Nonnegative 3-way tensor factorization via conjugate gradient with globally optimal stepsize

Jean-Philip Royer, Pierre Comon, Nadège Thirion-Moreau

NONNEGATIVE 3-WAY TENSOR FACTORIZATION VIA CONJUGATE GRADIENT WITH GLOBALLY OPTIMAL STEPSIZE

Jean-Philip Royer, Pierre Comon[*]

I3S, Algorithmes/Euclide-B,
2000 route des Lucioles, BP 121, F-06903,
Sophia Antipolis Cedex,
Tel/Fax: +33 4 9294 2717/2896

Nadège Thirion-Moreau

ISITV, LSEET, UMR CNRS 6017,
Université du Sud Toulon Var,
F-83957, La Garde Cédex, France,
Tel/Fax: +33 4 9414 2456/2671

ABSTRACT

This paper deals with the minimal polyadic decomposition (also known as canonical decomposition or Parafac) of a 3-way array, assuming each entry is positive. In this case, the low-rank approximation problem becomes well-posed. The suggested approach consists of taking into account the nonnegative nature of the loading matrices directly in the problem parameterization. Then, the three gradient components are derived allowing to efficiently implement the decomposition using classical optimization algorithms. In our case, we focus on the conjugate gradient algorithm, well matched to large problems. The good behaviour of the proposed approach is illustrated through computer simulations in the context of data analysis and compared to other existing approaches.

Index Terms— Data analysis, Non linear conjugate gradient, Quasi-Newton, nonnegative, 3-way array, tensor factorization, multi-way data analysis, Canonical Polyadic decomposition, CanDecomp, Parafac, Chemometrics, Data mining, Data analysis

1. INTRODUCTION

The polyadic decomposition of a tensor has a wide panel of applications. In some cases, it is necessary to consider positive tensors, for example when we deal with images as it is the case in the hyperspectral or chemometrics fields. Thus, the loading matrices have to be constrained to be positive. One advantage is that the problem of the decomposition becomes well-posed. In this paper, we present numerical algorithms that take into account this constraint focusing on the conjugate gradient algorithm; Quasi-Newton (BFGS and DFP) and gradient algorithms have also been studied but will not be presented in details here. Regarding the choice of the stepsize, two strategies are studied based either on a global search via enhanced line search (ELS) or on backtracking alternating with ELS.

After recalling some basic notions about third order tensor decomposition in section 2, we focus on nonnegative tensors in Section 3. The suggested algorithm is presented in Section 4. In section 5, computer simulations are performed in the context of data analysis. Finally, we summarize the advantages enjoyed by the suggested approach.

[*]This work has been supported by a "PRES euro-méditerranéen" grant.

2. PROBLEM STATEMENT

A tensor is an object defined on a product between linear spaces. Once the bases of these spaces are fixed, a third order tensor can be represented by a three-way array (or a hypermatrix). The order of a tensor hence corresponds to the number of indices of the associated array. One also talks about the number of *ways* or *modes* [1]. In this paper, due to the considered applications, including fluorescence spectroscopy [2][1] and hyperspectral imaging [3], we focus on real positive 3-way arrays denoted by $\mathbf{T} = (t_{ijk}) \in \mathbb{R}^{I\times J\times K}$, admitting the following trilinear decomposition: $\mathbf{T} = \sum_{f=1}^{F} \mathbf{a}_f \circledast \mathbf{b}_f \circledast \mathbf{c}_f$, also known as a triadic decomposition [4] of $\mathbf{T}$, where the three involved matrices $\mathbf{A} = (a_{if}) = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_F] \in \mathbb{R}^{I\times F}$, $\mathbf{B} = (b_{jf}) = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_F] \in \mathbb{R}^{J\times F}$, $\mathbf{C} = (c_{kf}) = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_F] \in \mathbb{R}^{K\times F}$ are the so-called *loading matrices*, whose columns are the *loading factors*, F is a sufficiently large integer and $\circledast$ stands for the outer product. Equivalently, $\forall i = 1, \ldots, I$, $\forall j = 1, \ldots, J$ and $\forall k = 1, \ldots, K$, the relation between array entries is given by

