The Analyst
FULL PAPER

A PARAFAC algorithm using penalty diagonalization error (PDE) for three-way data array resolution

Yu-Zhen Cao, Zeng-Ping Chen, Cui-Yun Mo, Hai-Long Wu and Ru-Qin Yu*

College of Chemistry and Chemical Engineering, Changsha, Hunan University, China.
E-mail: rqyu@hunu.edu.cn

Received 31st July 2000, Accepted 27th September 2000
First published as an Advance Article on the web 21st November 2000

A modified parallel factors analysis (PARAFAC) algorithm with the penalty diagnolization error (PDE) was developed. This algorithm can overcome the slow convergence problem of the traditional PARAFAC method and is insensitive to the number of components, *i.e.*, it is much faster than PARAFAC and insensitive to overestimation of the dimensionality of the model. The characteristic performance was demonstrated by treating simulated and real excitation–emission fluorescence data for samples of naphthalene, 1-naphthol and 2-naphthol with satisfactory results.

Introduction

Modern analytical instruments, which often generate two-way data for each sample, provide new ways of producing so-called three-way data. There are two main approaches used in three-way data analysis. One approach is based on generalized eigenanalysis, represented by the general rank annihilation method (GRAM)[1,2] and the direct trilinear decomposition (DTLD) method.[1–9] Unfortunately, in these methods, the requirement for reconstructing two pseudo samples may cause the loss of information and occasionally the appearance of annoying imaginary solutions. The other approach is an iterative one, exemplified by the parallel factors analysis (PARAFAC) and canonical decomposition (CANDECOMP),[10–25] developed and popularized by Harshman[11] and Carroll and Chang.[10] Although PARAFAC has been successfully applied to many chemical problems, there still remain several obstacles limiting its further applications.

Among others, the problem of accurate estimation of the number of underlying factors, often bringing about significant deviations from the model and the annoying slow-convergence problem, always requiring a long time to resolve the data array, are two aspects deserving detailed consideration.

Aimed at solving the slow convergence problem, several methods have been proposed to circumvent the problem. For instance, Mitchell and Burdick[15] realized that one reason leading to slow convergence was that the approach landed in a region of the solution space consisting of 'degenerate solutions'. They suggested that a new random start of the iteration is necessary when the algorithm landed in the 'swamp'. However, slow convergence is not always accompanied by near degeneracies. In particular, convergence tends to be slow in the case of the high multicollinearity, whereas degeneracies are only rarely encountered.

Another approach to handle slow convergence has already been proposed by Harshman[11] using a relaxation technique based on changing each variable individually as a function of its current short-range and long-range trends. A different procedure for accelerating convergence is of the Gauss–Newton type developed by Paatero.[20] These methods can improve the speed of the traditional PARAFAC. Bro and De Jong[26] proposed an algorithm called fast non-negative least squares (FNNLS) with a remarkable improvement of the speed of the PARAFAC and the original non-negative least squares (NNLS) itself. Another general approach for accelerating computing speed is to combine compression with the alternating least squares. For instance, Alsberg and Kvalheim,[27] Kiers and Harshman[28] and Bro and Andersson[22] utilized the method of compression to improve the computing speed of the algorithm. Recently, Kiers proposed a three-step algorithm[29] for PARAFAC analysis, which is an algorithm combining with compression. This algorithm can greatly improve the computation speed and smooth away the slow convergence caused by multicollinearity but also resolve the slow convergence problem brought about by random start through a three-step procedure. In essence, compression is a procedure of reducing the dimensionality of the three-way data array, which can be widely applied to many approaches.

We propose a modified PARAFAC algorithm with the penalty diagnolization error (PDE). This algorithm used the diagonalization error as the penalty term of the PARAFAC error, which is different from the above algorithms. The proposed algorithm can relieve the slow convergence caused by random initialization to some extent and partly overcome the slow convergence brought about by high multicollinearity. Furthermore, in order to improve further the speed of this algorithm, a compression procedure is applied under certain circumstances. In addition, the modified PARAFAC algorithm has another characteristic feature of being insensitive to the overestimation of the number of the actual factors. To put it another way, an accurate resolution could be obtained provided that the number of factors used in computation is not less than the number of actual factors in the model.

Model and algorithm

Nomenclature

Throughout this paper, scalars are represented using lower-case italics, vectors are denoted with bold lower-case characters, capitals represent two-way matrices and underlined bold capitals symbolize three-way data array. The details are as follows:

$\underline{\mathbf{X}}$: the three-way data array;
x_{ijk}: the ijkth elements of the three-way data array;
I, J, K: the dimensions of the three modes of $\underline{\mathbf{X}}$;
$\mathbf{X}_{\cdot\cdot k}$: the kth frontal slice of the three-way array $\underline{\mathbf{X}}$;
$\mathbf{X}_{\cdot j\cdot}$: the jth lateral slice of the three-way array $\underline{\mathbf{X}}$;
$\mathbf{X}_{i\cdot\cdot}$: the ith horizontal slice of the three-way array $\underline{\mathbf{X}}$.

The model

In the second-order calibration, according to the trilinear model, each element x_{ijk} of data array $\underline{\mathbf{X}}$ can be represented as the sum of a series of products:

$$x_{ijk} = \sum_{f=1}^{F} a_{if} b_{jf} c_{kf} \quad i = 1,2,3,\cdots,I;\ j = 1,2,3,\cdots,J;\ k = 1,2,3,\cdots,K \tag{1}$$

where F is the number of components and $\mathbf{a}_i$, $\mathbf{b}_j$ and $\mathbf{c}_k$ are the ith row of A, the jth row of B and the kth row of C, respectively. The trilinear model can be written in matrix form as follows:

$$X_{i\cdot\cdot} = \mathrm{B}diag(\mathbf{a}_i)C^t + E_{i\cdot\cdot} \quad i = 1,2,3,\ldots,I \tag{2}$$

$$X_{\cdot j\cdot} = Cdiag(\mathbf{b}_j)A^t + E_{\cdot j\cdot} \quad j = 1,2,3,\ldots,J \tag{3}$$

$$X_{\cdot\cdot k} = Adiag(\mathbf{c}_k)B^t + E_{\cdot\cdot k} \quad k = 1,2,3,\ldots,K \tag{4}$$

where $diag(\mathbf{a}_i)$, $diag(\mathbf{b}_j)$ and $diag(\mathbf{c}_k)$ denote the diagonal matrices of order $F \times F$. Regardless of scaling and permutation, the decomposition of the trilinear model proposed above will be a unique one provided that $k1 + k2 + k3 \geq 2F + 2$,[11–13] where $k1$, $k2$ and $k3$ 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.

