.6 | 32.8 | 99.6 | 33.6 | 99.8 | 33.6 |
| | INDAFAC | 100.0 | 33.4 | 100.0 | 33.0 | 99.8 | 33.6 |
| RMS[b] | ALS-SI | 89.1 | 15.6 | 89.3 | 25.8 | 90.3 | 58.6 |
| | INDAFAC | 89.4 | 14.9 | 90.1 | 25.3 | 90.3 | 59.9 |
| SMS[c] | ALS-SI | 56.0 | 70.4 | 41.9 | 13.8 | 42.9 | 15.9 |
| | INDAFAC | 58.1 | 65.3 | 46.0 | 16.4 | 48.5 | 18.4 |

[a] Randomly missing values.
[b] Randomly missing spectra.
[c] Systematically missing spectra.

SMS, **A** spans the column space of $\mathbf{X}^{(I\times JK)}$, which again contains only completely full or completely missing columns. This explains the better performance in the first mode in the SMS case and links the difficulty of the problem to the collinearity of the columns of $\mathbf{C}\odot\mathbf{B}$ once the rows corresponding to the missing elements are removed. Yet, the much greater difficulty in SMS compared to RMS remains largely unexplained by these arguments. The small simulation illustrated Appendix A suggests that the SMS pattern may interfere with trilinearity by allowing for "interactions" between the two loading matrices **B** and **C**.

Figs. 3 and 4 show the effect of the various factors on the quality of the solution in terms of MSE. Fig. 3 describes the behaviour of the two algorithms when both converge to a meaningful solution. The plot is derived on the RMV pattern, but it applies also to the other two when both algorithms converge. The solution of the two algorithms is substantially identical for the low-congruence case (i.e., the symbols lie on the diagonal), although INDAFAC tends to yield a lower value of MSE. This is particularly evident in the high-congruence case. It can also be seen that the effect of rank is very limited compared to that of noise and particularly to that of congruence. These observations are consistent with what was observed on the base of the recovery capability. Note that a value of approximately $10^{-2}$ for the MSE appears as a good choice to establish full recovery and yields, apart from a limited number of cases, the same results as the aforementioned 0.97 threshold for the congruence. In this sense, convergence to local minima or solutions with large artefacts can be easily identified in Fig. 4. In particular, it can be seen that when INDAFAC does not converge, it yields significantly larger values of MSE (i.e., symbols lying over the diagonal in the plot) than those of ALS-SI in analogous conditions. This may be related to some sort of stabilising effect associated to the fact that (1) imputed values in ALS-SI are indirectly found through linear combinations of the given values, and (2) if one looks at the value of the loss function, INDAFAC clearly outperforms ALS-SI in finding a minimum; it attained the lowest value of the loss function in 97.25% of the cases. Only in a fraction of these, though, the discrepancy (in relative terms) was larger than 0.01%, namely, in 52.7% (7.3% if one considered a difference of more than 1%) of the 2400 data sets. In all the 2.75% of the cases when ALS-SI found a lower minimum, the difference was larger than 0.01% (2% with a 1% threshold). Thus, when both algorithms do not converge, the better MSE of ALS-SI is likely due to a lower capability of attaining a minimum (albeit a non-relevant one) of this algorithm. Had ALS-SI been able to determine the solution better, this would have been just as bad in terms of MSE as the one obtained with INDAFAC.

Table 5 shows the median number of iterations and of computational time with respect to the percentage and the pattern of the missing values. As expected, the number of iterations increases with the number of missing values and grows more rapidly for ALS-SI than for INDAFAC. It is also very relevant that the number of iterations increases more as a result of the pattern; as an average, the SMS pattern requires 20 times as many iterations as the RMV case for ALS-SI. The ratio for INDAFAC varies with the fraction of missing values, but is at most in the order of 8–10. This trend in the number of iterations hardly ever turns into an advantage in terms of time for the RMV pattern; only in 33% of the cases INDAFAC is faster at 60% missing elements and in 40% of the cases at 70%. For all the other levels of missing values, ALS-SI was faster. Vice versa, in the SMS case, INDAFAC is faster in about one-third of the cases (35%) already at 30% missing and in at least four out of five cases for 50% of missing elements or more.

### 4.2. Real data sets

Data set I turned out to be most simple to fit, in perfect accordance with the expectations based on $\gamma_{\mathbf{J}}$ and core consistency. Both algorithms recovered the correct components in all the replicate models up to 70% missing values for both the RMV and RMS pattern (not shown). The results also confirmed that SMS is the most challenging among the studied patterns. Artefacts similar to those of

