JOURNAL OF CHEMOMETRICS
J. Chemometrics 2005; **19**: 65–76
Published online 7 September 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cem.911

Alternating penalty trilinear decomposition algorithm for second-order calibration with application to interference-free analysis of excitation–emission matrix fluorescence data

A-Lin Xia, Hai-Long Wu*, Dong-Mei Fang, Yu-Jie Ding, Le-Qian Hu and Ru-Qin Yu

State Key Laboratory of Chemo/Biosensing and Chemometrics, College of Chemistry and Chemical Engineering, Hunan University, Changsha 410082, People's Republic of China

Received 15 October 2004; Revised 12 March 2005; Accepted 12 March 2005

A new method, alternating penalty trilinear decomposition (APTLD), is developed for the decomposition of three-way data arrays. By utilizing the alternating least squares principle and alternating penalty constraints to minimize three different alternating penalty errors simultaneously, the intrinsic profiles are found. The APTLD algorithm can avoid the two-factor degeneracy problem and relieve the slow convergence problem, which is difficult to handle for the traditional parallel factor analysis (PARAFAC) algorithm. It retains the second-order advantage of quantification for analytes of interest even in the presence of potentially unknown interferents. In additions, it is insensitive to the estimated component number, thus avoiding the difficulty of determining a correct component number for the model, which is intrinsic in the PARAFAC algorithm. The results of treating one simulated and one real excitation–emission spectral data set showed that the proposed algorithm performs well as long as the model dimensionality chosen is not less than the actual number of components. Furthermore, the performance of the APTLD algorithm sometimes surpasses that of the PARAFAC algorithm in the prediction of concentration profiles even if the component number chosen is the same as the actual number of underlying factors in real samples. Copyright © 2005 John Wiley & Sons, Ltd.

KEYWORDS: alternating penalty trilinear decomposition (APTLD); three-way data; PARAFAC; ATLD; second-order calibration

1. INTRODUCTION

Determination of the components of interest in complex mixtures is a challenging problem in analytical chemistry. The common practice to tackle the problem is to resort to some physical and chemical separation techniques. Accordingly, this solution is always time-consuming and cost-expensive. Moreover, the equilibrium may be broken by the separation procedure when there exists a chemical equilibrium in the mixtures, which will mislead quantification of the analytical components of interest. However, with the development of modern high-order analytical instruments and data collection techniques as well as the application of chemometric methods dealing with three-way data sets, the problem may be solved. Furthermore, three-way data analysis has become an active domain in chemometrics research [1–7]. It has become ever more significant to develop available methods which may be applied to these three-way data [1,3,7,8]. The attractive merit derived from three-way data arrays lies in the fact that the analysis of several components of interest can be quantified even in the presence of unknown interferents, commonly called the 'second -order advantage' [3,9].

There are two types of algorithm for the decomposition of three-way data arrays. One type is based on generalized eigenanalysis to resolve the data arrays, with the well-known examples of the generalized rank annihilation method (GRAM) [10–12] and the direct trilinear decomposition (DTLD) method [13–15]. Unfortunately, GRAM is constrained to use only one standard and one mixture sample at a time. Although DTLD allows for a direct solution through multiple samples, it requires the construction of two pseudo-samples, which unavoidably causes a loss of information in multiple samples. Furthermore, these approaches may occasionally yield imaginary solutions and exhibit inflated variance. The other type of algorithm is an iterative

*Correspondence to: H.-L. Wu, State Key Laboratory of Chemo/Biosensing and Chemometrics, College of Chemistry and Chemical Engineering, Hunan University, Changsha 410082, People's Republic of China.
E-mail: hlwu@hnu.cn
Contract/grant sponsor: National Natural Science Foundation of China; contract/grant numbers: 20475014; 20375012; 20435010.

one [7,16–25], represented by the parallel factor analysis (PARAFAC) algorithm proposed by Harshman [26]. Although this method provides a best fit to the three-way data array and has been successfully utilized to solve many chemical problems, it will give chemically meaningless solutions when trapped in computational swamps. In addition, the solutions finally obtained using this method are rather unstable unless the chosen component number equals the actual one, which leads to a dilemma that is hard to handle in practical problem solving. Moreover, this method suffers from annoyingly slow convergence, thus always requiring a long time to resolve the data array. However, following alternating trilinear decomposition (ATLD) [18], some algorithms [27–29] which can avoid the above relatively strict constraint have tried to solve this problem.

Faber *et al*. [30] compared several recently proposed algorithms, including alternating least squares (ALS) [31], DTLD [13], ATLD [18], self-weighted alternating trilinear decomposition (SWATLD) [27], pseudo alternating least squares (PALS) [28], alternating coupled vector resolution (ACOVER) [29], alternating slice-wise diagonalization (ASD) [32] and alternating coupled matrix resolution (ACOMAR) [33]. It was found that the ALS estimated models are generally of a better quality than any of the alternatives even when overfactoring the model, but ALS is significantly slower. However, it must be pointed out that, according to References [30,34], the SWATLD algorithm has the advantages of fast convergence and insensitivity to excess factors used in calculations, and offers better results than other second-order algorithms (including ALS). (The routine used for applying SWATLD was generously supplied by N. M. Faber.) These results suggest that related works on three-way data analysis and second-order calibration are worthy of further study.

In this paper a new algorithm with some improved properties, alternating penalty trilinear decomposition (APTLD), is developed for the trilinear analysis of three-way data arrays. Unlike existing methods, the algorithm aims at using alternately the constraint functions as penalty terms of the PARAFAC error, which minimizes three new least squares-based objective functions. A virtue of the method compared with GRAM is that it provides a statistically plausible manner to make use of multi-sample information. Another salient advantage of APTLD is that the resolved profiles are very stable with respect to the model dimensionality when the chosen dimensionality is not less than the actual number of components, thus avoiding the dilemma in selecting a proper component number for the model. In addition, the proposed APTLD criterion provides a natural way to avoid the problem of so-called two-factor degeneracy [21,35,36]. Moreover, the proposed algorithm can overcome the slow convergence brought about by random initialization or high multicollinearity to some extent. In other words, the APTLD algorithm has a much higher convergence rate compared with the traditional PARAFAC algorithm [37–38]. The results of treating one simulated and one real excitation–emission spectral data set showed that the proposed algorithm performs well. Furthermore, the performance of APTLD sometimes surpasses that of PARAFAC in the prediction of concentration profiles even if the component number chosen is the same as the actual number of underlying factors in real samples.

2. NOMENCLATURE

Throughout this paper, scalars are represented by lowercase italics, bold lowercase characters denote vectors, bold capitals designate two-way matrices, underlined bold capitals symbolize three-way arrays, and superscript T denotes the transpose of a matrix. Before reading the following sections of the paper, readers are recommended to refer to the nomenclature below for detailed information:

- $\underline{\mathbf{X}}$—three-way data array;
- I, J, K—the dimensions of the three modes of $\underline{\mathbf{X}}$;
- $\mathbf{A}_{I\times N}$, $\mathbf{B}_{J\times N}$, $\mathbf{C}_{K\times N}$—the three underlying loading matrices of $\underline{\mathbf{X}}$ with dimensions $I\times N$, $J\times N$, and $K\times N$ respectively (which will be simply represented by $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ respectively in this paper);
- x_{ijk}—the ijkth element of $\underline{\mathbf{X}}$;
- a_{in}, b_{jn}, c_{kn}—the inth, jnth and knth elements of the three underlying loading matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ respectively;
- $\boldsymbol{a}_{(i)}$, $\boldsymbol{b}_{(j)}$, $\boldsymbol{c}_{(k)}$—the ith, jth and kth rows of profile matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ respectively;
- $\mathrm{diag}(\boldsymbol{a}_{(i)})$, $\mathrm{diag}(\boldsymbol{b}_{(j)})$, $\mathrm{diag}(\boldsymbol{c}_{(k)})$—diagonal matrices with elements $\boldsymbol{a}_{(i)}$, $\boldsymbol{b}_{(j)}$ and $\boldsymbol{c}_{(k)}$ respectively;
- $\mathbf{X}_{i..}, \mathbf{X}_{.j.}, \mathbf{X}_{..k}$—the ith horizontal, jth lateral and kth frontal slices of $\underline{\mathbf{X}}$ respectively;
- $\mathbf{E}_{i..}, \mathbf{E}_{.j.}, \mathbf{E}_{..k}$—the ith horizontal, jth lateral and kth frontal slices of the three-way residue array $\underline{\mathbf{E}}$ respectively;
- e_{ijk}—the ijkth element of the three-way residue array $\underline{\mathbf{E}}$;
- $\mathbf{A}^+, \mathbf{B}^+, \mathbf{C}^+$—the Moore–Penrose generalized inverses of matrices $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ respectively;
- $\|\cdot\|_F$—the Frobenius matrix norm.