The algorithm

From eqns. (2), (3) and (4), one can obtain the following three equations:

$$B^+X_{i\cdot\cdot}(C^+)^t - diag(\mathbf{a}_i) = B^+E_{i\cdot\cdot}(C^+)^t \tag{5}$$

$$C^+X_{\cdot j\cdot}(A^+)^t - diag(\mathbf{b}_j) = C^+E_{\cdot j\cdot}(A^+)^t \tag{6}$$

$$A^+X_{\cdot\cdot k}(B^+)^t - diag(\mathbf{c}_j) = A^+E_{\cdot\cdot k}(B^+)^t \tag{7}$$

where A^+, B^+ and C^+ are the pseudo inverse of the matrix A, B and C, respectively, and $B^+E_{i\cdot\cdot}(C^+)^t$, $C^+E_{\cdot j\cdot}(A^+)^t$ and $A^+E_{\cdot\cdot k}(B^+)^t$ are defined as the diagonalization error. The PDE utilizes the diagnolization error as the penalty term and combines it with the PARAFAC error to construct three objective functions. Then it decompose the model by alternately minimizing the following three objective functions:

$$F(A) = \sum_{i=1}^{I}\left[\left\|X_{i\cdot\cdot} - Bdiag(\mathbf{a}_i)C^t\right\|_F^2 + \lambda\left\|B^+X_{i\cdot\cdot}(C^+)^t - diag(\mathbf{a}_i)\right\|_F^2\right] \tag{8}$$

$$F(B) = \sum_{j=1}^{J}\left[\left\|X_{\cdot j\cdot} - Cdiag(\mathbf{b}_j)A^t\right\|_F^2 + \lambda\left\|C^+X_{\cdot j\cdot}(A^+)^t - diag(\mathbf{b}_j)\right\|_F^2\right] \tag{9}$$

$$F(C) = \sum_{k=1}^{K}\left[\left\|X_{\cdot\cdot k} - Adiag(\mathbf{c}_k)B^t\right\|_F^2 + \lambda\left\|A^+X_{\cdot\cdot k}(B^+)^t - diag(\mathbf{c}_k)\right\|_F^2\right] \tag{10}$$

where $\|\cdot\|_F^2$ denotes the Frobenius matrix norm and λ is an adjustable constant.

To put it in another way, the PDE algorithm computes loading matrix A with B and C fixed according to eqn. (8); then it uses the new A and C to update B according to eqn. (9); finally, according to eqn. (10), one computes C with the new A and the updated B.

The expressions for A, B and C can be obtained by the following reasoning.

Owing to the symmetry of the data array X, one can obtain

$$\sum_{i=1}^{I}\left\|X_{i\cdot\cdot} - Bdiag(\mathbf{a}_i)C^t\right\|_F^2 = \sum_{j=1}^{J}\left\|X_{\cdot j\cdot} - Cdiag(\mathbf{b}_j)A^t\right\|_F^2 = \sum_{k=1}^{K}\left\|X_{\cdot\cdot k} - Adiag(\mathbf{c}_k)B^t\right\|_F^2 \tag{11}$$

From eqns. (7) and (8), one can write

$$F(A) = \sum_{k}\left\|X_{\cdot\cdot k} - Adiag(\mathbf{c}_k)B^t\right\|_F^2 + \lambda\sum_{k}\left\|B^+X_{i\cdot\cdot}(C^+)^t - Adiag(\mathbf{a}_i)B^t\right\|_F^2 \tag{12}$$

From eqn. (9), it is easy to obtain the expression of matrix A using the differential equation of $F(A)$ under the circumstance of fixing matrix B and C:

Let

$$\frac{\partial F(A)}{\partial A} = 0$$

and

$$\mathbf{p}_i = diag[B^+diagX_{i\cdot\cdot}(C^+)^t] \quad i + 1, 2, 3, \ldots, I$$

$$\mathrm{P} = [\mathbf{p}_1; \mathbf{p}_2; \ldots; \mathbf{p}_I] \tag{13}$$

Then one can obtain

$$A = \left[\sum_{k} X_{\cdot\cdot k}Bdiag(\mathbf{c}_k) + \lambda P\right]\left[\lambda I + \sum_{k} diag(\mathbf{c}_k)B^tBdiag(\mathbf{c}_k)\right]^{-1} \tag{14}$$

Similarly, one can work out the other two loading matrices B and C:

$$\mathbf{q}_j = diag[C^+X_{\cdot j\cdot}(A^+)^t \quad j = 1, 2, \ldots, J)$$

$$\mathrm{Q} = [\mathbf{q}_1; \mathbf{q}_2; \ldots; \mathbf{q}_j]; \tag{15}$$

$$B = \left[\sum_{i} X_{i\cdot\cdot}Cdiag(\mathbf{a}_i) + \lambda Q\right]\left[\lambda I + \sum_{i} diag(\mathbf{a}_iC^tCdiag(\mathbf{a}_i)\right]^{-1} \tag{16}$$

$$\mathbf{r}_k = diag[A^+X_{\cdot\cdot k}(B^+)^t \quad k = 1, 2, \ldots, K$$

$$\mathrm{R} = [\mathbf{r}_1; \mathbf{r}_2; \ldots; \mathbf{r}_k]; \tag{17}$$

$$C = \left[\sum_{j} X_{\cdot j\cdot}Adiag(\mathbf{b}_j) + \lambda R\right]\left[\lambda I + \sum_{j} diag(\mathbf{b}_j)A^tAdiag(\mathbf{b}_j)\right]^{-1} \tag{18}$$

where I represents the identity matrix and -1 denotes the inverse of matrix.