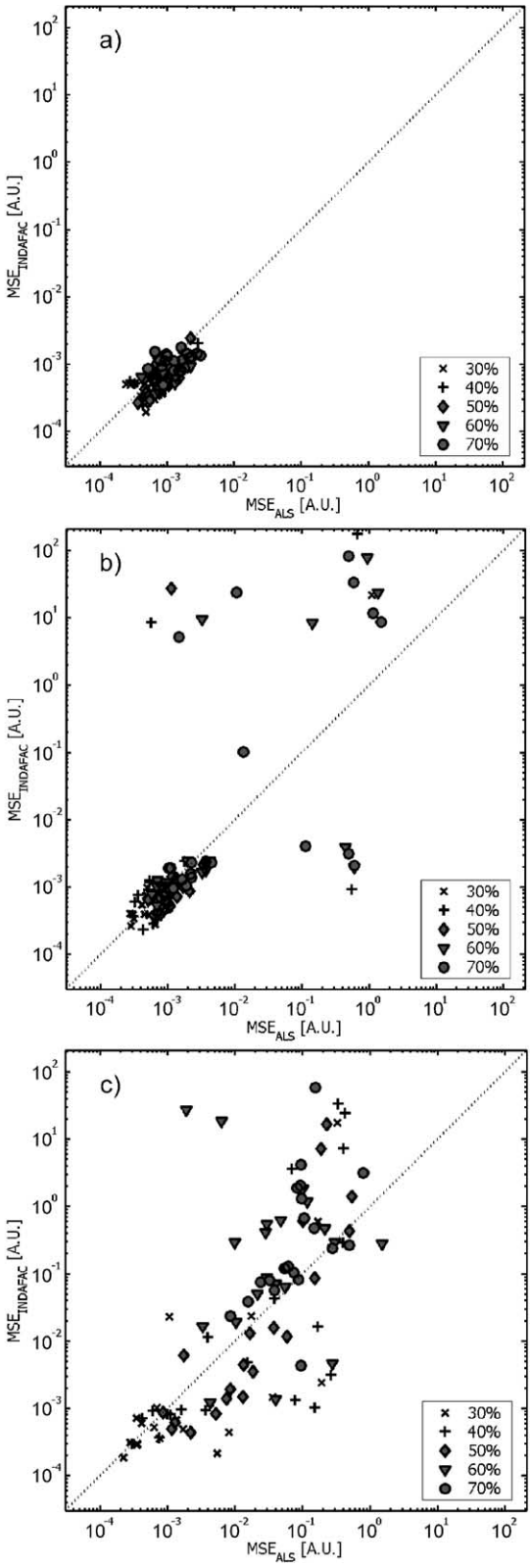


Fig. 4. MSE for the **A** matrix in the rank 3, high congruence and low-noise case. All replicates are displayed. When the MSE exceeds $10^{-2}$, the model can be considered as not converged to a meaningful solution. (a) RMV pattern, (b) RMS pattern, (c) SMS pattern.

Fig. 2 appeared in most of the cases, and both INDAFAC and ALS-SI were affected to the same extent. Nevertheless, **A**, which is the most relevant one for calibration purposes, was correctly recovered (again according to a threshold of 0.99 for each of the columns) in all the instances. Fig. 5 shows how the RMSEC varies with respect to the amount of missing elements for ALS-SI. The three studied patterns present remarkable differences. Whereas for both RMV and RMS, the effect of the missing values on the concentration estimates is very small, in the SMS case, there is an improvement in the predictions as the percentage of missing values increases for all constituents apart from phenylalanine. The reason for this appears to be the effect of the Raman scatter ([5,6]; i.e., the small ridge visible in Fig. 1d on the right hand of the main peak). Because of its position and magnitude, the Raman scatter is often left in the data and is given less attention than the Rayleigh one [25]. The setting of some elements to missing in the SMS pattern progressively cancels the Raman rather than removing it all at once from one level of missing to the following. The behaviour of the regression models then follows exactly the pattern described elsewhere [6] for the much more intense Rayleigh scatter; the predictions improve so long as more scatter, but not significant parts of the spectra, is removed. The degree of overlap of the single components determines whether the Raman removal will have an effect. E.g., phenylalanine lies on top of the Raman scatter ridge at the lowest excitation and emission wavelengths. Correspondingly, it is hardly affected by the removal of the Raman or by the setting of further missing values until 60% or 70% is reached. Contrariwise, e.g., tryptophan's main peak lies mostly off the Raman ridge, and its predictions benefit by the increase of the missing values. This observation is also consistent with the fact that both the RMV and RMS patterns have hardly any influence on the quality of the predictions. In theory, it might be that the removal of a certain part of the signal reduces the collinearity between the columns of product $\mathbf{C}\odot\mathbf{B}$ relative to these two constituents and the other constituents, but this does not seem to be the case here.

Fitting a PARAFAC model to data set II proved to be somewhat more difficult, once again in accordance with the considerations made in Section 3.2. Although the components were correctly recovered up to 70% missing values with RMV patterns, both algorithms failed to retrieve the underlying factors in 10–20% of the cases with the RMS pattern (not shown). The **A** matrix alone was correctly estimated at 70% of missing values only in 70% of the cases. The higher difficulty associated to fitting the PARAFAC model in presence of an SMS pattern is made apparent by the fact that full recovery is no longer guaranteed for the components when 20% of the elements are missing (for the **A** matrix, this happens at 50% of missing values). The variation of the RMSEC as a function of the percentage of missing values is more erratic than in