3. THEORY

3.1. Trilinear model for second-order calibration

In second-order calibration the famous trilinear model proposed by Harshman [26] and Carroll and Chang [39] has been widely accepted owing to its consistency with Beer's law in chemistry. According to the trilinear model, which is depicted in Figure 1, each element x_{ijk} of the data array $\underline{\mathbf{X}}$ can be represented as

$$x_{ijk} = \sum_{n=1}^{N} a_{in} b_{jn} c_{kn} + e_{ijk} \quad (i = 1,2,\ldots,I; j = 1,2,\ldots,J; k = 1,2,\ldots,K) \tag{1}$$

The trilinear model can be expressed in matrix form as

$$\mathbf{X}_{i..} = \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}} + \mathbf{E}_{i..} \quad (i = 1,2,\ldots,I) \tag{2}$$

$$\mathbf{X}_{.j.} = \mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}} + \mathbf{E}_{.j.} \quad (j = 1,2,\ldots,J) \tag{3}$$

$$\mathbf{X}_{..k} = \mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}} + \mathbf{E}_{..k} \quad (k = 1,2,\ldots,K) \tag{4}$$

Regardless of scaling and permutation, the decomposition of the trilinear model proposed above will be unique and have no free rotations provided that $k_1 + k_2 + k_3 \geqslant 2N + 2$ [36],

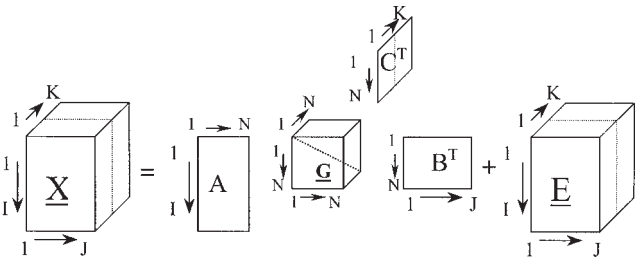

Figure 1. Graphical representation of trilinear model of three-way data array $\underline{\mathbf{X}}$: **A**, relative first loading matrix of size $I \times N$; **B**, relative second loading matrix of size $J \times N$; **C**, relative concentration matrix of size $K \times N$; $\underline{\mathbf{G}}$, three-way diagonal core array of size $N \times N \times N$ with ones on the superdiagonal and zeros elsewhere; $\underline{\mathbf{E}}$, three-way residual data array of size $I \times J \times K$. Note that I is the number of occasions (excitation wavelengths), J is the number of variables (emission wavelengths), K is the number of samples, including standards and unknowns, and N is the estimated number of factors.

where k_1, k_2 and k_3 are the k-ranks of **A**, **B** and **C** respectively. In other words, loading matrices **A**, **B** and **C** will be resolved in a unique way.

3.2. Least squares-based penalty term and alternating penalty (AP) error

From Equations (2)–(4) one can obtain the residue functions

$$\mathbf{E}_I(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{i=1}^{I} \left\| \mathbf{X}_{i..} - \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}} \right\|_F^2 \tag{5}$$

$$\mathbf{E}_J(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{j=1}^{J} \left\| \mathbf{X}_{.j.} - \mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{T} \right\|_F^2 \tag{6}$$

$$\mathbf{E}_K(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{k=1}^{K} \left\| \mathbf{X}_{..k} - \mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{T} \right\|_F^2 \tag{7}$$

According to the least squares principle, for a trilinear model $\mathbf{X}_{i..} = \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}} + \mathbf{E}_{i..}$ $(i = 1, 2, \ldots, I)$ the following equations hold:

$$\begin{aligned} \mathbf{B}^{+}\mathbf{X}_{i..} &= \mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}} \text{ and } \mathbf{X}_{i..}(\mathbf{C}^{\mathrm{T}})^{+} \\ &= \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)}) \quad (i = 1, 2, \ldots, I) \end{aligned} \tag{8}$$

$$\begin{aligned} \mathbf{C}^{+}\mathbf{X}_{.j.} &= \mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}} \text{ and } \mathbf{X}_{.j.}(A^{\mathrm{T}})^{+} \\ &= \mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)}) \quad (j = 1, 2, \ldots, J) \end{aligned} \tag{9}$$

$$\begin{aligned} \mathbf{A}^{+}\mathbf{X}_{..k} &= \mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}} \text{ and } \mathbf{X}_{..k}(B^{\mathrm{T}})^{+} \\ &= \mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)}) \quad (k = 1, 2, \ldots, K) \end{aligned} \tag{10}$$

Moreover, making use of the above equations alternately as the constraint terms and combining them with residue functions (5)–(7), one can obtain the following three constrainted problems:

$$\begin{aligned} &\text{problem 1:} \quad \min \mathbf{E}_I(\mathbf{A},\mathbf{B},\mathbf{C}) \quad \text{s.t.} \\ &\sum_{k=1}^{K} \left\| \mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C})))(\mathbf{A}^{+}\mathbf{X}_{..k} - \mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}}) \right\|_F^2 = 0 \\ &\sum_{i=1}^{I} \left\| (\mathbf{X}_{i..}(\mathbf{C}^{\mathrm{T}})^{+} - \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)}))\mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))) \right\|_F^2 = 0 \end{aligned} \tag{11}$$

$$\begin{aligned} &\text{problem 2:} \quad \min \mathbf{E}_J(\mathbf{A},\mathbf{B},\mathbf{C}) \quad \text{s.t.} \\ &\sum_{i=1}^{I} \left\| \mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A})))(\mathbf{B}^{+}\mathbf{X}_{i..} - \mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}}) \right\|_F^2 = 0 \\ &\sum_{j=1}^{J} \left\| (\mathbf{X}_{j.}(\mathbf{A}^{\mathrm{T}})^{+} - \mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)}))\mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))) \right\|_F^2 = 0 \end{aligned} \tag{12}$$

$$\begin{aligned} &\text{problem 3:} \quad \min \mathbf{E}_K(\mathbf{A},\mathbf{B},\mathbf{C}) \quad \text{s.t.} \\ &\sum_{j=1}^{J} \left\| \mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B})))(\mathbf{C}^{+}\mathbf{X}_{.j.} - \mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}}) \right\|_F^2 = 0 \\ &\sum_{k=1}^{K} \left\| \left(\mathbf{X}_{..k}(\mathbf{B}^{\mathrm{T}})^{+} - \mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)})\right)\mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))) \right\|_F^2 = 0 \end{aligned} \tag{13}$$

where s.t. stands for 'subject to' (constraint conditions), min denotes 'minimize', ./ denotes array division (e.g. suppose $\mathbf{x} = (x_i)$ and $\mathbf{y} = (y_i)$, then $\mathbf{x}./\mathbf{y} = (x_i/y_i)$); sqrt is the square root operator and **1** is a vector of length N with all elements equal to one; diagm denotes column vector with elements equal to diagonal elements of a square matrix.

Aimed at transforming the above constrained problems into non-constrained ones, the APTLD method utilizes the corresponding constraint functions as penalty terms and combines these with residue functions (5)–(7) to construct three objective functions. Then it decomposes the model by alternately minimizing the following three objective functions (AP error):

$$\begin{aligned} S(B) = &\sum_{i=1}^{I} \left\| \mathbf{X}_{i..} - \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}} \right\|_F^2 \\ &+ r\left(\sum_{k=1}^{K} \left\| \mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C})))(\mathbf{A}^{+}\mathbf{X}_{..k} - \mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}}) \right\|_F^2 \right. \\ &\left. + \sum_{i=1}^{I} \left\| \left(\mathbf{X}_{i..}(\mathbf{C}^{\mathrm{T}})^{+} - \mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)})\right)\mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))) \right\|_F^2 \right) \end{aligned} \tag{14}$$

$$\begin{aligned} S(C) = &\sum_{j=1}^{J} \left\| \mathbf{X}_{.j.} - \mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}} \right\|_F^2 \\ &+ p\left(\sum_{i=1}^{I} \left\| \mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A})))(\mathbf{B}^{+}\mathbf{X}_{i..} - \mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}}) \right\|_F^2 \right. \\ &\left. + \sum_{j=1}^{J} \left\| \left(\mathbf{X}_{.j.}(\mathbf{A}^{\mathrm{T}})^{+} - \mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)})\right)\mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))) \right\|_F^2 \right) \end{aligned} \tag{15}$$

$$\begin{aligned} S(A) = &\sum_{k=1}^{K} \left\| \mathbf{X}_{..k} - \mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}} \right\|_F^2 \\ &+ q\left(\sum_{j=1}^{J} \left\| \mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B})))(\mathbf{C}^{+}\mathbf{X}_{.j.} - \mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}}) \right\|_F^2 \right. \\ &\left. + \sum_{k=1}^{K} \left\| \left(\mathbf{X}_{..k}(\mathbf{B}^{\mathrm{T}})^{+} - \mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)})\right)\mathrm{diag}(\mathrm{sqrt}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))) \right\|_F^2 \right) \end{aligned} \tag{16}$$

where r, p and q are penalty factors.