Based on the brief introduction of the computation of loading matrix A, B and C described above, the PDE algorithm was designed as follows. Step 1: randomly initialize loading matrices B and C. Step 2: compute A using eqns. (13) and (14) with B and C fixed. Step 3: fix C and renew A to update B according to eqns. (15) and (16). Step 4: utilize the updated B

and A to calculate C according to eqns. (17) and (18). Step 5: update A, B and C according to Steps 2, 3 and 4 until a certain stop criterion has been reached.

Experimental

Simulated HPLC-DAD data

A typical data set obtainable using a high-performance liquid chromatographic (HPLC) system with diode array detection (DAD) on six samples of four species was simulated. The spectral profiles of the four species s1, s2, s3 and s4 were generated by:

$$s_{1,i} = 0.2gs(2i - 1,30,30) + 0.5gs(2i - 1,70,10) \quad i = 1, 2, \ldots, 50$$

$$s_{2,i} = 0.6gs(2i - 1,20,10) + 0.3gs(2i - 1,80,30) \quad i = 1, 2, \ldots, 50$$

$$s_{3,i} = 0.7gs(2i - 1,40,20) + 0.2gs(2i - 1,90,20) \quad i = 1, 2, \ldots, 50$$

$$s_{4,i} = 0.7gs(2i - 1,50,25) \quad i = 1, 2, \ldots, 50$$

where $gs(x,a,b)$ was defined as the value at x of the Gaussian function with center a and standard deviation b, namely, $gs(x,a,b) = \exp[-(x - a)^2/2b^2]$. The chromatographic profiles were simulated as the follows:

$$c_{1,j} = 0.5gs(4j - 3,40,5) \quad j = 1, 2, \ldots, 20$$

$$c_{1,j} = 0.5gs(4j - 3,30,10) \quad j = 1, 2, \ldots, 20$$

$$c_{1,j} = 0.5gs(4j - 3,50,10) \quad j = 1, 2, \ldots, 20$$

$$c_{1,j} = 0.5gs(4j - 3,40,9) \quad j = 1, 2, \ldots, 20$$

Each of the six samples contains the four species with random concentrations and the random errors obeyed a normal distribution with mean zero and standard deviation 0.005. The data array was treated utilizing both the PDE algorithm and the PARAFAC algorithm to resolve the profiles of each component in three modes.

Real excitation–emission fluorescence data

Naphthalene, 1-naphthol and 2-naphthol as possible pollutants existing in rivers and lakes were taken as object analytes for simultaneous determination in analytical samples. The steady state fluorescence of 10 samples of these three species was measured in 0.01 M NaOH solution. The concentrations of these species are shown in Table 1. The excitation wavelength was set from 220 to 300 nm with a fixed interval of 2 nm and the emission wavelength was set from 325 to 600 nm with an interval of 5 nm. The scan rate was 240 nm min^{-1}. The effect of Rayleigh scattering on each response matrix was compensated by subtracting the measurement matrix of a blank from the sample measurement. A $56 \times 41 \times 10$ three-way data array was thereby collected. This data array was analyzed using the modified PARAFAC algorithm with PDE and also the ordinary PARAFAC algorithm to recover the profiles of each component. All computing programs were written in Matlab 5.1 and run on a personal computer [Pentium processor (I)].

Results and discussion

Choice of the value of λ

Before the implementation of the PDE algorithm, the value of λ should be selected. From the objective functions of the PDE [eqns. (8), (9) and (10)] and the objective function of the traditional PARAFAC, one observes that if λ is very small, say 10^{-6}, the PDE algorithm is almost identical with the PARAFAC algorithm. In particular, if $\lambda = 0$, the PDE turns into the traditional PARAFAC. If λ is not zero, the solutions are no longer common sense least squares solutions of certain objective functions. They are the results of the balance among the three different objective functions employed. Owing to the obvious intrinsic relationships among the three objective functions, it can be expected that they would deviate only slightly from the results of PARAFAC, which is the actual least squares solution, under the conditions of moderate λ and mild noise. Furthermore, the slight bias in results will be compensated by the large increase in the convergence rate and the property of being insensitive to excess factors used in calculation. A large λ will further speed the convergence of the algorithm. However, the estimation error of the results will also increase. Therefore, a compromise between the speed and the accuracy of the algorithm should be reached. It has been shown that when $\lambda = 1$, the algorithm is capable of ensuring sufficient accuracy and overcoming the slow convergence problem.

The implementation of the PDE

For all the data arrays, the iterative optimum procedure of the PDE started from random initialization and was terminated when the following criterion reached a certain threshold ε ($\varepsilon = 1 \times 10^{-6}$):

$$\mathrm{SSR} = \sum_{k=1}^{K} \left\| X_{\cdot k} - A^{(m)} diag(C_k^{(m)}) B^{(m)t} \right\|_F^2$$

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

where m is the iteration number and SSR is the sum of the squares of the residues. A maximum iteration number of 1000 is adopted to avoid possible undue slow convergence.

Simulated HPLC-DAD data

Fig. 1 shows the recovered chromatographic and spectral profiles using the PDE with $\lambda = 1$ and using the ordinary PARAFAC under the condition that the correct number of actual factors is used. The recovered concentrations are given in Table 2. These results show that the resolved properties of PDE are as good as those of the traditional PAFAFAC when the number of the components used in computation (N) is the same as that of the actual ones (F). The correlation coefficients of the resolved spectral, chromatographic and concentration profiles are listed in Table 3. These correlation coefficients further demonstrate that the resolved profiles under two conditions are nearly the same.