Table 5
Median number of iterations (# it.) and of computational time for the two algorithms for both simulated and real data sets

| Pattern | Data set | Algorithm | Percentage of missing values | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 10 | | 20 | | 30 | | 40 | | 50 | | 60 | | 70 | |
| | | | # It. | Time (s) | # It. | Time (s) | # It. | Time (s) | # It. | Time (s) | # It. | Time (s) | # It. | Time (s) | # It. | Time (s) |
| RMV[a] | Monte Carlo | ALS-SI | – | – | – | – | 23 | 0.5 | 23 | 0.5 | 32 | 0.7 | 37 | 0.9 | 52 | 1.2 |
| | | INDAFAC | – | – | – | – | 9 | 3.2 | 9 | 3.1 | 9 | 3.0 | 9 | 2.8 | 9 | 2.7 |
| | I | ALS-SI | 40 | 1.2 | 40 | 1.2 | 46 | 1.4 | 50 | 1.6 | 56 | 1.8 | 72 | 2.4 | 101 | 3.4 |
| | | INDAFAC | 9 | 8.8 | 9 | 8.1 | 9 | 7.5 | 9 | 6.9 | 9 | 6.4 | 9 | 5.8 | 10 | 5.6 |
| | II | ALS-SI | 76 | 1.6 | 70 | 1.5 | 90 | 1.9 | 81 | 1.8 | 111 | 2.5 | 125 | 2.8 | 176 | 4.1 |
| | | INDAFAC | 30 | 12.1 | 18 | 7.8 | 23 | 7.8 | 16 | 6.1 | 20 | 6.3 | 22 | 5.9 | 25 | 5.9 |
| | III | ALS-SI | 243 | 12.2 | 235 | 13.6 | 292 | 15.8 | 349 | 19.9 | 376 | 20.9 | 520 | 29.6 | 598 | 35.0 |
| | | INDAFAC | 20 | 30.6 | 16 | 24.1 | 19 | 24.0 | 20 | 23.1 | 21 | 21.7 | 23 | 21.0 | 23 | 19.7 |
| RMS[b] | Monte Carlo | ALS-SI | – | – | – | – | 48 | 1.0 | 55 | 1.3 | 92 | 2.1 | 191 | 4.6 | 335 | 8.2 |
| | | INDAFAC | – | – | – | – | 9 | 3.4 | 10 | 3.4 | 10 | 3.3 | 13 | 3.3 | 16 | 3.4 |
| | I | ALS-SI | 38 | 1.1 | 42 | 1.3 | 48 | 1.5 | 50 | 1.6 | 77 | 2.5 | 348 | 11.4 | 718 | 24.1 |
| | | INDAFAC | 9 | 8.7 | 9 | 8.1 | 9 | 7.5 | 9 | 6.9 | 10 | 6.8 | 11 | 6.5 | 14 | 7.2 |
| | II | ALS-SI | 89 | 1.8 | 80 | 1.7 | 109 | 2.4 | 111 | 2.4 | 161 | 3.6 | 411 | 9.3 | 1114 | 25.8 |
| | | INDAFAC | 37 | 13.2 | 27 | 10.0 | 28 | 9.2 | 29 | 8.5 | 32 | 8.7 | 37 | 8.8 | 83 | 17.1 |
| | III | ALS | 242 | 12.9 | 273 | 14.6 | 271 | 14.9 | 352 | 20.0 | 436 | 23.1 | 486 | 29.1 | 728 | 43.1 |
| | | INDAFAC | 19 | 30.0 | 19 | 27.3 | 18 | 23.3 | 23 | 25.8 | 30 | 27.2 | 29 | 23.2 | 26 | 20.0 |
| SMS[c] | Monte Carlo | ALS-SI | – | – | – | – | 184 | 4.2 | 288 | 6.3 | 671 | 15.1 | 798 | 18.5 | 1267 | 30.4 |
| | | INDAFAC | – | – | – | – | 15 | 4.7 | 18 | 5.0 | 35 | 8.3 | 46 | 9.0 | 78 | 13.0 |
| | I | ALS-SI | 38 | 1.1 | 93 | 2.7 | 216 | 6.5 | 368 | 11.4 | 446 | 13.9 | 1294 | 42.0 | 2436 | 81.1 |
| | | INDAFAC | 9 | 8.7 | 14 | 11.7 | 15 | 11.2 | 18 | 11.9 | 48 | 22.3 | 58 | 25.3 | 83 | 30.3 |
| | II | ALS-SI | 104 | 2.1 | 320 | 6.6 | 785 | 16.4 | 1165 | 25.0 | 1843 | 40.3 | 4052 | 91.1 | 7222 | 164.1 |
| | | INDAFAC | 28 | 10.2 | 33 | 12.6 | 25 | 8.9 | 29 | 8.9 | 49 | 12.9 | 59 | 12.9 | 97 | 19.3 |
| | III | ALS-SI | 295 | 15.1 | 411 | 21.3 | 640 | 34.9 | 1770 | 90.8 | 3583 | 192.1 | 6110 | 359.1 | 10000 | 583.2 |
| | | INDAFAC | 25 | 37.7 | 27 | 36.9 | 30 | 34.9 | 37 | 37.7 | 73 | 62.0 | 81 | 60.5 | 131 | 88.1 |

[a] Randomly missing values.
[b] Randomly missing spectra.
[c] Systematically missing spectra.