3.3. APTLD algorithm

According to the above objective functions, an alternating penalty algorithm is used to exploit the solution. This

minimizes the three objective functions in an alternating manner, i.e. it minimizes the APTLD error Equation (14) over **B** for fixed **C** and **A**, Equation (15) over **C** for fixed **A** and **B** and Equation (16) over **A** for fixed **B** and **C**. Details of the procedure are described below.

If **B** minimizes $S(B)$ (Equation (14)) for fixed **C** and **A**, it is necessary for **B** to satisfy the condition

$$\begin{aligned}\frac{\partial S(B)}{\partial B} = \sum_{i=1}^{I}\Big[& -2\mathbf{X}_{i..}\mathbf{C}\mathrm{diag}(\boldsymbol{a}_{(i)}) + 2\mathbf{B}\mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}}\mathbf{C}\mathrm{diag}(\boldsymbol{a}_{(i)}) \\ & - 2r\mathbf{X}_{i..}(\mathbf{C}^{\mathrm{T}})^{+}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))\mathrm{diag}(\boldsymbol{a}_{(i)}) \\ & + 2r\mathbf{B}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))\mathrm{diag}(\boldsymbol{a}_{(i)}))^{2}\Big] \\ & + \sum_{k=1}^{K}\Big[-2r\mathbf{X}_{..k}^{\mathrm{T}}(\mathbf{A}^{+})^{\mathrm{T}}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))\mathrm{diag}(\boldsymbol{c}_{(k)}) \\ & + 2r\mathbf{B}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))(\mathrm{diag}(\boldsymbol{c}_{(k)}))^{2}\Big] = 0 \end{aligned} \tag{17}$$

For fixed **A** and **B** the reasonable **C** should satisfy

$$\begin{aligned}\frac{\partial S(C)}{\partial C} = \sum_{j=1}^{J}\Big[& -2\mathbf{X}_{.j.}\mathbf{A}\mathrm{diag}(\boldsymbol{b}_{(j)}) + 2\mathbf{C}\mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}}\mathbf{A}\mathrm{diag}(\boldsymbol{b}_{(j)}) \\ & - 2p\mathbf{X}_{.j.}(\mathbf{A}^{\mathrm{T}})^{+}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))\mathrm{diag}(\boldsymbol{b}_{(j)}) \\ & + 2p\mathbf{C}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))(\mathrm{diag}(\boldsymbol{b}_{(j)}))^{2}] \\ & + \sum_{i=1}^{I}\Big[-2p\mathbf{X}_{i..}^{\mathrm{T}}(\mathbf{B}^{+})^{\mathrm{T}}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))\mathrm{diag}(\boldsymbol{a}_{(i)}) \\ & + 2p\mathbf{C}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))(\mathrm{diag}(\boldsymbol{a}_{(i)}))^{2}\Big] = 0 \end{aligned} \tag{18}$$

For fixed **B** and **C** the reasonable **A** should satisfy

$$\begin{aligned}\frac{\partial S(A)}{\partial A} = \sum_{k=1}^{K}\Big[& -2\mathbf{X}_{..k}\mathbf{B}\mathrm{diag}(\boldsymbol{c}_{(k)}) + 2\mathbf{A}\mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}}\mathbf{B}\mathrm{diag}(\boldsymbol{c}_{(k)}) \\ & - 2q\mathbf{X}_{..k}(\mathbf{B}^{\mathrm{T}})^{+}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))\mathrm{diag}(\boldsymbol{c}_{(k)}) \\ & + 2q\mathbf{A}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))(\mathrm{diag}(\boldsymbol{c}_{(k)}))^{2}\Big] \\ & + \sum_{j=1}^{J}\Big[-2q\mathbf{X}_{.j.}^{\mathrm{T}}(\mathbf{C}^{+})^{\mathrm{T}}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))\mathrm{diag}(\boldsymbol{b}_{(j)}) \\ & + 2q\mathbf{A}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))(\mathrm{diag}(\boldsymbol{b}_{(j)}))^{2}\Big] = 0 \end{aligned} \tag{19}$$

Thus one can obtain the update of **B** for fixed **C** and **A** from Equation (17) as

$$\begin{aligned}\mathbf{B} = & \Big(\sum_{i=1}^{I}\mathbf{X}_{i..}(\mathbf{C} + r(\mathbf{C}^{\mathrm{T}})^{+}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A})))\mathrm{diag}(\boldsymbol{a}_{(i)}) \\ & + r\sum_{k=1}^{K}\mathbf{X}_{..k}^{\mathrm{T}}(\mathbf{A}^{+})^{\mathrm{T}}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))\mathrm{diag}(\boldsymbol{c}_{(k)})\Big) \\ & \Big(\sum_{i=1}^{I}\mathrm{diag}(\boldsymbol{a}_{(i)})\,(\mathbf{C}^{\mathrm{T}}\mathbf{C} + r\,\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A})))\mathrm{diag}(\boldsymbol{a}_{(i)}) \\ & + r\sum_{k=1}^{K}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C}))\,(\mathrm{diag}(\boldsymbol{c}_{(k)}))^{2}\Big)^{+} \end{aligned} \tag{20}$$

From Equation (18), the update of **C** for fixed **A** and **B** is

$$\begin{aligned}\mathbf{C} = & \Big(\sum_{j=1}^{J}\mathbf{X}_{.j.}(\mathbf{A} + p(\mathbf{A}^{\mathrm{T}})^{+}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B})))\mathrm{diag}(\boldsymbol{b}_{(j)}) \\ & + p\sum_{i=1}^{I}\mathbf{X}_{i..}^{\mathrm{T}}(\mathbf{B}^{+})^{\mathrm{T}}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))\mathrm{diag}(\boldsymbol{a}_{(i)})\Big) \\ & \Big(\sum_{j=1}^{J}\mathrm{diag}(\boldsymbol{b}_{(j)})(\mathbf{A}^{\mathrm{T}}\mathbf{A} + p\,\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B})))\mathrm{diag}(\boldsymbol{b}_{(j)}) \\ & + p\sum_{i=1}^{I}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{A}^{\mathrm{T}}\mathbf{A}))\;(\mathrm{diag}(\boldsymbol{a}_{(i)}))^{2}\Big)^{+} \end{aligned} \tag{21}$$

From Equation (19) the update of **A** for fixed **C** and **B** is

$$\begin{aligned}\mathbf{A} = & \Big(\sum_{k=1}^{K}\mathbf{X}_{..k}(\mathbf{B} + q(\mathbf{B}^{\mathrm{T}})^{+}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C})))\mathrm{diag}(\boldsymbol{c}_{(k)}) \\ & + q\sum_{j=1}^{J}\mathbf{X}_{.j.}^{\mathrm{T}}(\mathbf{C}^{+})^{\mathrm{T}}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))\mathrm{diag}(\boldsymbol{b}_{(j)})\Big) \\ & \Big(\sum_{k=1}^{K}\mathrm{diag}(\boldsymbol{c}_{(k)})(\mathbf{B}^{\mathrm{T}}\mathbf{B} + q\;\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{C}^{\mathrm{T}}\mathbf{C})))\mathrm{diag}(\boldsymbol{c}_{(k)}) \\ & + q\sum_{j=1}^{J}\mathrm{diag}(\mathbf{1}./\mathrm{diagm}(\mathbf{B}^{\mathrm{T}}\mathbf{B}))\;(\mathrm{diag}(\boldsymbol{b}_{(j)}))^{2}\Big)^{+} \end{aligned} \tag{22}$$