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

Sample	1	2	3	4	5	6	7	8	9	10
Naphthalene	0.5135	0.0000	0.0000	0.5135	0.5135	0.0000	0.5135	0.5135	1.6405	1.0270
1-Naphthol	0.0000	1.0279	0.0000	1.0279	0.0000	1.0279	1.0279	2.002	1.0279	3.0600
2-Naphthol	0.0000	0.0000	3.3874	0.0000	3.3874	3.3874	3.3874	9.0004	6.2304	3.3874

Aiming to study the speed of the PDE algorithm, the data array was decomposed 10 times using the PDE and the ordinary PARAFAC. Because the PDE algorithm is iterative, there is a requirement for initialization. The 10 times random runs prove that the random initialization has little effect on the convergence of the PDE algorithm, which is different from that using the traditional PARAFAC algorithm. The mean iteration number (IT) of the PDE algorithm (IT = 35) is much less than that (IT = 856) of the ordinary PARAFAC algorithm. The time required to decompose the data array accurately is 7 s using the PDE compared with 53 s for the ordinary PARAFAC. The mean errors between the mean recovered profiles and the actual profiles are small (Table 4) and the mean resolved spectral and chromatographic profiles are shown in Fig. 2. The results of the 10-times iteration reveal that the PDE is much faster than the traditional PARAFAC algorithm when the model dimensionality is the same as the number of the actual factors and the results obtained using both PDE and the ordinary PARAFAC are almost the same.

In order to investigate the property of being insensitive to the overestimating factors of the PDE, the simulated data array was computed using the PDE and the traditional PARAFAC with $N = 5$. The recovered spectral, chromatographic and concentrations profiles shown in Fig. 3 indicate that using the PDE method, a random noise component is extracted from the data array and the excess noise component has little effect on obtaining the correct resolution of the actual underlying factors. In contrast, the ordinary PARAFAC algorithm can hardly provide accurate resolutions of the profiles of the actual components when the number of factors used in calculation (N) is greater than the number of actual factors (F). The excess factors are extracted in the form of noise using the PDE

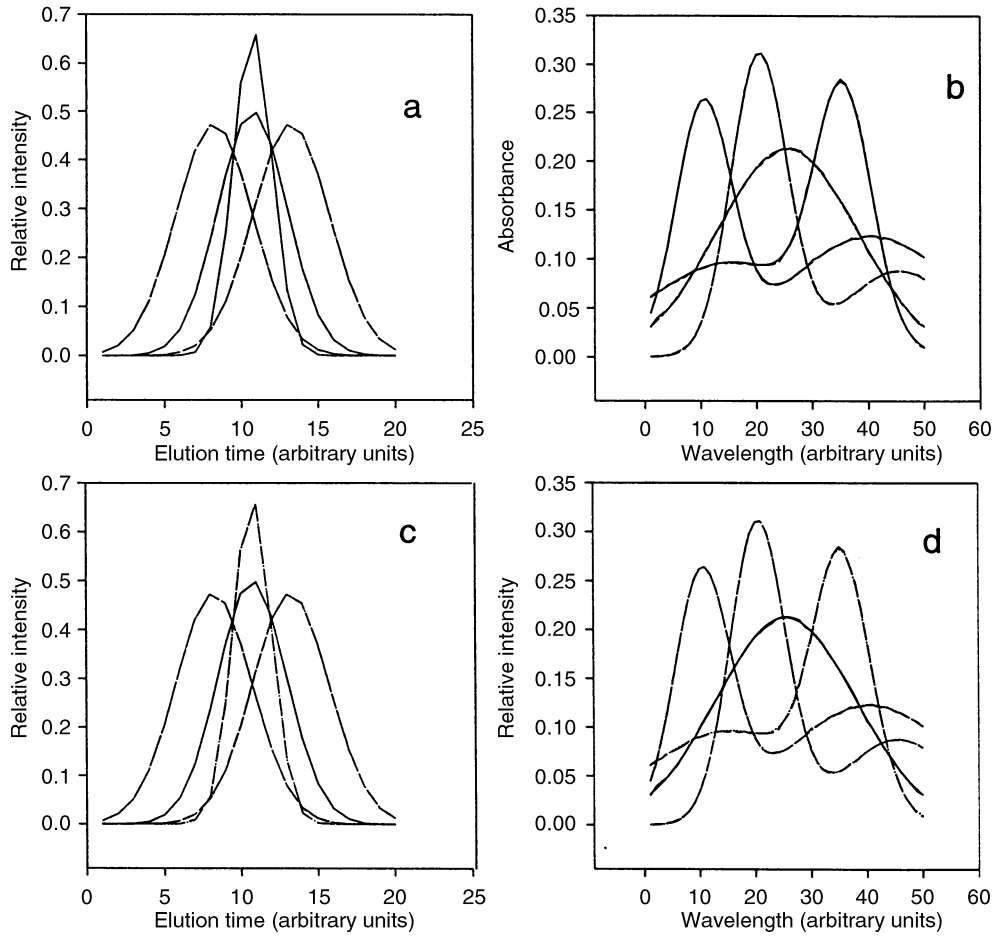

Fig. 1 Expected profiles (solid lines) and true profiles (dashed lines) using the PDE method and the traditional PARAFAC in the simulated data. (a), (b) Resolved profiles of the PDE; (c), (d) resolved profiles of the ordinary PARAFAC.

Table 2 Resolved concentrations of the six samples in the simulated data with $N = 4$ using the PDE method and the traditional PARAFAC method

	Real concentration				PDE				Traditional PARAFAC			
Sample	1	2	3	4	1	2	3	4	1	2	3	4
1	0.9501	0.4565	0.9218	0.4103	0.9346	0.4556	0.9306	0.4162	0.9480	0.4556	0.9222	0.4124
2	0.2311	0.0185	0.7382	0.8936	0.2253	0.0203	0.7506	0.8855	0.2293	0.0189	0.7390	0.8946
3	0.6068	0.8214	0.1763	0.0579	0.5959	0.8207	0.1784	0.6972	0.3020	0.8211	0.1768	0.0589
4	0.4860	0.4447	0.4057	0.3529	0.4771	0.4449	0.4114	0.3556	0.4851	0.4447	0.4063	0.3532
5	0.8913	0.6154	0.9355	0.8132	0.8746	0.6158	0.9472	0.8167	0.8889	0.6153	0.9356	0.8153
6	0.7621	0.7919	0.9169	0.0099	0.7521	0.7916	0.9254	0.0098	0.7614	0.7908	0.9175	0.0107
MSD[a]					0.0292	0.0022	0.0218	0.0150	0.0400	0.0015	0.0140	0.0340

[a] MSD is the Euclidean distance between the real concentrations and the recovered concentration profiles.

algorithm. The PDE method will not only supply accurate recovered profiles but also provide a method to estimate accurately the number of components, since the profiles of noise are easy to distinguish from the actual profiles (Fig. 3).