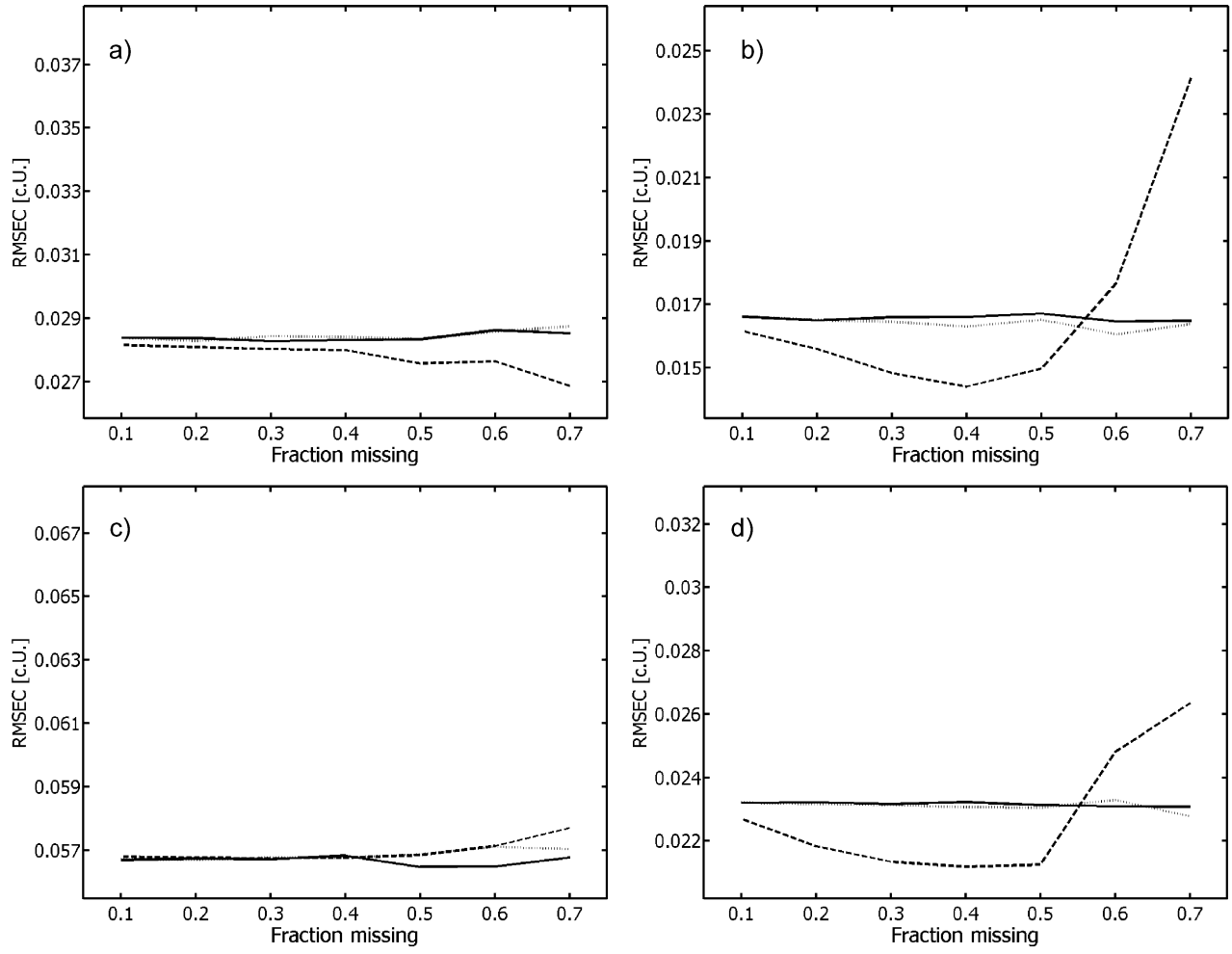


Fig. 5. Median RMSEC for the four constituents of data set I: (a) DOPA, (b) hydroquinone, (c) phenylalanine, and (d) tryptophan. The solid line refers to the RMV pattern, the dotted line to the RMS pattern, and the dashed line to the SMS pattern. All plots refer to the ALS-SI solutions.

data set I, most likely reflecting the higher difficulty, although some specific aspects are retained (Fig. 6). Tryptophan prediction improves with an increasing percentage of missing values, reaching a maximum at 70%. For DOPA, the minimum is reached between 30% and 50%, and then the quality slightly deteriorates. These observations are consistent with the hypothesis of influence of the Raman scatter; further analyses of the raw data showed that the subtraction of a blank is insufficient to completely remove the Raman scatter peaks. For the other constituents, the behaviour is quite the opposite and the predictions worsen considerably along with an increasing amount of missing information. Data set III (Fig. 7) essentially confirms the results illustrated thus far: tryptophan predictions slightly improve with the percentage of missing elements in the SMS case, with a clear worsening starting at 60%; hydroquinone and tyrosine behave analogously to data set II. The different behaviour of the same analyte (particularly of hydroquinone) with respect to the different sets is probably to be associated to more aspects than the sole Raman scatter (e.g., the higher difficulty of the fitting problems or instrumental effects); nonetheless, the results for the three data sets appear rather consistent.

The two algorithms, in terms of quality of the predictions, performed equivalently for all data sets and patterns. In general, the relative difference of the RMSEC between the two was contained within 0.1%, becoming larger only when the models themselves become very unstable.

With respect to the computational efficiency, the real data sets confirmed all the observations made for the simulations: the number of iterations required for the SMS case is much higher than for the two other patterns, to the point that at 70% missing 10 000 iterations were not sufficient for ALS-SI to converge to a solution. Table 5 makes apparent the correctness of the classification of the problems in terms of $\gamma_{\mathbf{J}}$: the number of iterations increases going from data set I to II to III, which is clearly the most difficult problem.

