A new penalized nonnegative third order tensor decomposition using a block coordinate proximal gradient approach: application to 3D fluorescence spectroscopy

Xuan Vu, Caroline Chaux, Nadège Thirion-Moreau, Sylvain Maire, Elfrida Mihaela Carstea

A new penalized nonnegative third order tensor decomposition using a block coordinate proximal gradient approach: application to 3D fluorescence spectroscopy

Xuan Vu*†, Caroline Chaux†, Nadège Thirion-Moreau*, Sylvain Maire*, Elfrida Mihaela Carstea‡

*Aix-Marseille Université, CNRS, ENSAM, LSIS, UMR 7296, F-13397 Marseille

Université de Toulon, CNRS, LSIS, UMR 7296, F-83957 La Garde, France, Tel: +33 4 94 14 24 56

{thirion, maire}@univ-tln.fr thi-thanh-xuan.vu@lsis.org

†Aix Marseille Univ, CNRS, Centrale Marseille, I2M, Marseille, France

39 rue F. Joliot-Curie, 13453 Marseille, France, Tel: +33 4 13 55 13 35

caroline.chaux@univ-amu.fr

‡National Institute for Optoelectronics, Atomistilor 409 Magurele, Ilfov, 077125, Romania

elfrida.carstea@inoe.ro

Abstract

In this article, we address the problem of tensor factorization subject to certain constraints. We focus on the Canonical Polyadic Decomposition (CPD) also known as Parafac. The interest of this multi-linear decomposition coupled with 3D fluorescence spectroscopy is now well established in the fields of environmental data analysis, biochemistry and chemistry. When real experimental data (possibly corrupted by noise) are processed, the actual rank of the "observed" tensor is generally unknown. Moreover, when the amount of data is very large, this inverse problem may become numerically ill-posed and consequently hard to solve. The use of proper constraints reflecting some a priori knowledge about the latent (or hidden) tracked variables and/or additional information through the addition of penalty functions can prove very helpful in estimating more relevant components rather than totally arbitrary ones. The counterpart is that the cost functions that have to be considered can be non convex and sometimes even non differentiable making their optimization more difficult, leading to a higher

Caroline Chaux is the corresponding author.

This work has been supported by Labex Archimède (ANR-11-LABX-0033).

computing time and a slower convergence speed. Block alternating proximal approaches offer a rigorous and flexible framework to properly address that problem since they are applicable to a large class of cost functions while remaining quite easy to implement. Here, we suggest a new block coordinate variable metric forward-backward method which can be seen as a special case of Majorize-Minimize (MM) approaches to derive a new penalized nonnegative third order CPD algorithm. Its interest, efficiency, robustness and flexibility are illustrated thanks to computer simulations carried out on both simulated and real experimental 3D fluorescence spectroscopy data.

Index Terms

Constrained optimization - Proximal approaches - Block alternating minimization - Nonnegative tensor factorization (NTF) - 3D fluorescence spectroscopy

I. INTRODUCTION

Acquired data sets in numerous modern applications are now often organized into multi-way arrays of numerical values because they are obtained thanks to ever more performant acquisition, data transmission and massive data storage systems. This is typically the case for the spectrofluorimetry [1] which is targeted in this article. This analysis technique is generally used to study the composition of solutions in order to detect the fluorescent chemical species (also called fluorophores) that are present. Once a coordinate basis is fixed, a tensor of order N can always be represented as an organized N-way array of numerical values. Since tensors and multi-linear algebra constitute a rigorous and natural mathematical framework for the formulation of many models and for the resolution of the problems that come along, they have been the subject of a growing interest in recent years. The order of a tensor corresponds to the dimensionality of the array needed to represent it (*i.e.* the number of indices required to reference the elements of the array). In this article, we will focus on third order tensors, yet, the approach suggested here can be generalized to higher tensor orders. Then, the tensors can be processed which means either directly compressed or analyzed using a great variety of tensor decompositions.

The most popular one certainly remains the low tensor rank decomposition also known by a number of other names among which are Canonical Polyadic Decomposition (CPD[1]), Candecomp [2], CanD or Parafac (for PARAllel FACtor analysis) [3].

This factorization method constitutes an informative and compact model which has proven to be relevant in many application fields, including those in which we are most interested here, namely chemistry and chemometry [4][5], process analysis/monitoring [6] and environmental data mining [7][8][9]. The Polyadic Decomposition consists of decomposing a tensor into a sum of R rank-1 tensors (with R suitably large). When this number R of rank-1 terms is minimal, the decomposition is referred to as the Canonical Polyadic Decomposition and R is

[1]It is the acronym that we will use in the rest of this article

called the *rank* of the tensor. However one of the main difficulties faced by practical fluorescence spectroscopy applications, is that the number of fluorophores is generally unknown and thus the actual rank of the "observed" tensor (*i.e.* the tensor of the acquired Fluorescence Emission Excitation Matrices, FEEM) is mostly unknown making the estimation of reliable fluorescent chemical compounds more difficult. Several works have been dedicated to the tensor rank estimation problem (see COre CONsistency DIAgnostic or CORCONDIA [10] and Threshold-CORCONDIA [11] to cite a few), however this problem is still hard and open. Consequently, we are interested here in new penalized methods ensuring the sparsity of the estimated loading matrices and thus insensitive to overestimation problems.