$$t_{ijk} = \sum_{f=1}^{F} a_{if} b_{jf} c_{kf}. \qquad (1)$$

The smallest integer F that can be found such that the equality above holds exactly is called the *tensor rank* [5]. For this value of F, the above decomposition is called the Canonical Polyadic decomposition (CP) of tensor $\mathbf{T}$. Note that this acronym may also stand for CanDecomp/Parafac, if some readers prefer [6]. The aforementionned model can be written

in a compact form using the Khatri-Rao product $\odot$, as

$$\begin{aligned}\mathbf{T}_{(1)}^{I,JK} &= \mathbf{A}(\mathbf{C}\odot\mathbf{B})^T,\\ \mathbf{T}_{(2)}^{J,KI} &= \mathbf{B}(\mathbf{C}\odot\mathbf{A})^T,\\ \mathbf{T}_{(3)}^{K,JI} &= \mathbf{C}(\mathbf{B}\odot\mathbf{A})^T,\end{aligned} \tag{2}$$

where $\mathbf{T}_{(1)}^{I,JK}$ (resp. $\mathbf{T}_{(2)}^{J,KI}$ and $\mathbf{T}_{(3)}^{K,JI}$) is the matrix of size $I \times JK$ (resp. $J \times KI$ and $K \times JI$) obtained by unfolding the array $\mathbf{T}$ of size $I \times J \times K$ in the first mode (resp. second and third mode).

2.1. CP decomposition of 3-way tensors

Considering third-order tensors and assuming that F is known or overestimated, the goal of CP decomposition is to estimate the three loading matrices $\mathbf{A} \in \mathbb{R}^{I\times F}$, $\mathbf{B} \in \mathbb{R}^{J\times F}$ and $\mathbf{C} \in \mathbb{R}^{K\times F}$. To achieve this task, the quadratic error between data and model has to be minimized with respect to the three loading matrices, i.e. generally the following cost function is considered:

$$\begin{aligned}&\mathcal{F}(\mathbf{A},\mathbf{B},\mathbf{C}) = \|\mathbf{T}_{(1)}^{I,JK} - \mathbf{A}(\mathbf{C}\odot\mathbf{B})^T\|_F^2\\ &= \|\mathbf{T}_{(2)}^{J,KI} - \mathbf{B}(\mathbf{C}\odot\mathbf{A})^T\|_F^2 = \|\mathbf{T}_{(3)}^{K,JI} - \mathbf{C}(\mathbf{B}\odot\mathbf{A})^T\|_F^2,\end{aligned} \tag{3}$$

where $\|\cdot\|_F$ stands for the Frobenius norm. In the problem of the polyadic canonical decomposition of the 3-way tensor $\mathbf{T}$, its rank F has to be estimated too.

3. NONNEGATIVE 3-WAY ARRAYS FACTORIZATION

If one imposes all loading matrices to be nonnegative, the infimum of (3) is reached and problem is well posed [7]. It is then referred to as the Nonnegative Tensor Factorization problem (NTF).

3.1. Loading matrices parameterization

Approaches have already been developed to solve the NTF. However, instead of modifying the form of the cost function by adding regularization terms like in [8], we constraint the loading matrices to have positive entries via their parameterization. In order to express the fact that a matrix, say $\mathbf{A}$, has nonnegative entries, we simply write them as squares, that is $a'_{ij} = a_{ij}^2$. Using the Hadamard entry-wise product, this leads to: $\mathbf{A}' = \mathbf{A}\boxdot\mathbf{A}$. The cost function can be rewritten as:

$$\begin{aligned}&\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C}) = \mathcal{F}(\mathbf{A}\boxdot\mathbf{A},\mathbf{B}\boxdot\mathbf{B},\mathbf{C}\boxdot\mathbf{C})\\ &= \|\mathbf{T}_{(1)}^{I,JK} - (\mathbf{A}\boxdot\mathbf{A})\left[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{B}\boxdot\mathbf{B})\right]^T\|_F^2 = \|\boldsymbol{\delta}_{(1)}\|_F^2\\ &= \|\mathbf{T}_{(2)}^{J,KI} - (\mathbf{B}\boxdot\mathbf{B})\left[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{A}\boxdot\mathbf{A})\right]^T\|_F^2 = \|\boldsymbol{\delta}_{(2)}\|_F^2\\ &= \|\mathbf{T}_{(3)}^{K,JI} - (\mathbf{C}\boxdot\mathbf{C})\left[(\mathbf{B}\boxdot\mathbf{B})\odot(\mathbf{A}\boxdot\mathbf{A})\right]^T\|_F^2 = \|\boldsymbol{\delta}_{(3)}\|_F^2\end{aligned} \tag{4}$$