With respect to the time consumption, INDAFAC was faster on data sets II and III in 93–100% of the replicates for the SMS pattern starting at about 30–40% missing elements. On the other hand, ALS-SI was faster in the vast majority of

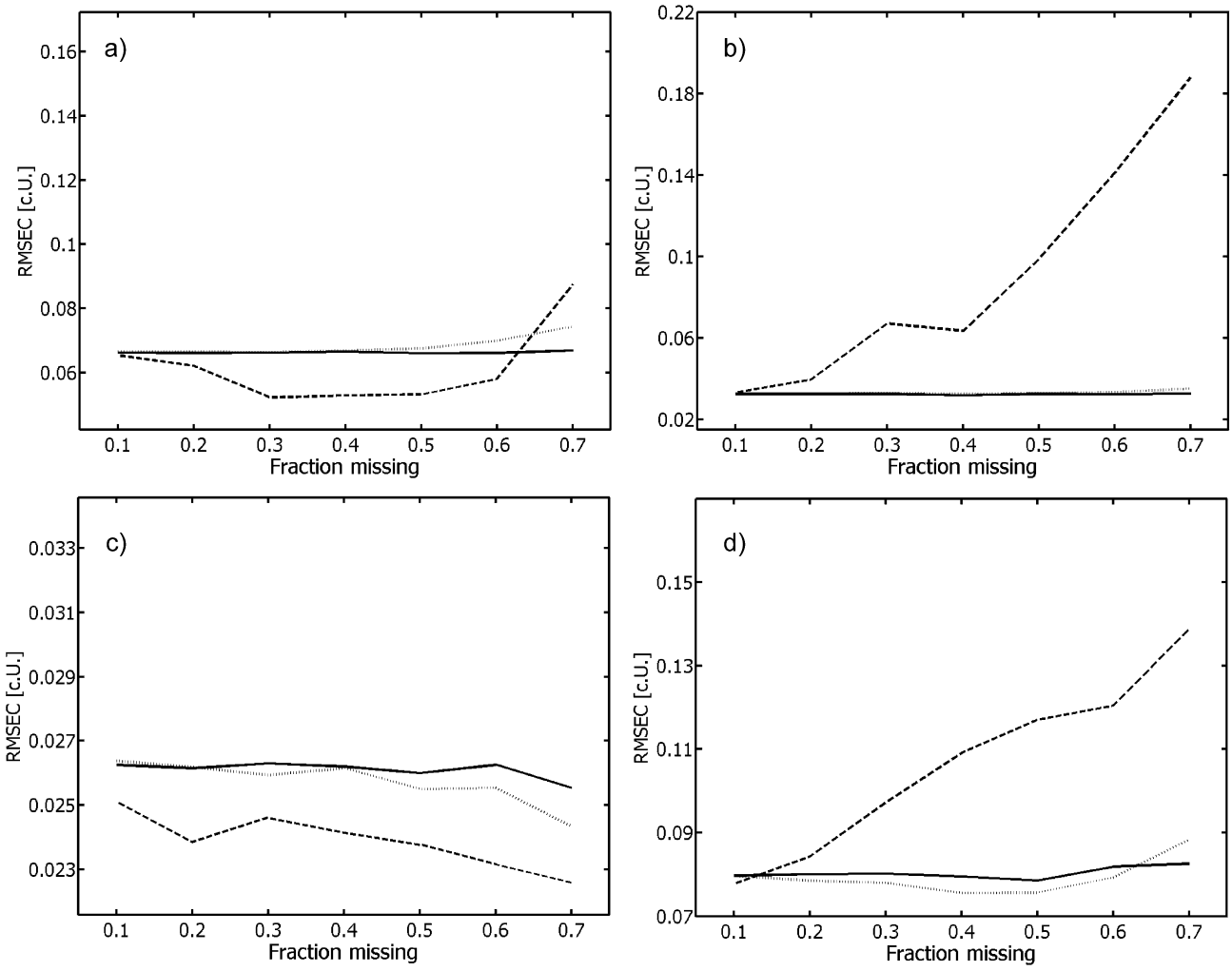


Fig. 6. Median RMSEC for the four constituents of data set II: (a) DOPA, (b) hydroquinone, (c) tryptophan, and (d) tyrosine. The solid line refers to the RMV pattern, the dotted line to the RMS pattern, and the dashed line to SMS pattern. All plots refer to the ALS-SI solutions.

the cases (for all the data sets and patterns) when 10% or 20% elements were missing.

## 5. Conclusions

Two algorithms for fitting the PARAFAC model in presence of missing values, ALS with single imputation, and INDAFAC—based on a Levenberg–Marquadt method for non-linear least squares—have been tested by means of a Monte Carlo simulation and on three fluorescence data sets of various complexity. In terms of capability of recovering the correct solution, they performed almost equally, although INDAFAC appeared slightly better for difficult problems (e.g., when the underlying components are very collinear). In terms of time consumption, the derivative-based algorithm is faster when the fraction of missing values exceeds 30% for patterns typical of fluorescence data and 60% when they are uniformly scattered over the array.

A classification was proposed for the possible patterns of missing elements within an array: randomly missing values (RMV) and spectra (RMS), and systematically missing values (SMV) and spectra (SMS). A clear association has been shown between these patterns and the performances of the two algorithms in fitting a PARAFAC model.