The CPD can also be regarded as a generalization of the matrix Singular Value Decomposition (SVD) to tensors or as a special case of another tensor decomposition known as the Tucker decomposition [12] by restricting the core tensor to be "diagonal". Another interesting advantage of this decomposition is its uniqueness under mild conditions [13][14][15], involving that on perfectly multilinear, noiseless models of known rank, the use of constraints is not necessary.

When real experimental data are processed, there might be noises, the constituent vectors of the *loading matrices* (also known as the *loading factors* or *vectors*) can be close and the observed system can be dynamic (appearance or disappearance of compounds). The actual rank of the observed tensor is unknown and may be difficult to estimate. Moreover, when the amount of data becomes very large, this inverse problem may become numerically ill-posed [16] and very difficult to solve [17][18]. The use of proper constraints reflecting some *a priori* knowledge about the latent or hidden variables that are tracked can be very helpful for the "unsupervised" estimation of reliable components. The whole processing chain can be rendered more robust and easier to use which is interesting when real-time monitoring or automated control systems are considered. Problems such as the automatic detection of water pollution could be tackled by this kind of approach.

Therefore, in the context of the 3D fluorescence spectroscopy analysis [19][20][7] (assuming the absence of errors coming either from the pre-processing used to remove Raman and Rayleigh scattering or possible bad settings of the devices) nonnegativity constraints should be considered given the physical nature of the hidden variables. In fact, in this particular application (see [21] for a reminder of the links that exist between 3D fluorescence spectroscopy and CPD), the loading vectors stand for physical quantities intrinsically nonnegative since they are related to emission and excitation spectra and concentrations through the samples acquired at different times in monitoring applications (or locations or pH for other kind of applications). The rank of the CP model that will approximate the "observed" tensor is closely linked to the number of fluorescent chemical compounds that are present in the studied samples.

The nonnegativity of the considered datasets as well as the one of the quantities that have to be estimated is also crucial for numerous other leading applications of CPD, especially those encountered in the area of

image processing (for example hyperspectral imaging [22][23], computer vision [24][25], biomedical image processing [26][27] or functional magnetic resonance imaging (fMRI) for brain mapping [28][29], etc.). It is the reason why it has given rise to numerous nonnegative CPD algorithms (see [30] or [21] for example for an overview of those NTF algorithms).

If nonnegativity constraints are sufficient on known simulated models, sometimes they may not be for the correct estimation of latent variables in case of noise, model errors or real experimental data (*i.e.* unknown mixtures of an unknown number of compounds possibly corrupted by noise). When dealing with complicated scenarios, most algorithms are unable to identify the relevant components, leaving the end-user to decide which components have a chemical meaning. To automate decisions, algorithms can be helped to recover more reliable components. This is achieved by the use of additional information or constraints. These are the reasons why we are concerned with possibly sparse loading matrices and/or much more continuous than discontinuous loading factors. We will address these issues with the help of the addition of penalty/regularization functions. The counterpart is that the cost functions that have to be considered are not necessarily convex and sometimes are even non differentiable making their optimization more difficult. The introduction of constraints may also involve a higher computing time and a reduced convergence speed.

To properly address the problem of tensor factorizations subject to certain constraints, we suggest, here, to consider a Block Coordinate Variable Metric Forward-Backward (BC-VMFB) approach [31] where the forward stage consists of a gradient step and the backward stage consists of a proximal step. This algorithm could thus also be called a Block Coordinate Proximal Gradient algorithm. The term “variable metric” means that a preconditioning is used. The aim of preconditioning is to increase the convergence rate. Such an approach has already been successfully used in the case of sparse non Negative Matrix Factorization (NMF) problems i) without preconditioning and the resulting algorithm is denoted by PALM (for Proximal Alternating Linearized Minimization) [32] and ii) with preconditioning [33] (but we notice that in this case the authors were using an alphabet of known spectra for the endmembers). An unpreconditioned version has been proposed for third order tensor decompositions in [34], where both CPD and Tucker decomposition have been studied. NTF problem has also been solved recently in [35] using an accelerated projection gradient based algorithm. Finally, we recall that BC-VMFB approaches can be seen as a special case of block alternating Majorize-Minimize approaches. MM approaches [36] offer a rigorous and flexible framework to properly address the problem of penalized non-negative third order CPD since they are applicable to a large class of cost functions while remaining quite easy to implement.