Using the differential $\mathsf{d}\mathcal{H}$ of $\mathcal{H}$, will allow us to calculate the gradient components ($I\times F$ matrix $\nabla_{\mathbf{A}}\mathcal{H}$, $J\times F$ matrix $\nabla_{\mathbf{B}}\mathcal{H}$ and $K \times F$ matrix $\nabla_{\mathbf{C}}\mathcal{H}$). With this goal, recall that the Frobenius scalar product $\langle\mathbf{A},\mathbf{B}\rangle = \mathsf{trace}\{\mathbf{A}^T\mathbf{B}\}$. We also have: $\langle\mathbf{A},\mathbf{A}\rangle = \|\mathbf{A}\|_F^2 = \mathsf{trace}\{\mathbf{A}^T\mathbf{A}\}$. As a consequence, the cost function $\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C})$ can be rewritten, in the first mode for example:

$$\begin{aligned}&\langle\boldsymbol{\delta}_{(1)},\boldsymbol{\delta}_{(1)}\rangle = \mathsf{trace}\left\{\boldsymbol{\delta}_{(1)}^T\boldsymbol{\delta}_{(1)}\right\}\\ &= \mathsf{trace}\left\{\left(\mathbf{T}_{(1)}^{I,JK} - (\mathbf{A}\boxdot\mathbf{A})[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{B}\boxdot\mathbf{B})]^T\right)^T\cdot\right.\\ &\left.\left(\mathbf{T}_{(1)}^{I,JK} - (\mathbf{A}\boxdot\mathbf{A})[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{B}\boxdot\mathbf{B})]^T\right)\right\}.\end{aligned}$$

The calculation of $\mathsf{d}\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C})$ is performed in Appendix A. It is shown to be equal to:

$$\begin{aligned}&\mathsf{d}\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C})\\ &= \langle 4\left[\mathbf{A}\boxdot\left((-\boldsymbol{\delta}_{(1)})\left[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{B}\boxdot\mathbf{B})\right]\right)\right],\mathsf{d}\mathbf{A}\rangle\\ &+ \langle 4\left[\mathbf{B}\boxdot\left((-\boldsymbol{\delta}_{(2)})\left[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{A}\boxdot\mathbf{A})\right]\right)\right],\mathsf{d}\mathbf{B}\rangle\\ &+ \langle 4\left[\mathbf{C}\boxdot\left((-\boldsymbol{\delta}_{(3)})\left[(\mathbf{B}\boxdot\mathbf{B})\odot(\mathbf{A}\boxdot\mathbf{A})\right]\right)\right],\mathsf{d}\mathbf{C}\rangle.\end{aligned} \tag{5}$$

3.2. Gradient matrices

Using (5), the three gradient components $\nabla_{\mathbf{A}}\mathcal{H}$, $\nabla_{\mathbf{B}}\mathcal{H}$ and $\nabla_{\mathbf{C}}\mathcal{H}$ can be derived. We define the gradient respect to $\mathbf{X}$ of a function $\mathcal{I}$ of matrix variables, as the matrix $\nabla_{\mathbf{X}}\mathcal{I}(\mathbf{X},\mathbf{Y},\mathbf{Z},...)$ with entries $\frac{\partial\mathcal{I}(\mathbf{X},\mathbf{Y},\mathbf{Z},...)}{\partial X_{ij}}$. Thus, the gradient components are found to be equal to:

$$\begin{aligned}\nabla_{\mathbf{A}}\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C}) &= 4\mathbf{A}\boxdot\left((-\boldsymbol{\delta}_{(1)})\left[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{B}\boxdot\mathbf{B})\right]\right),\\ \nabla_{\mathbf{B}}\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C}) &= 4\mathbf{B}\boxdot\left((-\boldsymbol{\delta}_{(2)})[(\mathbf{C}\boxdot\mathbf{C})\odot(\mathbf{A}\boxdot\mathbf{A})]\right),\\ \nabla_{\mathbf{C}}\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{C}) &= 4\mathbf{C}\boxdot\left((-\boldsymbol{\delta}_{(3)})[(\mathbf{B}\boxdot\mathbf{B})\odot(\mathbf{A}\boxdot\mathbf{A})]\right).\end{aligned} \tag{6}$$