In addition, when $p = q = r = 0$, Equations (20)–(22) will reduce to

$$\mathbf{B} = \Big(\sum_{i=1}^{I}\mathbf{X}_{i..}\mathbf{C}\mathrm{diag}(\boldsymbol{a}_{(i)})\Big)\Big(\sum_{i=1}^{I}\mathrm{diag}(\boldsymbol{a}_{(i)})\mathbf{C}^{\mathrm{T}}\mathbf{C}\mathrm{diag}(\boldsymbol{a}_{(i)})\Big)^{+} \tag{23}$$

$$\mathbf{C} = \Big(\sum_{j=1}^{J}\mathbf{X}_{.j.}\mathbf{A}\mathrm{diag}(\boldsymbol{b}_{(j)})\Big)\Big(\sum_{j=1}^{J}\mathrm{diag}(\boldsymbol{b}_{(j)})\mathbf{A}^{\mathrm{T}}\mathbf{A}\mathrm{diag}(\boldsymbol{b}_{(j)})\Big)^{+} \tag{24}$$

$$\mathbf{A} = \Big(\sum_{k=1}^{K}\mathbf{X}_{..k}\mathbf{B}\mathrm{diag}(\boldsymbol{c}_{(k)})\Big)\Big(\sum_{k=1}^{K}\mathrm{diag}(\boldsymbol{c}_{(k)})\mathbf{B}^{\mathrm{T}}\mathbf{B}\mathrm{diag}(\boldsymbol{c}_{(k)})\Big)^{+} \tag{25}$$

Obviously these formulae can be considered as a variant of the traditional PARAFAC algorithm.

Having derived the updating equations for these parameter matrices, we can present the general algorithm for the APTLD method as follows.

1. Randomly initialize **A** and **B** and choose an appropriate penalty factor p, q, r.
2. Compute **C** using Equation (21).
3. Compute **A** using Equation (22).
4. Compute **B** using Equation (20).
5. Scale **A** to be columnwise normalized.
6. Scale **B** to be columnwise normalized.
7. Compute **C** using Equation (21).
8. Repeat steps 3–7 until a stopping criterion is satisfied.

4. EXPERIMENTAL

To investigate the performance of the proposed method, we employ two experiments as examples. One experiment is a computer simulation and the other is a chemical analysis for

fluoroquinolones. The data analysis and simulation were carried out in the Matlab environment.

4.1. Simulated excitation–emission matrix fluorescence data

A data array was simulated which was produced by a fluorescence spectrophotometer on seven samples of four species. The excitation spectral profiles a_1–a_4 of the four species were generated by

$$a_{1,i} = 0.8 * \mathrm{gs}(2i-1, 30, 30) + 0.8 * \mathrm{gs}(2i-1, 60, 10)$$

$$a_{2,i} = 0.5 * \mathrm{gs}(2i-1, 20, 20) + 0.3 * \mathrm{gs}(2i-1, 50, 30)$$

$$a_{3,i} = 0.8 * \mathrm{gs}(2i-1, 30, 15) + 0.2 * \mathrm{gs}(2i-1, 60, 20)$$

$$a_{4,i} = 0.3 * \mathrm{gs}(2i-1, 55, 10) + 0.5 * \mathrm{gs}(2i-1, 30, 25)$$

with $i = 1, 2, 3, \ldots, 50$, where $\mathrm{gs}(x, a, b)$ refers to the value at x of a Gaussian function with centre a and standard deviation b, i.e. $\mathrm{gs}(x, a, b) = \exp[-(x-a)^2/2b^2]$. The emission spectral profiles $b_1 - b_4$ of the four species were produced by

$$b_{1,j} = 0.6 * \mathrm{gs}(4j-3, 50, 10) + 0.3 * \mathrm{gs}(4j-3, 60, 10)$$

$$b_{2,j} = 0.8 * \mathrm{gs}(4j-3, 30, 10) + 0.4 * \mathrm{gs}(4j-3, 70, 25)$$

$$b_{3,j} = 0.7 * \mathrm{gs}(4j-3, 50, 20) + 0.3 * \mathrm{gs}(4j-3, 60, 25)$$

$$b_{4,j} = 0.5 * \mathrm{gs}(4j-3, 40, 10) + 0.4 * \mathrm{gs}(4j-3, 50, 25)$$

The first two with $i = 1, 2, 3, \ldots, 25$ simulated samples contained only species 1–3, while the remaining five samples contained all four species. Their concentrations were uniformly distributed in the range 0–1. The three-way responses were generated exactly according to Equation (1), in which random errors were normally distributed with mean zero and standard deviation 0.3%. The data array was treated employing not only the APTLD and traditional PARAFAC algorithms but also the ATLD method to resolve the profiles of each component in three modes. Furthermore, the APTLD and ATLD algorithms as well as the traditional PARAFAC method were performed on the simulated data for comparison.

4.2. Real excitation–emission matrix fluorescence data

Ten samples containing different quantities of three chemical species, ofloxacin (OFL), norfloxacin (NOR) and enoxacin (ENO), were analysed by fluorescence spectrophotometry. The concentrations of each component are shown in Table I. The first six samples are used as concentration calibration samples and the last four are used as concentration prediction samples with ENO as interference species. All response matrices were recorded using a Hitachi F-4500 fluorescence spectrophotometer with excitation and emission wavelengths ranging from 260 to 340 nm at intervals of 5 nm and from 378 to 501 nm at intervals of 3 nm respectively. The slit width was 5/5 nm. The scan rate was 1200 nm min^{-1}. The effect of Rayleigh scattering on all response matrices was roughly reduced by subtracting the response matrix of an average blank solution from all sample response matrices. A $42 \times 17 \times 10$ three-way data array was thereby assembled. This data array was treated using the APTLD and the PARAFAC algorithms as well as the ATLD algorithm to recover the spectral profiles of each component and the OFL and NOR concentrations in the presence of interference species ENO.

All computer programs were written in Matlab and all calculations were carried out on a personal computer Pentium IV processor with 128 MB RAM under the Windows 2000 operating system.

5. RESULTS AND DISCUSSION

5.1. Choice of value of penalty factor p, q, r

The value of penalty factor p, q, r should be chosen before implementation of the APTLD algorithm. The performance of APTLD with regard to choice of penalty factor p, q, r was scrutinized. Table II reveals that choosing very small p, q and r (such as $p = q = r = 10^{-3}$ or 0) will lead to a large number of iterations and sensitivity to excess factors used in calculations for the final results of APTLD, which is similar to the situation with the PARAFAC algorithm. In addition, high variance of the quality of results has been observed with p, q, $r \leqslant 1$ and $N = 5$. On the other hand, different runs may converge to different final results even if they have the same residual sum of squares. Selecting larger p, q and r (e.g. $p = q = r = 10^4$) will make APTLD insensitive to excess factors and speed up convergence of the algorithm. A further increase in p, q and r will make APTLD perform even better according to the variance among different trials and computational burdens. However, no obvious variance of the quality of results has been observed with p, q and r varying from 10^4 to 10^{20}. In addition, the results of a number of runs indicate that the performance of APTLD is hardly improved when $p, q, r \geqslant 10^4$. In this paper we choose $p = q = r = 10^{20}$.

5.2. Implementation of APTLD, ATLD and PARAFAC

For all three-way data arrays, random initialization was implemented to start the iterative optimization procedures of APTLD and ATLD as well as PARAFAC. The optimization procedures of APTLD, ATLD and PARAFAC are terminated when the following criterion reaches a certain threshold ε($\varepsilon = 10^{-6}$ in this paper):

$$\left| \frac{\mathrm{SSR}^{(m)} - \mathrm{SSR}^{(m-1)}}{\mathrm{SSR}^{(m-1)}} \right| < \varepsilon \qquad (25)$$

Table I. Real concentrations (μg ml^{-1}) of samples 1–10

Species	1	2	3	4	5	6	7	8	9	10
OFL	0.0400	0.0000	0.0120	0.0240	0.0360	0.0480	0.0160	0.0200	0.0280	0.0400
NOR	0.0000	0.0160	0.0160	0.0120	0.0160	0.0200	0.0120	0.0200	0.0160	0.0200
ENO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0800	0.0800	0.0800	0.0800

Table II. Effect of penalty factor (p, q, r) on recoveries of concentration profiles for four species of an arbitrarily chosen one of seven simulated samples (six randomly initialized runs were performed) and iterations of APTLD[a]