The most remarkable result is that a PARAFAC model can be successfully fit even when 70% of the values are missing compared to, for example, PCA on a matrix, for which the limit appears to be in the order of 25–40% [1,4]. The reason for this can be found in the trilinear structure of the PARAFAC model and its added rigidity. In spite of very large fractions of missing values, it was possible to adequately predict the concentration of analytes in synthetic solutions of up to five constituents.

Furthermore, possible explanations were given for the different behaviour of the algorithms with respect to the pattern of missing values, especially with respect to the SMS case, which is by far the most common in, e.g.,

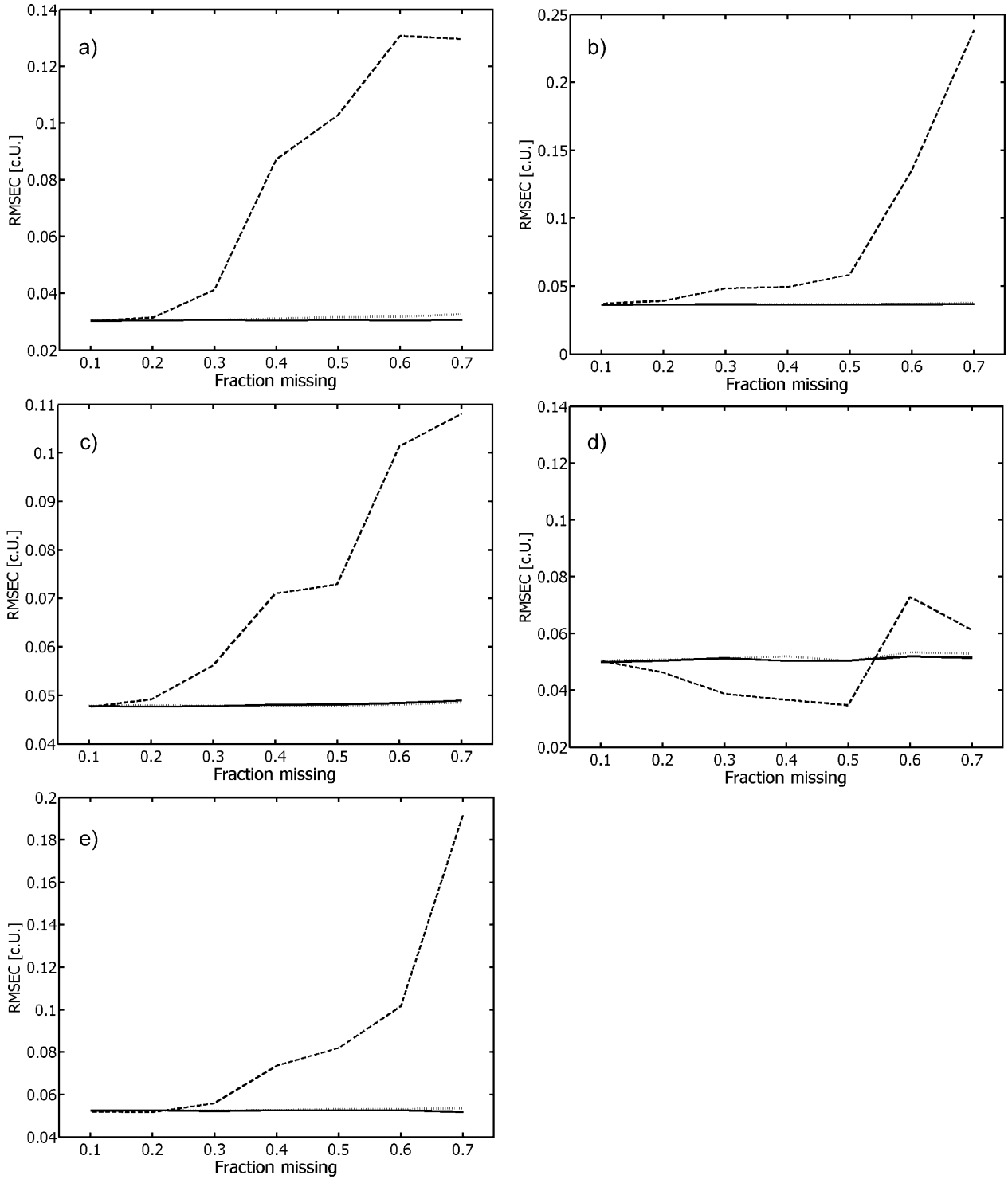


Fig. 7. Median RMSEC for the four constituents of data set III: (a) cathecol, (b) hydroquinone, (c) indole, (d) tryptophan, and (e) tyrosine. The solid line refers to the RMV pattern, the dotted line to the RMS pattern, and the dashed line to SMS pattern. All plots refer to the ALS-SI solutions.

fluorescence spectroscopy. Possibly, this will provide new tools for studying the application of more complex missing values patterns that do not interfere with multilinearity to the same extent as the SMS case presented here [25], or of ad hoc techniques for dealing with missing values and non-multilinear variation [31,32].

Finally, a new and very general tool has been proposed for establishing the difficulty of fitting a PARAFAC model: the Jacobian (practical) condition number $\gamma_{\mathbf{J}}$, which accounts not only for the collinearity in any of the loading matrices but also of the "interaction" between the model and the data it is fitted to.