In order to investigate further the properties of the proposed algorithm, the error between the recovered and the actual profiles of the first component was calculated when N was changed from 2 to 11. The logarithm of error defined as the

Table 3 Correlation coefficients between the resolved files on the condition of $\lambda = 1$ and $\lambda = 0$

Mode[a]	Component 1	Component 2	Component 3	Component 4
X	1.0000	1.0000	0.9999	0.9999
Y	0.9999	0.9998	1.0000	0.9999
Z	1.0000	1.0000	0.9999	1.0000

[a] *X* denotes the chromatographic mode, *Y* the spectral mode and *Z* the concentration mode.

Table 4 Euclidean distances between the resolved profiles and the mean profiles (all values have been multiplied by 10^4)

	PDE			PARAFAC		
Component	*X*	*Y*	*Z*	*X*	*Y*	*Z*
1	0.8892	0.3550	0.2200	0.2609	0.1646	0.4012
2	0.3715	0.2998	0.0090	0.0087	0.0028	0.0011
3	0.2332	0.0950	0.0234	0.0026	0.0187	0.0175
4	0.5949	0.2040	0.0990	0.0471	0.0348	0.7246

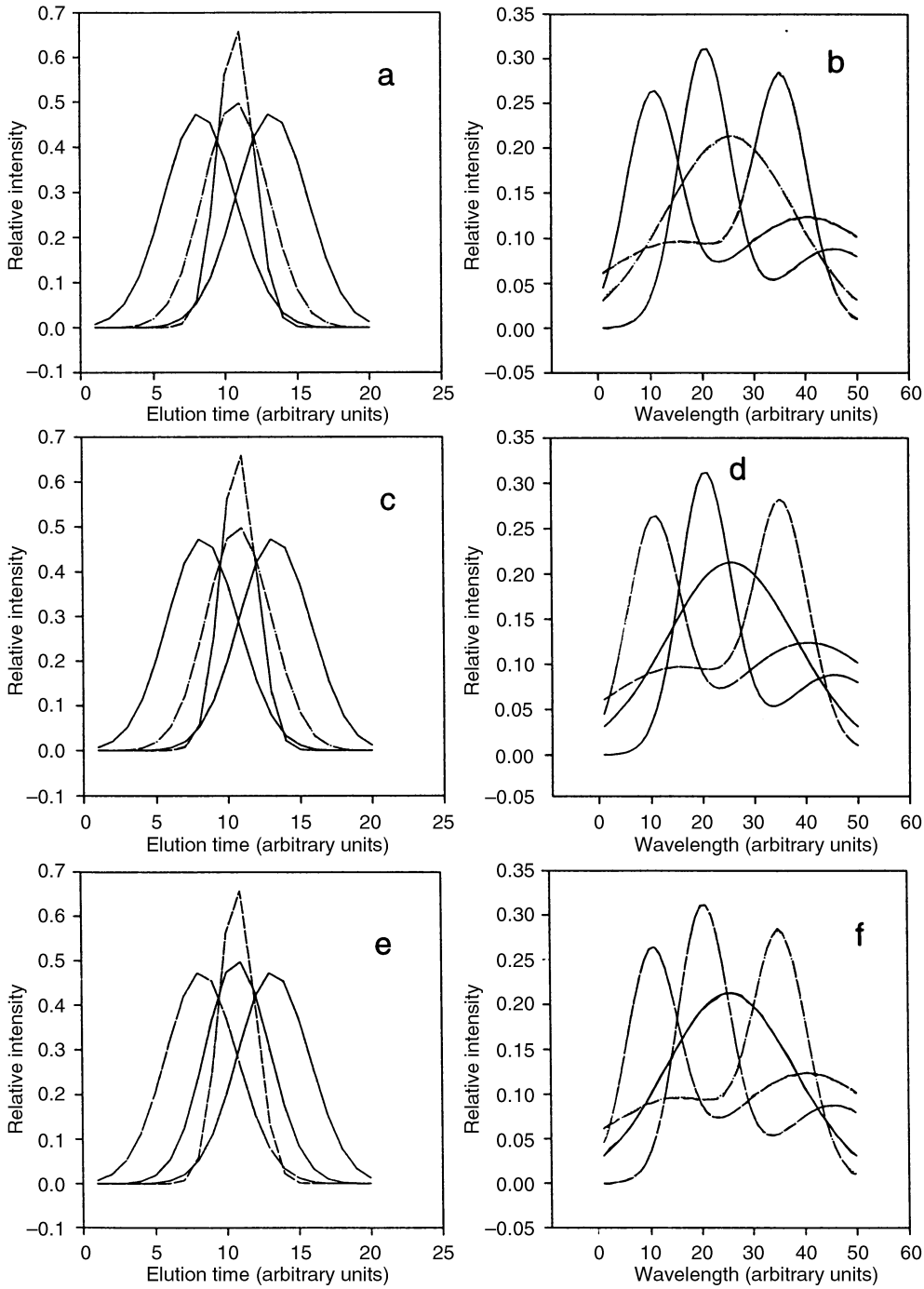

Fig. 2 Mean chromatographic and spectral profiles of 10-times random iterations using PDE and the traditional PARAFAC (solid lines) and true profiles (dashed lines) in the simulated data and the mean profiles of PDE (solid lines) and those of the ordinary PARAFAC (dashed lines). (a), (b) PDE against true; (c), (d) traditional PARAFAC against true; (e), (f) traditional PARAFAC against PDE.