Using (6), we can build the two following $(I+J+K)F\times 1$ vectors:

$$\begin{aligned}\mathbf{g}^{(k)} &= \begin{pmatrix}\mathsf{vec}\{\nabla_{\mathbf{A}}\mathcal{H}(\mathbf{A}^{(k)},\mathbf{B}^{(k)},\mathbf{C}^{(k)})\}\\ \mathsf{vec}\{\nabla_{\mathbf{B}}\mathcal{H}(\mathbf{A}^{(k)},\mathbf{B}^{(k)},\mathbf{C}^{(k)})\}\\ \mathsf{vec}\{\nabla_{\mathbf{C}}\mathcal{H}(\mathbf{A}^{(k)},\mathbf{B}^{(k)},\mathbf{C}^{(k)})\}\end{pmatrix},\\ \mathbf{x}^{(k)} &= \begin{pmatrix}\mathsf{vec}\{\mathbf{A}^{(k)}\}\\ \mathsf{vec}\{\mathbf{B}^{(k)}\}\\ \mathsf{vec}\{\mathbf{C}^{(k)}\}\end{pmatrix}\end{aligned} \tag{7}$$

where the $\mathsf{vec}\{\cdot\}$ operator applied on a given matrix stacks its columns into a column vector.

4. NON LINEAR CONJUGATE GRADIENT

To estimate the three loading matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, $\mathcal{H}$ given in (4) has to be minimized with respect to those three matrices. To that aim, we suggest to use the non linear conjugate

gradient algorithm, in which the vector $\mathbf{x}$ (given in (7)) is updated at each step k ($k = 1, 2...$) according to the following adaptation rule:

$$\left\{ \begin{array}{ll} \mathbf{x}^{(k+1)} & = \mathbf{x}^{(k)} + \mu^{(k)} \mathbf{d}^{(k)} \\ \mathbf{d}^{(k+1)} & = -\mathbf{g}^{(k+1)} + \beta^{(k)} \mathbf{d}^{(k)}. \end{array} \right. \quad (8)$$

Two expressions of β are classically used: the Fletcher-Reeves (β_{FR}) and the Polak-Ribière (β_{PR}) formula [9]: $\beta_{\mathsf{FR}}^{(k+1)} = \frac{\mathbf{g}^{(k+1)^T}\mathbf{g}^{(k+1)}}{\mathbf{g}^{(k)^T}\mathbf{g}^{(k)}}$, $\beta_{\mathsf{PR}}^{(k+1)} = \frac{\mathbf{g}^{(k+1)^T}(\mathbf{g}^{(k+1)}-\mathbf{g}^{(k)})}{\mathbf{g}^{(k)^T}\mathbf{g}^{(k)}}$. As noticed in [9], if a conjugate gradient method is reinitialized, from time to time, by setting $\mathbf{d}^{(i)} = -\mathbf{g}^{(i)}$, we might get better performance; that is why in our case we have chosen to perform this "restart" every $(I+J+K)F$ iterations. With regard to the choice of step size $\mu^{(k)}$, several approaches can be envisaged. We can use either a fixed stepsize, which leads to the poorest results, or a global search denoted by ELS (which amounts, in this specific case, to the calculation of the real and positive roots of a 11-th order polynomial), or else a locally optimal stepsize using a backtracking method as described in [10]. Those two alternatives are compared in Fig. 3.

5. COMPUTER SIMULATIONS

Simulations are now provided in the context of data analysis (more precisely in chemometrics with fluorescence images).