## Acknowledgments

The authors would like to acknowledge F. van den Berg for the fruitful discussion, and Å. Rinnan for providing data set III.

## Appendix A. Effect of the SMS pattern on the convergence

A noiseless 2×20×20 array $\underline{\mathbf{X}}$ was generated where **A**=**I**, $\mathbf{b}_1$, and $\mathbf{c}_1$ were Gaussian curves with $\mu$=3, and $\sigma$=2, $\mathbf{b}_2$ and $\mathbf{c}_2$ were Gaussians with $\mu$=17 and $\sigma$=2 (Fig. 8a). The two components are orthogonal, and on the complete array with random initialisation, the algorithm always converged to the correct solution within the first five iterations. On the other hand, if 50% of the values were set to missing according to an SMS pattern (Fig. 8b), both algorithms never converged within the first 10 000 iterations. Fig. 8c shows the loading vectors of **B** and **C** for such problem: small peaks are visible in each loading vector in correspondence with the peak of the other component. These small "ghost" peaks have no effect on the loss function as they are entirely included in the missing areas (Fig. 8d), on the other hand, they interfere with the trilinear structure, as part of the second component is described by the first and vice versa. The problem becomes apparent if one computes the Tucker core and the core consistency [26] relative to the two solutions. When missing values are present, the core consistency is lower than 100% (99.28% in the case showed in the figure), and the Tucker core, while dominated by the two elements on the "superdiagonal", contains small values with opposite signs and almost equal magnitude($1.4 \cdot 10^{-5}$) at positions $g_{211}$ and $g_{122}$:

$$\mathbf{G} = \left[\begin{array}{cc|cc} 4.0 \cdot 10^{-2} & 4.1 \cdot 10^{-3} & 4.1\ 10^{-3} & -1.4 \cdot 10^{-5} \\ 1.5 \cdot 10^{-5} & 4.3 \cdot 10^{-3} & 4.2\ 10^{-3} & 4.1 \cdot 10^{-2} \end{array}\right].$$

As the iterative method proceeds (both ALS-SI and INDAFAC), the decrease in the loss function becomes increasingly small, the loss function tends to zero, and the core consistency to 100%. Repeated tests confirmed this observation. The shape of the ghost peaks in real life would be affected by noise or other nonmultilinear structures in the data [6].