Euclidean distance between the resolved and the actual profiles is plotted against N in Fig. 4. The results showed that the deviation is much larger if $N < F$ and the resolution is not accurate. When $N \geq F$, the error is much less and stable. The PDE algorithm is insensitive to the excess factors, *i.e.*, it is possible to supply accurate resolution even if $N > F$.

Real excitation–emission fluorescence data of naphthalene and 1- and 2-naphthol

To resolve the actual profiles of the components, the data array was analysed using the PDE and the ordinary PARAFAC. Three factors were used in computation which was the same as the dimensionality of the model. The concentration profiles are listed in Table 5. The results reveal that if $N = 3$, the dimensionality of the actual model, both the PDE and the traditional PARAFAC, can give accurate resolutions of the underlying factors. In addition, the resolved profiles are nearly the same. However, the iteration number using the PDE algorithm is only 12 and is independent of the random starting values. The iteration number using the PARAFAC is 167 and is closely related to the random initialized value. The results demonstrate that the PDE algorithm can not only resolve the profiles accurately but also overcome the slow convergence caused by the random initialization to some extent.

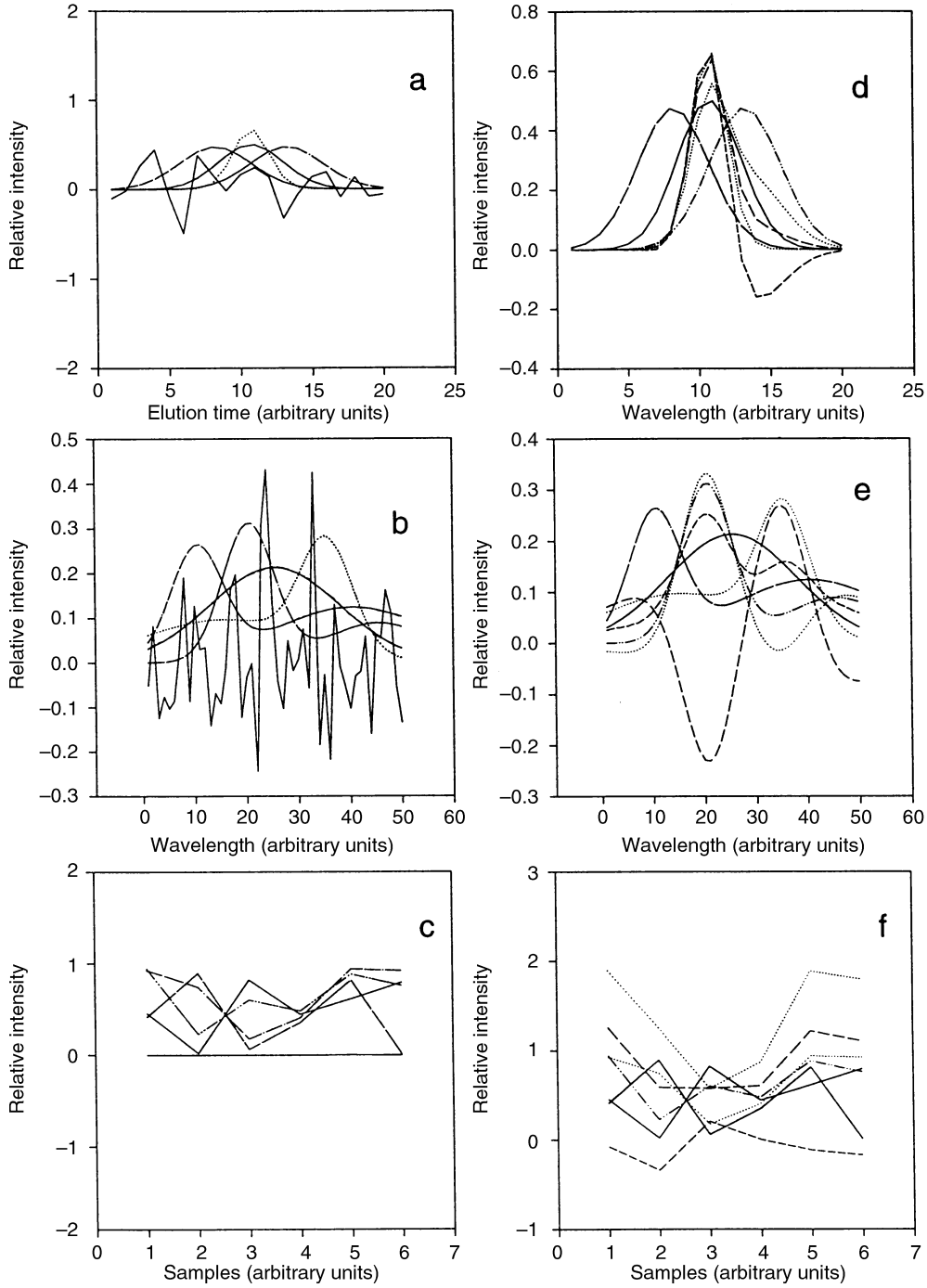

Fig. 3 Expected profiles (solid lines) and actual profiles (dotted line) using (a)–(c) PDE and (d)–(f) traditional PARAFAC with $N = 5$ in the simulated data.

Further improvement of the speed of the PDE algorithm

On condition that the number of the underlying factors can be estimated accurately, an improvement is proposed below to speed up the PDE algorithm further.

First, one computes the common columns space (U) and the common rows space (V), using the singular value decomposition of the following two matrices:

$$R1 = \sum_k X_{\cdot\cdot k}{}^t X_{\cdot\cdot k}$$

$$R2 = \sum_k X_{\cdot\cdot k} X_{\cdot\cdot k}{}^t$$

Then one projects the k frontal slices on to the U, V space, *i.e.*, $X_{\cdot\cdot k} = U^t X_{\cdot\cdot k} V$. After the orthonormal projection, matrices $X_{\cdot\cdot k}$ (k = 1, 2, 3, …, K) have become ($F \times F$) matrices. Finally, one uses the above steps of the PDE algorithm to resolve the new data array to recover the true profiles.