		$N=4$					$N=5$				
PF	IND	1	2	3	4	IT	1	2	3	4	IT
$p=q=r=0$	Max	1.0032[b]	0.9948	1.0002	1.0254	1038					
	Min	1.0031	0.9936	0.9929	0.9950	833					
$p=q=r=10^{-3}$	Max	1.0033	0.9948	1.0001	1.0258	1054					
	Min	1.0029	0.9935	0.9928	0.9955	750					
$p=q=r=1$	Max	1.0030	0.9974	0.9925	1.0553	342	1.0029	0.9969	0.9897	1.0710	350
	Min	1.0028	0.9960	0.9856	1.0252	214	1.0006	0.8992	0.9812	0.8987	128
$p=1$, $q=r=10^4$	Max	1.0052	1.0059	0.9925	1.0131	121	1.0058	1.0060	0.9934	1.0137	291
	Min	1.0052	1.0059	0.9925	1.0131	56	1.0054	1.0059	0.9923	1.0076	59
$p=q=r=10^2$	Max	1.0041	1.0037	0.9906	1.0289	171	1.0049	1.0079	0.9900	1.0260	155
	Min	1.0041	1.0036	0.9898	1.0253	70	1.0048	1.0076	0.9890	1.0230	50
$p=q=r=10^4$	Max	1.0046	1.0067	0.9923	1.0185	55	1.0063	1.0131	0.9922	1.0135	105
	Min	1.0046	1.0067	0.9912	1.0141	36	1.0052	1.0062	0.9907	1.0095	59
$p=10^4$, $q=r=10^8$	Max	1.0046	1.0067	0.9913	1.0184	59	1.0063	1.0131	0.9911	1.0104	141
	Min	1.0046	1.0067	0.9912	1.0181	30	1.0063	1.0059	0.9908	1.0094	21
$p=q=r=10^8$	Max	1.0046	1.0067	0.9924	1.0182	52	1.0063	1.0131	0.9911	1.0200	128
	Min	1.0046	1.0067	0.9913	1.0138	30	1.0048	1.0059	0.9908	1.0106	28
$p=10^8$, $q=r=10^4$	Max	1.0046	1.0067	0.9913	1.0185	72	1.0063	1.0130	0.9922	1.0135	99
	Min	1.0046	1.0067	0.9912	1.0183	56	1.0052	1.0062	0.9908	1.0098	63
$p=q=r=10^{20}$	Max	1.0046	1.0067	0.9913	1.0184	64	1.0063	1.0131	0.9910	1.0106	79
	Min	1.0046	1.0067	0.9912	1.0181	27	1.0063	1.0130	0.9908	1.0095	51

[a]PF, IND, Max, Min and IT denote 'penalty factor', 'index', 'maximum', 'minimum' and 'iterations' respectively.
[b]1.0032 is recovery (resolved over actual concentration). For convenience of presentation, all recoveries have only four significant digits after the decimal point.

where $\mathrm{SSR}^{(m)} = \sum_{k=1}^{K} \|\mathbf{X}_{..k} - \mathbf{A}^{(m)}\mathrm{diag}(\boldsymbol{c}_{(k)})^{(m)}(\mathbf{B}^{\mathrm{T}})^{(m)}\|_F^2$, SSR is the residual sum of squares and m is the current iteration number. A maximal number of 3000 iterations was chosen to avoid unduly slow convergence.

5.3 Simulated excitation–emission matrix fluorescence data

First of all, for the proposed APTLD with $p=q=r=10^{20}$ the number of components is chosen to be four, which is the true dimensionality of the underlying model. The resolved excitation spectra, emission spectra and concentration profiles are plotted together with the actual profiles in Figures 2(A_1)–2(C_1) respectively. For comparison the PARAFAC and ATLD algorithms were also performed with the same component number. Figures 2(A_2)–2(C_2) for PARAFAC and Figures 2(A_3)–2(C_3) for ATLD display the estimated profiles together with the actual profiles in the three respective modes. These results indicate that the APTLD method performed well as well as the PARAFAC algorithm in the case where the model dimensionality was correctly chosen. However, the performance of the ATLD method was not satisfactory in resolving the concentration profile. Secondly, the simulated data were analysed with five components in order to illustrate the insensitivity to the chosen component number and the resolution accuracy of the proposed method. The resolved profiles together with the actual profiles in the three respective modes are showed in Figures 3(A_1)–3(C_1) for APTLD, Figures 3(A_2)–3(C_2) for PARAFAC and Figures 3(A_3)–3(C_3) for ATLD. These results reveal that the APTLD algorithm works well but the PARAFAC algorithm is difficult to obtain satisfactory results. Moreover, the performance of the APTLD method in resolving the concentration profile is better than that of ATLD. To elucidate further the insensitivity to the chosen component number, recoveries of concentration profiles for four species of an arbitrarily chosen one of seven simulated samples and iterations of APTLD were investigated. The results in Table III reveal that the APTLD method performed well when the number of components was increased from the actual dimensionality to larger values. Therefore one need not determine the component number accurately in the APTLD method as long as the component number is chosen greater than the possible model dimensionality. Furthermore, the convergence rate of APTLD is much faster than that of PARAFAC but slower than that of ATLD when the component number chosen is the same as the actual number of factors. The results of 10 random runs showed that the mean iteration number for APTLD (IT = 40) was much less than that for PARAFAC (IT = 1235) but greater than that for ATLD (IT = 9).

5.4. Real excitation–emission matrix fluorescence data

To resolve the actual profiles of the components, the data array of 10 mixture samples was analysed using the APTLD and PARAFAC algorithms as well as the ATLD method. The first six samples are used as concentration calibration samples and the last four are used as concentration prediction samples. At the same time the component enoxacin (ENO) is used as interference species. The concentrations of all samples are given in Table I. Figure 4 shows the resolved excitation and emission spectral profiles together with the actual profiles using APTLD with $p=q=r=10^{20}$, PARAFAC and ATLD under the condition that three factors were

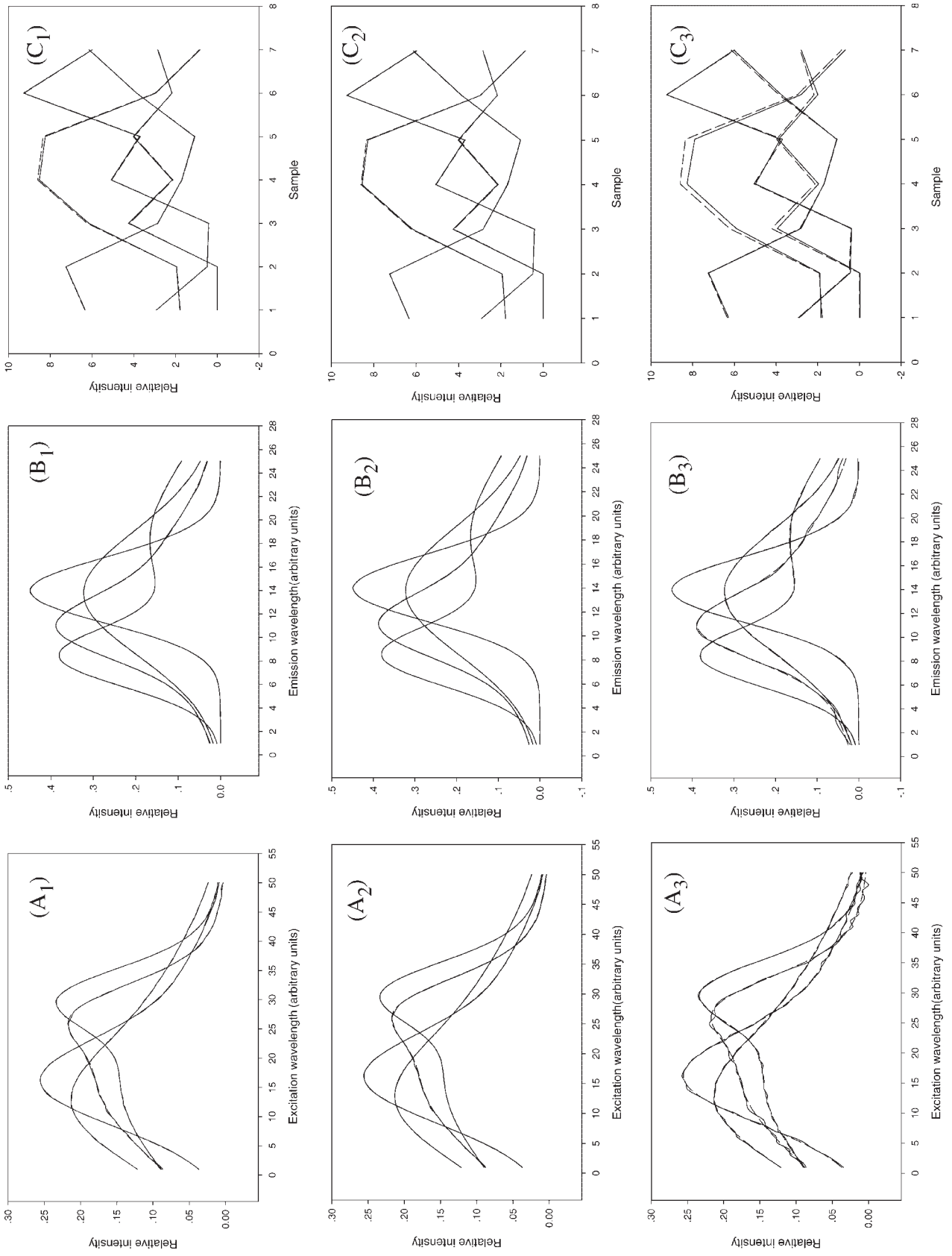

Figure 2. Resolved (full line) and actual (broken line) profiles using three algorithms on simulated data set when component number was chosen as four: (A_1)–(C_1) APTLD; (A_2)–(C_2) PARAFAC; (A_3)–(C_3) ATLD ($N = 4$).