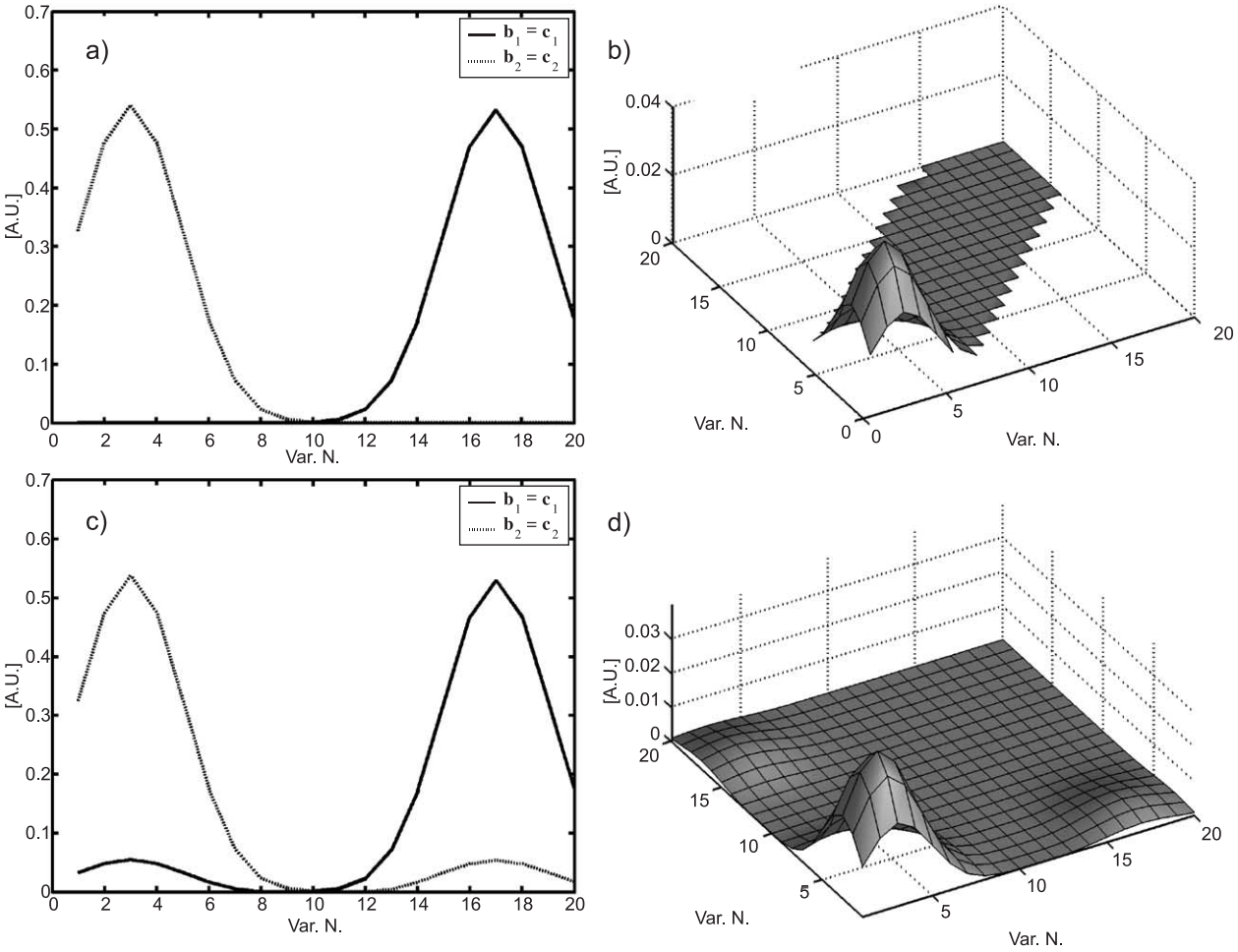


Fig. 8. (a) **B** (and **C**) of a noiseless 2×20×20 array; (b) landscape of the first horizontal slab of $\underline{\mathbf{X}}$; (c) **B** (and **C**) of a noiseless 2×20×20 array with 50% missing values in the SMS pattern; (d) landscape of the first horizontal slab in the PARAFAC model of $\underline{\mathbf{X}}$.

## References


[1] B. Walczak, D.L. Massart, Chemom. Intell. Lab. Syst. 58 (2001) 15–27.
[2] B. Walczak, D.L. Massart, Chemom. Intell. Lab. Syst. 58 (2001) 29–42.
[3] P.R.C. Nelson, P.A. Taylor, J.F. MacGregor, Chemom. Intell. Lab. Syst. 35 (1996) 45–65.
[4] B. Grung, R. Manne, Chemom. Intell. Lab. Syst. 42 (1998) 125–139.
[5] L.G. Thygesen, A. Rinnan, S. Barsberg, J.K.S. Moller, Chemom. Intell. Lab. Syst. 71 (2004) 97–106.
[6] C.M. Andersen, R. Bro, J. of Chemom. 17 (2003) 200–215.
[7] R. Bro, Multi-way Analysis in the Food Industry. Models, Algorithms, and Applications, PhD thesis, University of Amsterdam, 1998.
[8] J.R. Magnus, H. Neudecker, Matrix Differential Calculus with Applications in Statistics and Econometrics Rev., John Wiley and Sons, New York, NY, USA, 1999.
[9] L. Vega-Montoto, P.D. Wentzell, J. Chemom. 17 (2003) 237–253.
[10] N.M. Faber, R. Bro, P.K. Hopke, Chemom. Intell. Lab. Syst. 65 (2003) 119–137.
[11] G. Tomasi, R. Bro, A comparison of methods fitting the PARAFAC model, submitted for publication.
[12] R. Bro, Chemom. Intell. Lab. Syst. 38 (1997) 149–171.
[13] R.A. Harshman, UCLA Work. Pap. Phon. 16 (1970) 1–84.
[14] P. Paatero, J. Chemom. 14 (2000) 285–299.
[15] B.C. Mitchell, D.S. Burdick, J. Chemom. 8 (1994) 155–168.
[16] A.P. Dempster, N.M. Laird, D.B. Rubin, J. R. Stat. Soc., B, Methodol. 39 (1977) 1–38.
[17] R.J.A. Little, D.B. Rubin, Statistical Analysis with Missing Data, John Wiley and Sons, New York, NY, USA, 1987.
[18] G.J. McLachlan, T. Krishnan, The EM Algorithm and Extensions, John Wiley and Sons, New York, NY, USA, 1997.
[19] Å. Björck, Numerical Methods for Least Squares Problems, Society for Industrial and Applied Mathematics, Philadephia, PA, USA, 1996, p. 339.
[20] K. Madsen, H.B. Nielsen, O. Tingleff, Methods for Non-linear Least Squares Problems, Dept. Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 2004.
[21] P. Paatero, Chemom. Intell. Lab. Syst. 38 (1997) 223–242.
[22] P. Paatero, J. Comput. Graph. Stat. 8 (1999) 854–888.
[23] H.A.L. Kiers, J.M.F. Ten Berge, R. Bro, J. Chemom. 13 (1999) 275–294.
[24] D. Baunsgaard, Factors Affecting 3-way Modelling (PARAFAC) of Fluorescence Landscapes, The Royal Veterinary and Agricultural University, Frederiksberg, Denmark, 1999.
[25] Å. Rinnan, Application of PARAFAC on spectral data, PhD thesis, The Royal Veterinary and Agricultural University, Frederiksberg, Denmark, 2004.
[26] R. Bro, H.A.L. Kiers, J. Chemom. 17 (2003) 274–286.
[27] H.A.L. Kiers, J. Chemom. 12 (1998) 155–171.
[28] B.C. Mitchell, D.S. Burdick, Chemom. Intell. Lab. Syst. 20 (1993) 149–161.
[29] J. Riu, R. Bro, Chemom. Intell. Lab. Syst. 65 (2003) 35–49.
[30] K.R. Gabriel, S. Zamir, Technometrics 21 (1979) 489–498.
[31] P.K. Hopke, P. Paatero, H. Jia, R.T. Ross, R.A. Harshman, Chemom. Intell. Lab. Syst. 43 (1998) 25–42.
[32] R. Bro, N.D. Sidiropoulos, A.K. Smilde, J. Chemom. 16 (2002) 387–400.