After such a treatment, the iteration number of the PDE is only slightly changed. The essence of the improvement is a compression treatment in the case of principal component analysis, which can not only reduce the time of each iteration of the PDE but also smooth away parts of the noise.

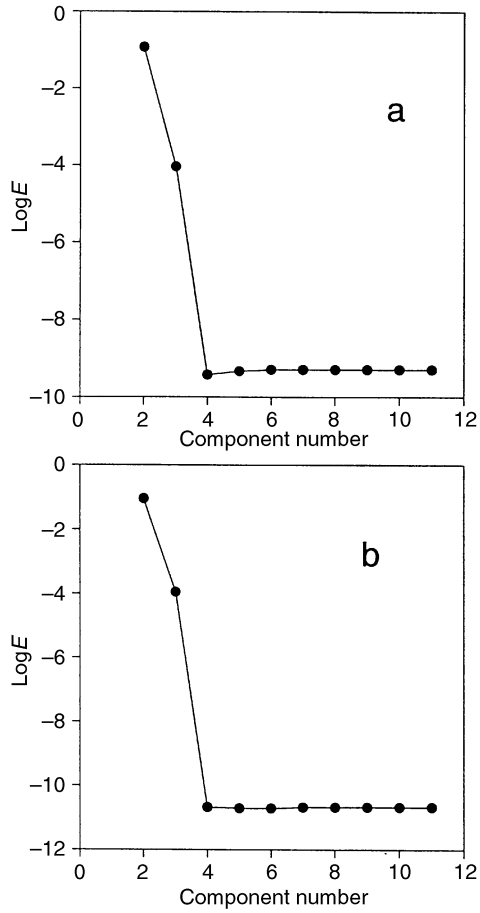

Fig. 4 Logarithm of the error between the first recovered component and the actual component against component number. (a) Error of the first chromatographic profiles; (b) Error of the spectral profiles.

Conclusions

The modified version of the PARAFAC algorithm using PDE for three-way data array resolution described in this paper can overcome some of the difficulties encountered when the ordinary PARAFAC algorithm is utilized. The proposed algorithm can relieve the slow convergence caused by the random initialization, which is inevitable while using the traditional PARAFAC algorithm. Moreover, it is insensitive to the overestimation of the dimensionality of the model, *i.e.*, it can supply accurate resolution of the profiles of the actual factors provided that the number of factors used in the calculation is not less than the number of actual underlying factors.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (Grant No. 29735150).

Table 5 Resolved concentrations using the PDE and the traditional PARAFAC

	Real concentration/μg ml^{-1}			Traditional PARAFAC/μg ml^{-1}			PDE/μg ml^{-1}		
Sample	A	B	C	A	B	C	A	B	C
1	0.5135	0.0000	0.0000	0.4213	−0.1038	−0.0796	0.421	−0.1041	−0.0801
2	0.0000	1.0279	0.0000	−0.0924	0.9814	−0.0536	−0.0930	0.9831	−0.0527
3	0.0000	0.0000	3.3874	−0.0751	−0.1241	3.3172	−0.0749	−0.1232	3.3169
4	0.5135	1.0279	0.0000	0.5338	1.0682	0.0144	0.5335	1.0688	0.0131
5	0.5135	0.0000	3.3874	0.5249	−0.0707	3.4425	0.5250	−0.0683	3.4402
6	0.0000	1.0279	3.3874	−0.0627	0.9260	3.4274	−0.0628	0.9257	3.4268
7	0.5135	1.0279	3.3874	0.50610	0.9522	3.3612	0.5062	0.9545	3.3589
8	0.5135	2.0002	9.0004	0.5814	1.8908	8.6864	0.5834	1.8968	8.6816
9	1.6405	1.0279	6.2304	1.6616	1.2119	6.1908	1.6630	1.2183	6.1838
10	1.0270	3.0600	3.3874	1.0560	2.9141	3.3754	1.0561	2.9177	3.7120
MAE[a]				0.0478	0.0877	0.0720	0.0482	0.0889	0.0713

[a] MAE is defined as the mean absolute error between the expected profiles and the actual profiles. A naphthalene; B 1−naphthnol; C 2−naphthnol.