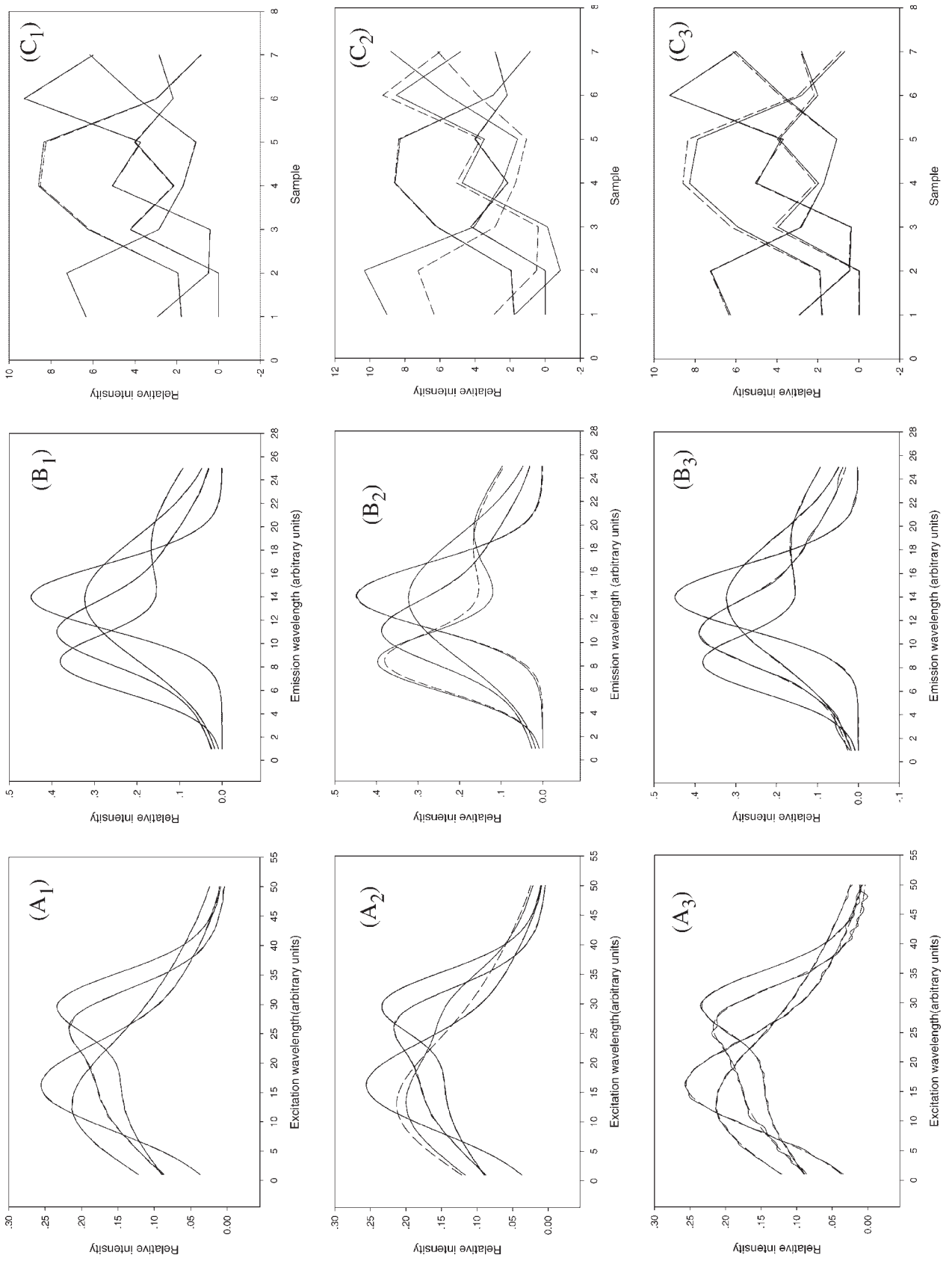

Figure 3. Resolved (full line) and actual (broken line) profiles using three algorithms on simulated data set when component number was chosen as five: (A_1)–(C_1) APTLD; (A_2)–(C_2) PARAFAC; (A_3)–(C_3) ATLD ($N = 5$).

Table III. Effect of N on recoveries of concentration profiles for four species of an arbitrarily chosen one of seven simulated samples (six randomly initialized runs were performed) and iterations of APTLD

N	1	2	3	4	Iterations
4	1.0046	1.0067	0.9912	1.0184	40
5	1.0027	1.0093	0.9903	1.0250	68
7	1.0040	1.0146	0.9881	1.0271	146
10	1.0068	1.0091	0.9914	1.0101	153
15	1.0038	1.0045	0.9887	1.0318	252
25	1.0120	1.0039	0.9762	1.0626	230

chosen, which was the same as the dimensionality of the model. Table IV displays the resolved concentrations of prediction samples using APTLD, PARAFAC and ATLD. These results indicate that all three algorithms give satisfactory resolutions for excitation and emission spectral profiles. However, the APTLD method performs a little better than the PARAFAC and ATLD algorithms in resolveing concentrations of prediction samples. Moreover, the mean iteration number for APTLD (IT = 19) was also much less than that for PARAFAC (IT = 324) but a little more than that for ATLD (IT = 9). The results reveal that the APTLD algorithm can not only resolve the profiles accurately but also overcome the