This article is organized as follows: after a brief introduction, the notations are given and some recalls about constrained optimization are performed in Section II. We focus more specifically on proximity operators and block-coordinate forward-backward methods. This general framework is then used to tackle the problem of the

penalized nonnegative canonical polyadic decomposition of third order tensors in Section III. To that aim, the model and the objective functions that are considered in this case are first introduced. Then, it is shown how the CPD problem can be integrated into the variational approaches framework. The gradient matrices are recalled and the preconditioning matrices that will be used are provided. Finally, algorithms devoted to the problem of CPD under different constraints can be derived. In Section IV, computer simulations are performed in the specific context of 3D fluorescence spectroscopy. Both simulated and real experimental data are considered. The obtained results illustrate the interest and the good behavior of the proposed algorithm for such applications. The suggested algorithm is also compared to other existing approaches. Finally, a conclusion is drawn and perspectives are delineated.

Notations: in this article, scalars, vectors, matrices and tensors are denoted by lower case (*e.g.* x), bold lower case (*e.g.* $\mathbf{a}$), bold upper case (*e.g.* $\mathbf{A}$) and calligraphic (*e.g.* $\mathcal{T}$) letters, respectively. $\mathbb{R}$ stands for the set of real numbers, $\mathbb{R}_+$ stands for the set of real nonnegative numbers and $\mathbb{N}$ for the set of natural numbers. The domain of $\mathcal{R}$ is denoted by $\mathsf{dom}\mathcal{R}$: it is the set of values for which the function $\mathcal{R}$ is defined. $\|\cdot\|$, $\|\cdot\|_F$ and $\|\cdot\|_1$ are respectively the norm, the Frobenius norm and the ℓ_1-norm. Π_C is the projector onto a closed subset C, whereas $\langle\cdot,\cdot\rangle$ is the inner product. The inverse of a matrix is denoted by $(\cdot)^{-1}$, its transpose by $(.)^\top$. The outer product of vectors is denoted by $\circ$, $\odot$ stands for the Khatri-Rao product, $\oslash$ for the Hadamard division between two matrices and $\boxdot$ for the Hadamard product between two matrices. The matrix trace is denoted by $\mathsf{trace}(.)$ and $\mathsf{Diag}(.,\ldots,.)$ is a diagonal matrix whose diagonal elements are those given in arguments.

II. Theoretical background

A. Context

A way to address inverse problems is to formulate them under a variational approach. This means that we seek a solution to an optimization problem and more precisely to a minimization problem in which the involved functional to be minimized often consists of two terms: one linked to the noise properties, named "data fidelity term" and another one linked to *a priori* information on the target solution, named "regularization" [37]. The minimization problem can thus be expressed as

$$\underset{\mathbf{x}\in\mathbb{R}^L}{\text{minimize}} \quad \underbrace{\mathcal{F}(\mathbf{x})}_{\text{Fidelity}} \quad + \quad \underbrace{\mathcal{R}(\mathbf{x})}_{\text{Regularization}} \quad . \tag{1}$$

Concerning the data fidelity term (here denoted by $\mathcal{F}$), one may choose *e.g.*, in relation to the noise statistics that may corrupt the data, a quadratic term (linked to a Gaussian noise) or a Kullback-Leibler divergence (linked to a Poisson noise). The regularization $\mathcal{R}$ can aim at enforcing the solution sparsity possibly in a transformed domain (*e.g.* wavelet transform, Fourier Transform, time-frequency domain, etc.) [38] or at favoring a piecewise smooth behavior (Total-Variation). It can also model some hard constraints such as, for example, a range constraint or

more precisely, a nonnegativity constraint. In this article, $\mathcal{F}$ and $\mathcal{R}$ are assumed to satisfy assumptions $\mathbf{H}_1$ and $\mathbf{H}_2$ stated as follows.

$\mathbf{H}_1$: $\mathcal{F}$ and $\mathcal{R}$ are proper[2] lower semi-continuous functions[3].

$\mathbf{H}_2$: $\mathcal{F}$ is differentiable with a β-Lipschitz gradient [4] on the domain $\mathsf{dom}\mathcal{R}$ and its gradient is denoted by $\nabla\mathcal{F}(\mathbf{x})$. In other words it means that $\mathcal{F}$ is sufficiently regular.

These technical assumptions play a prominent role in the algorithm's convergence proof.

In Problem (1), the regularization term $\mathcal{R}(\mathbf{x})$ may be split in a sum of J terms, leading to

$$\mathcal{R}(\mathbf{x}) = \sum_{j=1}^{J} \mathcal{R}_j(\mathbf{x}) \tag{2}$$

where for all $j = 1, \ldots, J$, $\mathcal{R}_j : \mathbb{R}^L \to]-\infty, +\infty]$ satisfies $\mathbf{H}_1$ and $\mathbf{H}_3$.