5.1. Application field

Excitation of dissolved organic matter by an excitation wavelength leads to several effects: Raman and Rayleigh diffusion and fluorescence. The following relation holds: $I(\lambda_f, \lambda_e, k) = I_o\, \gamma(\lambda_f)\, \epsilon(\lambda_e)\, c_k$, where λ_f and λ_e are respectively the fluorescence emission wavelength and the excitation wavelength. k is the number of sample, for which concentration can vary. γ is the relative emission spectrum, ϵ is the relative excitation spectrum, and c_k is the concentration of the component in the sample k. When excitation wavelengths in a certain range are used, an image called Fluorescent Excitation-Emission Matrices (FEEM) can be constructed. It becomes more difficult when several organic components are dissolved in the solution since the components have to be restored from their mixing. Yet, at low concentrations, the Beer-Lambert law is linear and reads:

$$I(\lambda_f, \lambda_e, k) = I_o \sum_{\ell} \gamma_\ell(\lambda_f)\, \epsilon_\ell(\lambda_e)\, c_{k,\ell} \quad (9)$$

Assuming a finite number of excitation and emission wavelengths, the data can be stored into a third order array. Comparing equations (9) and (1), we can identify the components, because of the uniqueness of the CP decomposition, guaranteed if F is not too large; see *e.g* [6] and refs therein. Then, up to scale and permutation, the separation provides the two relative spectra (from which we are able to recover the fluorescence images), $\gamma_\ell(\lambda_f(i))$ and $\epsilon_\ell(\lambda_e(j))$, that match respectively a_{if} and b_{jf}, and also the concentration of the components through all the samples $c_{k,\ell}$, that matches c_{kf}.

5.2. Obtained results

In the following, we will consider a tensor $\mathcal{T}$ with a number of components (or rank) $F = 4$ and whose emission-excitation matrices are of size 71×47 ($\mathbf{a}_i\mathbf{b}_i^T$, $\forall i = 1, \ldots, 4$). We randomly generate a positive matrix $\mathbf{C}$ (size 20×4). Thus, the considered tensor $\mathcal{T}$ is $71 \times 47 \times 20$. To compare the different algorithms, an error index is introduced: $E = \|\mathbf{T} - \widehat{\mathbf{T}}\|_F^2$ or, in dB, $E_{dB} = 10\log_{10}(E)$, where $\widehat{\mathbf{T}} = \sum_{f=1}^{F} \widehat{\mathbf{a}}_f \circledast \widehat{\mathbf{b}}_f \circledast \widehat{\mathbf{c}}_f$ and $\widehat{\mathbf{a}}$, $\widehat{\mathbf{b}}$ and $\widehat{\mathbf{c}}$ are the estimated factors. The reconstruction error of $\widehat{\mathbf{T}}$ is minimal and, consequently the best results are obtained, when $E = 0$ ($-\infty$ in logarithmic scale). In Fig. 1, we have compared the behaviour of different algorithms (our gradient and conjugate gradient methods with positivity constraint and the ALS suggested in [11, 8] with regularization of the cost function and projection). The gradient is the slowest and have the poorest results. The conjugate gradient offers a good compromise between speed and performances. Besides, contrary to Quasi-Newton algorithms (BFGS and DFP [9]), the conjugate gradient algorithm does not require the estimation of the $(I+J+K)F \times (I+J+K)F$ Hessian matrices (or their approximation), and as a consequence it can be applied to very large tensors. Concerning the ALS algorithm of [11], it is indeed rather fast to converge, but the performances are always satisfactory, even when the reconstruction error is small. In fact, as shown in the next section, a small reconstruction error does not necessarily imply a good estimation of the loading matrices.

5.3. Number of components overestimated

A mixture containing 4 components has been built, but we have assumed that $F = 5$ to perform the simulations (the rank is overestimated). The 5 reconstructed fluorescence images are displayed in Fig. 2. On the left, we have used the conjugate gradient algorithm whereas the ALS algorithm of [11] was used on the right. With the conjugate gradient algorithm, the four components have been well estimated (the fifth image is nearly null). With the ALS algorithm of [11], we cannot assert that there are 4 components instead of 5.