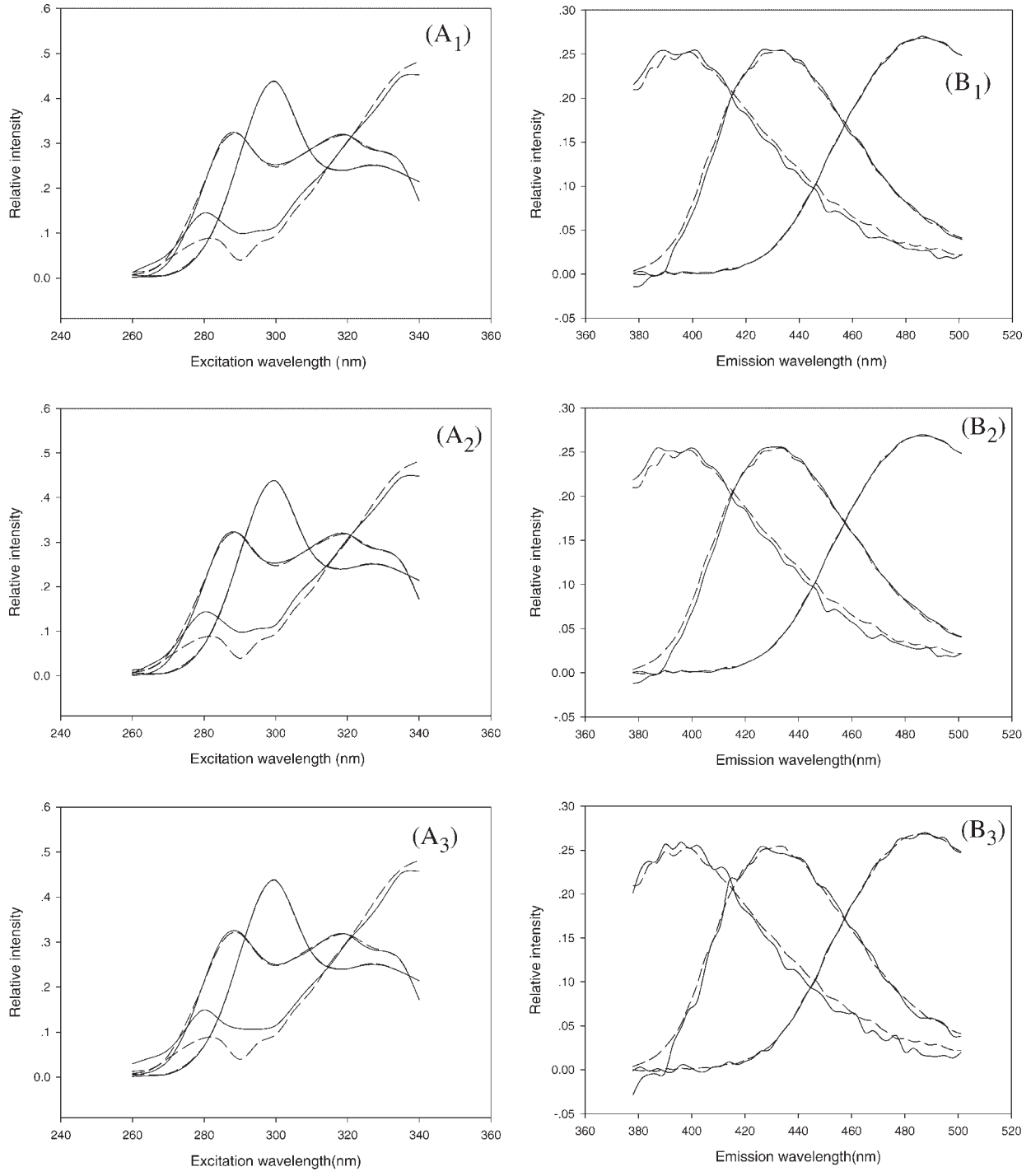

Figure 4. Resolved (full line) and actual (broken line) profiles using three algorithms on real data set when component number was chosen as four: (A_1), (B_1) APTLD; (A_2), (B_2) PARAFAC; (A_3), (B_3) ATLD ($N = 4$).

Table IV. Resolved concentrations of prediction samples using APTLD, PARAFAC and ATLD

	Real concentration		APTLD		PARAFAC		ATLD	
Sample	OFL	NOR	OFL	NOR	OFL	NOR	OFL	NOR
7	0.0160	0.0120	0.0159	0.0119	0.0159	0.0118	0.0159	0.0118
8	0.0200	0.0200	0.0201	0.0195	0.0202	0.0194	0.0202	0.0197
9	0.0280	0.0160	0.0282	0.0160	0.0283	0.0159	0.0282	0.0156
10	0.0400	0.0200	0.0404	0.0199	0.0405	0.0198	0.0403	0.0196
ED[a]			0.4356	0.5660	0.5947	0.6750	0.4243	0.6708

[a]Euclidean distance ($\times 10^{-3}$) between real concentrations and recovered concentrations profiles.

slow convergence brought on by the random initialization to some extent.

6. CONCLUSIONS

The APTLD method has been developed for the trilinear analysis of three-way data arrays. Its performance was compared with that the PARAFAC and ATLD algorithms by simulation and practical application to excitation–emission matrix fluorescence data. The results presented have shown that the proposed algorithm can simultaneously provide solutions with acceptable accuracy for all analytes present in the samples and relieve the slow convergence brought on by the random initialization. Moreover, the insensitivity to the estimated component number is of practical significance, eliminating the need for selection of the proper one. Furthermore, the performance of the APTLD method sometimes surpasses that of the PARAFAC algorithm in the prediction of concentration profiles even if the component number chosen is the same as the actual number of underlying factors in real samples. In addition, the performance of the APTLD method sometimes exceeds that of the ATLD algorithm in the prediction of concentration profiles when the component number chosen is greater than or equal to the actual number of underlying factors. Although the penalty factors (p, q and r) have some influence, the performance of APTLD is very stable as long as the value of the penalty factors is greater than 10^4.

APPENDIX: PROGRAM WRITTEN IN MATLAB FOR APTLD ALGORITHM

```
%XPI=[X_..1, X_..2,..., X_..k], size: (I * J )* K
%epsilon is the tolerance
%I is the row number
%J is the column number
%K is the channel number
%N is the estimated component number
%LFT is the loss function
%M is the iterative number
function [A, B, C, LFT, M]=APTLD (XPI,K,N,epsilon)
if nargin < 4
epsilon=10*eps*norm(XPI,1)*max(size(XPI));
end
[I, JK]=size(XPI); J=JK/K;
XIJK=reshape (XPI, I, J, K); %cut X along K direction
XJKI=shiftdim(XIJK,1); %cut X along I
XKIJ=shiftdim(XIJK,2); %cut X along J
% initialization of A & B
A=rand(I,N);B=rand(J,N);C=zeros(K,N);
TOL=10; M=0; LFT=[ ]; LF=0.01; p=10^20;q=10^20;r=10^20;
% initialization of C
CD1=0;CD2=0;aa=0;bb=0;CD3=0;CD4=0;PB=pinv(B',epsi-
lon);PA=pinv(A',epsilon);Da=diag(ones(N,1)./diag(A'*A));
Db=diag(ones(N,1)./diag(B'*B));
for i=1:I
  CD1=CD1+XJKI(:,:,i)'* PB*Da*diag(A(i,:));
  aa=aa+diag(A(i,:))*Da*diag(A(i,:));
end
for j=1:J
  CD2=CD2+XKIJ(:,:,j)*PA*Db*diag(B(j,:));
  bb=bb+diag(B(j,:))*Db*diag(B(j,:));
  CD3=CD3+XKIJ(:,:,j)*A*diag(B(j,:));
  CD4=CD4+diag(B(j,:))*A'*A*diag(B(j,:));
end
  C=(p*(CD1+CD2)+CD3)*pinv(p*aa+p*bb+CD4, epsilon);
%start to caculate LFT and do iteration
while TOL>epsilon & M<500
%estimation of B
BD1=0;BD2=0; BD3=0;BD4=0;cc=0;aa=0;PC=pinv(C',epsilon);
Dc=diag(ones(N,1)./diag(C'*C));
for k=1:K
  BD1=BD1+XIJK(:,:,k)'*PA*Dc*diag(C(k,:));
  cc=cc+diag(C(k,:))*Dc*diag(C(k,:));
end
for i=1:I
BD2=BD2+XJKI(:,:,i)*PC*Da*diag(A(i,:));
    aa=aa+diag(A(i,:))*Da*diag(A(i,:));
    BD3=BD3+XJKI(:,:,i)*C*diag(A(i,:));
    BD4=BD4+diag(A(i,:))*C'*C*diag(A(i,:));
end
B=(q*(BD1+BD2)+BD3)*pinv(q*cc+q*aa+BD4, epsilon);
%estimation of A
AD1=0;AD2=0;AD3=0;AD4=0;bb=0;cc=0;PB=pinv(B',epsilon);
Db=diag(ones(N,1)./diag(B'*B));
for j=1:J
AD1=AD1+XKIJ(:,:,j)'*PC*Db*diag(B(j,:));
    bb=bb+diag(B(j,:))*Db*diag(B(j,:));
end
for k=1:K
AD2=AD2+XIJK(:,:,k)*PB*Dc*diag(C(k,:));
  cc=cc+diag(C(k,:))*Dc*diag(C(k,:));
  AD3=AD3+XIJK(:,:,k)*B*diag(C(k,:));
  AD4=AD4+diag(C(k,:))*B'*B*diag(C(k,:));
end
A=(r*(AD1+AD2)+AD3)* pinv (r*bb+r*cc+AD4, epsilon);
% normalization of A & B
for n=1:N
```

```
    A(:,n)=A(:,n)./sqrt(sum(A(:,n).*A(:,n))); B(:,n)=B(:,n)./
      sqrt(sum(B(:,n).*B(:,n)));
end
%estimation of C
CD1=0;CD2=0;aa=0;bb=0;CD3=0;CD4=0;PA=
  pinv(A',epsilon);
Da=diag(ones(N,1)./diag(A'*A));
for i=1:I
  CD1=CD1+XJKI(:,:,i)'*PB*Da*diag(A(i,:));
  aa=aa+diag(A(i,:))*Da*diag(A(i,:));
end
for j=1:J
  CD2=CD2+XKIJ(:,:,j)*PA*Db*diag(B(j,:));
  bb=bb+diag(B(j,:))*Db*diag(B(j,:));
  CD3=CD3+XKIJ(:,:,j)*A*diag(B(j,:));
  CD4=CD4+diag(B(j,:))*A'*A*diag(B(j,:));
end
  C=(p*(CD1+CD2)+CD3)*pinv(p*aa+p*bb+CD4, epsilon);
%stopping criterion
%calculate loss function
LFTT=0;
for k=1:K
    XXX(:,:,k)=A*diag(C(k,:))*B';
end
for k=1:K
    LFTT=LFTT+trace((XIJK(:,:,k)-XXX(:,:,k))'*(XIJK(:,:,k)-
      XXX(:,:,k)));
end
    TOL=abs((LFTT-LF)/LF);
    LFT=[LFT,LFTT];
    LF=LFTT;
      M=M+1;
end
%post-processing to keep sign convention
[maxa,inda]=max(abs(A)); [maxb,indb]=max(abs(B));
asign=ones(N,1);bsign=ones(N,1);
for n=1:N
  asign(n)=sign(A(inda(n),n));
  bsign(n)=sign(B(indb(n),n));
end
A=A*diag(asign);B=B*diag(bsign);C=C*diag(asign)
  *diag(bsign);
```

Acknowledgements

The authors would like to thank the National Nature Science Foundation of China for financial support (Grant No. 20475014, 20375012 and 20435010) and the Chinese Ministry of Education for Excellent Young Teachers Program.