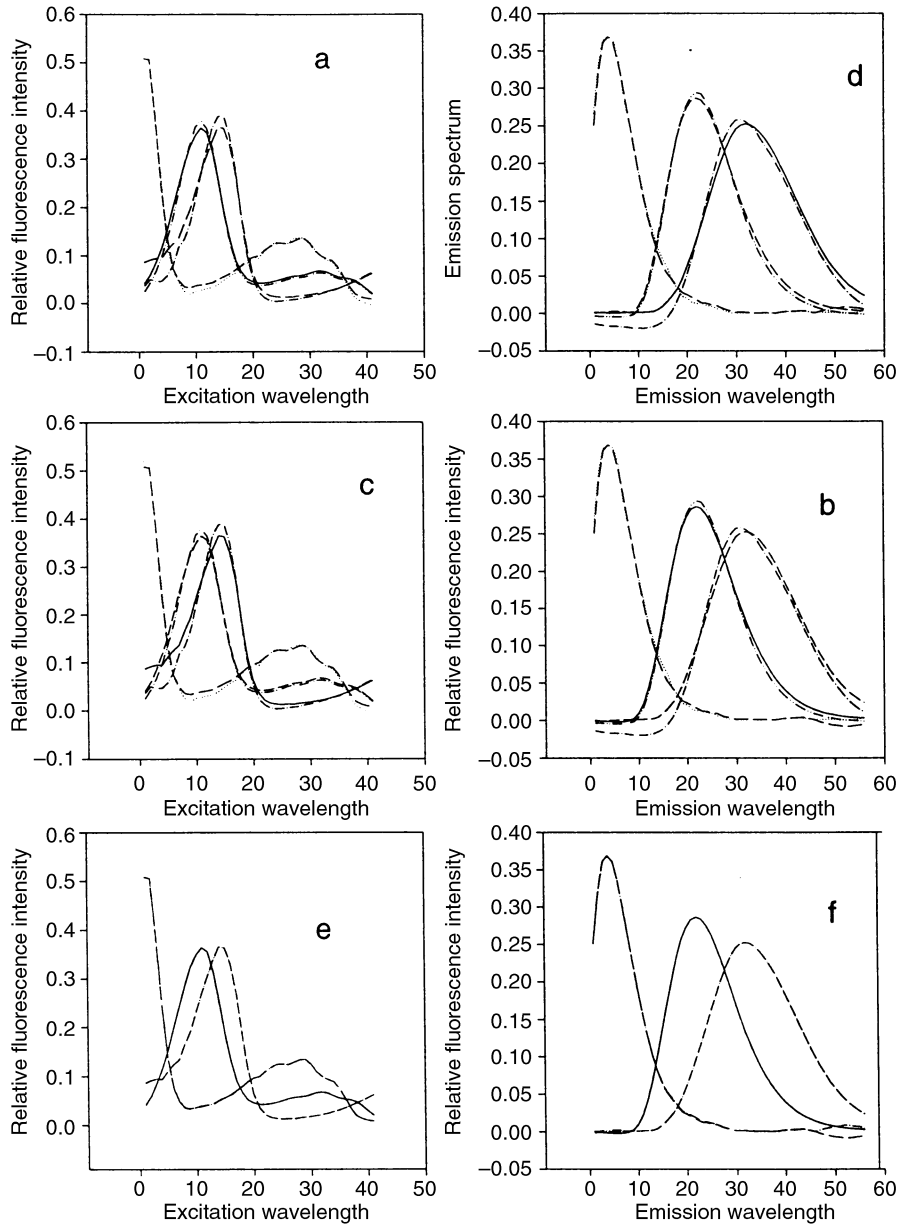

Fig. 5 Recovered profiles (solid lines) of the PDE and the traditional PARAFAC and the actual profiles (dashed lines) and the profiles of the PDE (solid lines) and those of the ordinary PARAFAC. (a), (b) PDE against true; (c), (d) traditional PARAFAC against true; (e), (f) PDE against the ordinary PARAFAC. The abscissa of (a), (c) and (e) are excitation wavelength from 220 to 300 nm with an interval of 2 nm, the abscissa of (b), (d) and (f) are emission wavelength from 325 to 600 nm with an interval of 5 nm.

References

1 B. Grung and O. M. Kvalheim, *Chemom. Intell. Lab. Syst.*, 1995, **29**, 213.
2 D. W. Millican and L. B. McGown, *Anal. Chem.*, 1990, **62**, 2242.
3 S. E. Leurgans, R. T. Ross and R. B. Abal, *SIAM J. Matrix. Anal. Appl.*, 1993, **14**, 1064.
4 A. D. Juan, S. C. Rutan, R. Tauler and D. L. Massart , *Chemom. Intell. Lab. Syst.*, 1998, **40**, 19.
5 E. Sanchez and B. R. Kowalski, *Anal. Chem.*, 1986, **58**, 499.
6 B. Wilson, E. Sanchez and B. R. Kowalski, *J. Chemom.*, 1989, **3**, 493.
7 E. Sanchez and B. R. Kowalski, *J. Chemom.*, 1990, **4**, 29.
8 S. Li, C. Hamilton and P. Gemperline, *Anal. Chem.*, 1992, **64**, 599.
9 K. S. Booksh, Z. Lin , Z. Wang and B. R. Kowalski, *Anal. Chem.*, 1994, **66**, 2561.
10 J. D. Carroll and J. J. Chang, *Psychometrics*, 1970, **35**, 283.
11 R. A. Harshman, *UCLA Working Pap. Phonet.*, 1970, **16**, 1.
12 C. J. Appellof and E. R. Davidson, *Anal. Chem.*, 1981, **53**, 2053.
13 P. Geladi, *Chemom. Intell. Lab. Syst.*, 1989, **7**, 11.
14 A. K. Smilde and D. A. Doornbos, *J. Chemom.*, 1991, **5**, 345.
15 B. C. Mitchell and D. S. Burdick, *J. Chemom.*, 1994, **8**, 155.
16 R. Bro, *Chemom. Intell. Lab. Syst.*, 1997, **38**, 149.
17 J. B. Kruskal, *Linear Algebra Appl.*, 1977, **18**, 95.
18 H. A. L. Kiers and A. K. Smilde, *J. Chemom.*, 1995, **9**, 179.
19 W. P. Krijnen, *The Analysis of Three-way Arrays by Constrained PARAFAC Methods*, DSWO Press, Leiden, 1993.
20 P. Paatero, *Chemom. Intell. Lab. Syst.*, 1997, **38**, 223.
21 H. L. Wu, M. Shibukawa and K. Oguma, *J. Chemom.*, 1998, **12**, 1.
22 R. Bro and C. A. Andersson, *Chemom. Intell. Lab. Syst.*, 1998, **42**, 105.
23 M. Linder and R. Sundberg, *Chemom. Intell. Lab. Syst.*, 1998, **42**, 159.
24 P. K. Hopke, P. Paatero, H. Jia, R. T. Ross and R. A. Harshman, *Chemom. Intell. Lab. Syst.*, 1998, **43**, 25.
25 J. L. Beltran, J. Guiteras and R. Ferrer, *Anal. Chem.*, 1998, **70**, 1949.
26 R. Bro and S. De Jong, *J. Chemom.*, 1997, **11**, 393.
27 B. K. Alsberg and O. M. Kvalheim, *Chemom. Intell. Lab. Syst.*, 1994, **24**, 43.
28 H. A. L. Kiers and R. A. Harshman, *Chemom. Intell. Lab. Syst.*, 1997, **36**, 31.
29 H. A. L. Kiers, *J. Chemom.*, 1998, **12**, 155.