6. CONCLUSION

In this paper, we have proposed a new algorithm combining global search and conjugate gradient, to address the problem of the computation of the minimal polyadic decomposition of nonnegative three-way arrays. First, the gradient matrices have been derived, then, classical optimization algorithms such as gradient and conjugate gradient have been

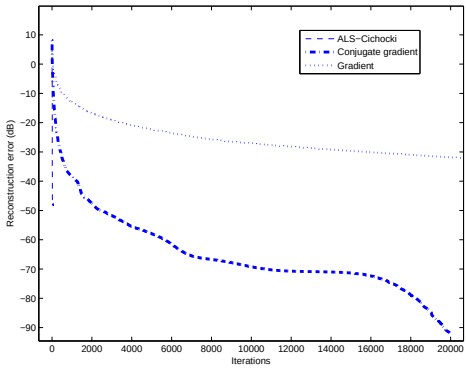

Fig. 1. Reconstruction error (dB) versus the number of iterations using a nonnegative $71 \times 47 \times 20$ tensor.

used to solve the factorization problem. Two solutions have been systematically studied: the ELS version and the backtracking version (alternating with ELS). Computer simulations have been provided to illustrate the good behaviour of the suggested approach and to compare it to other algorithms found in the literature.

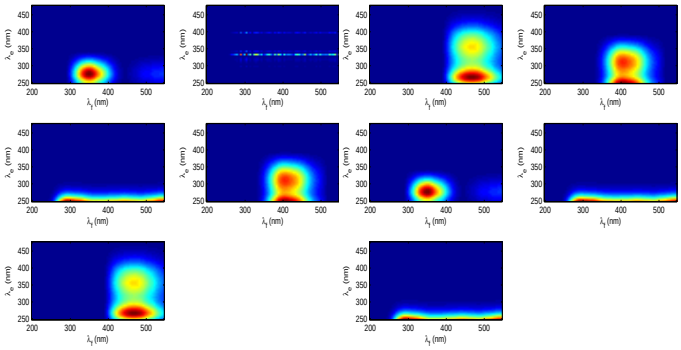

Fig. 2. Mixture of 4 factors, assuming $F = 5$; the 5 estimated emission-excitation images using the Conjugate Gradient algorithm with positivity constraint (left) and the ALS algorithm of [11, 8] (regularization + projection) (right).

Appendix A

Denote: $\mathbf{K}_1 = (\mathbf{C} \boxdot \mathbf{C}) \odot (\mathbf{B} \boxdot \mathbf{B})$, $\mathbf{K}_2 = (\mathbf{C} \boxdot \mathbf{C}) \odot (\mathbf{A} \boxdot \mathbf{A})$, $\mathbf{K}_3 = (\mathbf{B} \boxdot \mathbf{B}) \odot (\mathbf{A} \boxdot \mathbf{A})$. Then for $i \in \{1, 2, 3\}$:

$$
\begin{aligned}
&\mathsf{d}\mathcal{H}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = 2\mathsf{trace}\left\{\boldsymbol{\delta}_{(i)}^T \mathsf{d}\boldsymbol{\delta}_{(i)}\right\} \\
&= 4\mathsf{trace}\left\{-\boldsymbol{\delta}_{(1)}^T (\mathbf{A} \boxdot \mathsf{d}\mathbf{A})\mathbf{K}_1^T - \boldsymbol{\delta}_{(2)}^T (\mathbf{B} \boxdot \mathsf{d}\mathbf{B})\mathbf{K}_2^T \right. \\
&\left. -\boldsymbol{\delta}_{(3)}^T (\mathbf{C} \boxdot \mathsf{d}\mathbf{C})\mathbf{K}_3^T\right\} \\
&= \mathsf{trace}\left\{4\left(\mathbf{K}_1^T(-\boldsymbol{\delta}_{(1)})^T\right)(\mathbf{A} \boxdot \mathsf{d}\mathbf{A})\right\} \\
&+ \mathsf{trace}\left\{4\left(\mathbf{K}_2^T(-\boldsymbol{\delta}_{(2)})^T\right)(\mathbf{B} \boxdot \mathsf{d}\mathbf{B})\right\} \\
&+ \mathsf{trace}\left\{4\left(\mathbf{K}_3^T(-\boldsymbol{\delta}_{(3)})^T\right)(\mathbf{C} \boxdot \mathsf{d}\mathbf{C})\right\}
\end{aligned}
$$