REFERENCES

1. Lorber A. Quantifying chemical composition from two-dimensional data arrays. *Anal. Chim. Acta* 1984; **164**: 293–297.
2. Tauler R, Kowalski BR, Fleming S. Multivariate curve resolution applied to spectral data from multiple runs of an industrial process. *Anal. Chem.* 1993; **65**: 2040–2047.
3. Ho CN, Christian GD, Davidson ER. Application of the method of rank annihilation to quantitative analyses of multicomponent fluorescence data from the video fluorometer. *Anal. Chem.* 1978; **50**: 1108–1113.
4. Bro R. Multiway calibration. Multilinear PLS. *J. Chemometrics* 1996; **10**: 47–61.
5. Booksh KS, Henshaw JM, Burgess LW, Kowalski BR. The second-order standard addition method with an application to *in situ* environmental monitors. *J. Chemometrics* 1995; **9**: 263–282.
6. Smilde AK. Comments on multilinear PLS. *J. Chemometrics* 1997; **11**: 367–377.
7. Geladi P. Analysis of multi-way (multi-mode) data. *Chemometrics Intell. Lab. Syst.* 1989; **7**: 11–25.
8. Smilde AK. Three-way analyses—problems and prospects. *Chemometrics Intell. Lab. Syst.* 1992; **15**: 143–157.
9. Sanchez E, Kowalski BR. Tensorial calibration: II. Second-order calibration. *J. Chemometrics* 1988; **2**: 265–280.
10. Wilson BE, Sanchez E, Kowalski BR. An improved algorithm for the generalized rank annihilation method. *J. Chemometrics* 1989; **3**: 493–509.
11. Sanchez E, Kowalski BR. Generalized rank annihilation factor analysis. *Anal. Chem.* 1986; **58**: 496–501.
12. Faber K, Lorber A, Kowalski BR. Generalized rank annihilation method: standard errors in the estimated eigenvalues if the instrumental errors are heteroscedastic and correlated. *J. Chemometrics* 1997; **11**: 95–109.
13. Sanchez E, Kowalski BR. Tensorial resolution: a direct trilinear decomposition. *J. Chemometrics* 1990; **4**: 29–45.
14. Booksh KS, Lin Z, Wang Z, Kowalski BR. Extension of trilinear decomposition method with an application the flow probe sensor. *Anal. Chem.* 1994; **66**: 2561–2569.
15. Gui M, Rutan SC, Agbodjan A. Kinetic detection of overlapped amino acids in thin-layer chromatography with a direct trilinear decomposition method. *Anal. Chem.* 1995; **67**: 3293–3299.
16. Gemperline PJ, Miller KH, West TL, Weinstein JE, Hamilton JG, Bray JT. Principal component analysis, trace elements, and blue crab shell disease. *Anal. Chem.* 1992; **64**: 523A–532A.
17. Smilde AK, Doornbos DA. Three-way methods for the calibration of chromatographic systems: Comparing PARAFAC and three-way PLS. *J. Chemometrics* 1991; **5**: 345.
18. Wu HL, Shibukawa M, Oguma K. An alternating trilinear decomposition algorithm with application to calibration of HPLC-DAD for simultaneous determination of overlapped chlorinated aromatic hydrocarbons. *J. Chemometrics* 1998; **12**: 1–26.
19. Mitchell BC, Burdick DS. Slowly converging PARAFAC sequences: swamps and two-factor degeneracies. *J. Chemometrics* 1994; **8**: 155–168.
20. Appellof CJ, Davidson ER. Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents. *Anal. Chem.* 1981; **53**: 2053–2056.
21. Bro R. PARAFAC: tutorial and applications. *Chemometrics Intell. Lab. Syst.* 1997; **38**: 149–171.
22. Kiers HAL, Smilde AK. Some theoretical results on second-order calibration methods for data with and without rank overlap. *J. Chemometrics* 1995; **9**: 179–195.
23. Paatero P. A weighted non-negative least squares algorithm for three-way 'PARAFAC' factor analysis. *Chemometrics Intell. Lab. Syst.* 1997; **38**: 223–242.
24. Linder M, Sundberg R. Second-order calibration: bilinear least squares regression and a simple alternative. *Chemometrics Intell. Lab. Syst.* 1998; **42**: 159–178.
25. Beltran JL, Guiteras J, Ferrer R. Three-way multivariate calibration procedures applied to high-performance liquid chromatography coupled with fast-scanning fluorescence spectrometry detection. Determination of polycyclic aromatic hydrocarbons in water samples. *Anal. Chem.* 1998; **70**: 1949–1955.
26. Harshman RA. Foundations of the PARAFAC procedure: models and conditions for an 'exploratory'

multimode factor analysis. *UCLA Working Papers in Phonetics* 1970; **16**: 1–84.
27. Chen ZP, Wu HL, Yu RQ. On the self-weighed alternating trilinear decomposition algorithm for the property of being insensitive to excess factors used in calculation. *J. Chemometrics* 2001; **15**: 439–453.
28. Chen ZP, Li Y, Yu RQ. Pseudo alternating least squares algorithm for trilinear decomposition. *J. Chemometrics* 2001; **15**: 149–167.
29. Jiang JH, Wu HL, Li Y, Yu RQ. Alternating coupled vectors resolution (ACOVER) method for trilinear analysis of three-way data. *J. Chemometrics* 1999; **13**: 557–578.
30. Faber NM, Bro R, Hopke PK. Recent developments in CANDECOMP/PARAFAC algorithms: a critical review. *Chemometrics Intell. Lab. Syst.* 2003; **65**: 119–137.
31. Bro R. Muti-way Analysis in the Food Industry, Model, Algorithms and Applications. Doctoral Thesis, University of Amsterdam, 1998.
32. Jiang JH, Wu HL, Li Y, Yu RQ. Three-way data resolution by alternating slice-wise diagonalization (ASD) method. *J. Chemometrics* 2000; **14**: 15–36.
33. Li Y, Jiang JH, Wu HL, Chen ZP, Yu RQ. Alternating coupled matrices resolution method for three-way arrays analysis. *Chemometrics Intell. Lab. Syst.* 2000; **52**:33.
34. Hergert LA, Escandar GM. Spectrofluorimetric study of the β-cyclodextrin-ibuprofen complex and determination of ibuprofen in pharmaceutical preparations and serum. *Talanta* 2003; **60**: 235–246.
35. Rayens WS, Mitchell BC. Two-factor degeneracies and a stabilization of PARAFAC. *Chemometrics Intell. Lab. Syst.* 1997; **38**: 173–181.
36. Kruskal JB, Harshman RA, Lundy ME. Rank, decomposition, and uniqueness for 3-way and N-way arrays. In *Multiway Data Analysis*, Coppi R, Bolasco S (eds). Elsevier: Amsterdam, 1989; 7–18.
37. Fleming CM, Kowalski BR. Second-order calibration and higher. In *Encyclopedia of Analytical Chemistry*, Myers RA. (ed.). Wiley and Sons: Chichester, 2000, 9736–9764.
38. Kiers HAL, Krijnen WP. An efficient algorithm for PARAFAC of three-way data with large numbers of observation units. *Psychometrika* 1991; **56**:147–152.
39. Carroll JD, Chang JJ. Analysis of individual differences in multidimensional scaling via an n-way generalization of 'Eckart-Young' decomposition. *Psychometrika* 1970; **35**: 283–319.