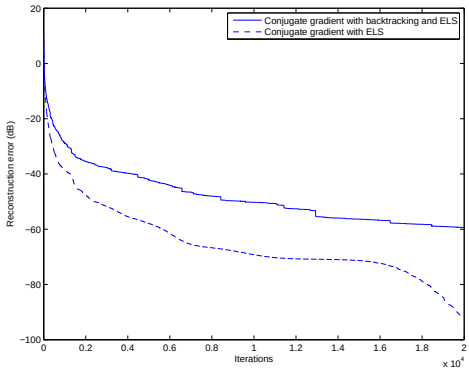

Fig. 3. Reconstruction error (dB) versus the number of iterations for ELS and backtracking (alternating with ELS) versions of the conjugate gradient algorithm.

$$
\begin{aligned}
&= \mathsf{trace}\left\{4\left[\left(\mathbf{K}_1^T(-\boldsymbol{\delta}_{(1)})^T\right) \boxdot \mathbf{A}^T\right]\mathsf{d}\mathbf{A}\right\} + \mathsf{trace}\left\{4\right. \\
&\left.\left[\left(\mathbf{K}_2^T(-\boldsymbol{\delta}_{(2)})^T\right) \boxdot \mathbf{B}^T\right]\mathsf{d}\mathbf{B} + 4\left[\left(\mathbf{K}_3^T(-\boldsymbol{\delta}_{(3)})^T\right) \boxdot \mathbf{C}^T\right]\mathsf{d}\mathbf{C}\right\} \\
&= \left\langle 4\left[\mathbf{A} \boxdot \left(-\boldsymbol{\delta}_{(1)}\mathbf{K}_1\right)\right], \mathsf{d}\mathbf{A}\right\rangle + \left\langle 4\left[\mathbf{B} \boxdot \left(-\boldsymbol{\delta}_{(2)}\mathbf{K}_2\right)\right], \mathsf{d}\mathbf{B}\right\rangle \\
&+ \left\langle 4\left[\mathbf{C} \boxdot \left(-\boldsymbol{\delta}_{(3)}\mathbf{K}_3\right)\right], \mathsf{d}\mathbf{C}\right\rangle
\end{aligned}
$$

7. REFERENCES

[1] A. Smilde, R. Bro, and P. Geladi, *Multi-Way Analysis with applications in the chemical sciences*, Wiley, 2004.

[2] R. Bro, "Parafac: tutorial and applications," *Chemometr. Intell. Lab.*, vol. 38, pp. 149–171, 1997.

[3] Q. Zhang, H. Wang, R. Plemmons, and P. Pauca, "Tensors methods for hyperspectral data processing: a space object identification study," *Journal of Optical Society of America A*, Dec. 2008.

[4] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *J. Math. and Phys.*, vol. 6, pp. 165–189, 1927.

[5] T. Lickteig, "Typical tensorial rank," *Linear Algebra Appl.*, vol. 69, pp. 95–120, 1985.

[6] P. Comon, X. Luciani, and A. L. F. De Almeida, "Tensor decompositions, alternating least squares and other tales," *Jour. Chemometrics*, vol. 23, pp. 393–405, Aug. 2009.

[7] L-H. Lim and P. Comon, "Nonnegative approximations of nonnegative tensors," *Jour. Chemometrics*, vol. 23, pp. 432–441, Aug. 2009.

[8] A. Cichocki, R. Zdunek, A. H. Phan, and S. I. Amari, *Non negative matrix and tensor factorizations*, Wiley, 2009.

[9] E. Polak, *Optimization algorithms and consistent approximations*, Springer, 1997.

[10] S. Boyd and L. Vandenberghe, *Convex optimization*, Cambridge University Press, Mar. 2004.

[11] M. Plumbley, A. Cichocki, and R. Bro, "Non-negative mixtures," in *Handbook of Blind Source Separation*, P. Comon & C.Jutten, Ed., chapter 1, pp. 515–547. Academic Press, London, UK, 2010.