$\mathbf{H}_3$: $\mathcal{R}_j$ is assumed to be bounded from below by an affine function, and its restriction to its domain is continuous.

This allows to consider simultaneously various *a priori* on the solution (*e.g.* sparsity, nonnegativity, regularity, and so on).

Instead of performing the optimization on the whole set of unknowns at once, it is sometimes fruitful (either because of the intrinsic structure of the data, or performance issues or high size of the dataset) to compute updates and consider regularization on smaller sets of unknowns (called *blocks* in which the unknowns are stored[5]). The optimization is then performed on one block at a time. The different blocks are inspected according to a certain scanning rule, but after a given number of iterations all the blocs must have been swept. Such an idea is also used in randomization or stochastic optimization [39][40] to be able to tackle the computational problems that occur when large scale datasets are processed. It can also be interesting if local instead of global constraints have to be enforced. If data blocks are denoted by $\mathbf{x}^{(j)}$, we thus have $\mathbf{x} = (\mathbf{x}^{(j)})_{1 \leq j \leq J}$ and each block belongs to $\mathbb{R}^{L_j}$ where $\sum_{j=1}^{J} L_j = L$. The regularization term thus becomes

$$\mathcal{R}(\mathbf{x}) = \sum_{j=1}^{J} \mathcal{R}_j(\mathbf{x}^{(j)}) \tag{3}$$

where for all $j = 1, \ldots, J$, $\mathcal{R}_j : \mathbb{R}^{L_j} \to]-\infty, +\infty]$ satisfies $\mathbf{H}_1$ and $\mathbf{H}_3$.

[2]which means that $\mathsf{dom}\mathcal{F}$ and $\mathsf{dom}\mathcal{R}$ are non empty.

[3]$\mathcal{F}$ (resp. $\mathcal{R}$) is lower-semicontinuous at $\mathbf{x} \in \mathbb{R}^L$ if, for every sequence $(\mathbf{x}_i)_{i\in\mathbb{N}} \in \mathbb{R}^L$, $\lim_{i\to+\infty} \|\mathbf{x}_i - \mathbf{x}\| = 0 \Rightarrow \mathcal{F}(\mathbf{x}) \leq \liminf \mathcal{F}(\mathbf{x}_i)$ (resp. $\mathcal{R}(\mathbf{x}) \leq \liminf \mathcal{R}(\mathbf{x}_i)$).

[4]which means that $\forall(\mathbf{x}, \mathbf{y}) \in (\mathsf{dom}\mathcal{F})^2$, $\|\nabla\mathcal{F}(\mathbf{x}) - \nabla\mathcal{F}(\mathbf{y})\| \leq \beta \|\mathcal{F}(\mathbf{x}) - \mathcal{F}(\mathbf{y})\|$ where $\beta \in]0, +\infty[$ (β is called the Lipschitz constant)

[5]In the case of the CPD problem, we will see in the next section, that one "natural" partitioning of the unknowns is the one provided by the factor matrices themselves (such a partitioning is used in all popular alternate algorithms (ALS, HALS, fast HALS, etc.)), but other subsets of unknowns could be used

At this stage, it should be mentioned that:

1) the resulting functional to be minimized (Eqs. (1)-(3)) is often non differentiable and is sometimes not convex which makes the minimization problem difficult to solve,
2) regarding the blocks, a strategy to scan them has to be adopted. Indeed, they can be checked either i) randomly, ii) cyclically or iii) quasi-cyclically, making sure to check each block at least once regularly. These different scanning strategies play a role in the algorithm convergence proof but also in the algorithm convergence rate.

In this work, we will use a block alternating minimization method based on proximal gradient steps as described here after. Indeed, the Block Coordinate Variable Metric Forward-Backward algorithm (BC-VMFB) [31], [41] allows to tackle the general minimization problem described by Eqs. (1)-(3) and additionally the convergence to a critical point is guaranteed. Furthermore, the introduction of preconditioners can help to accelerate the convergence. Before describing the algorithm, we now briefly review the notion of proximity operators.

B. An introduction to proximity operators

We mainly work, here, with functions φ satisfying $\mathbf{H}_1$ (also denoted by $\varphi \in \Gamma_0(\mathbb{R})$). Their proximity operator [42] is defined as

$$\mathrm{prox}_{\varphi}\colon \mathbb{R} \to \mathbb{R}\colon v \mapsto \arg\min_{u\in\mathbb{R}} \frac{1}{2}\left\|u-v\right\|^2 + \varphi(u), \tag{4}$$

where $\arg\min$ means the position at which the function is minimum. The proximity operator can be viewed as a generalization of the projection operator. Indeed, when φ is the indicator function ι_C of a nonempty closed convex subset C of $\mathbb{R}$, *i.e.* it takes on the value 0 in C and $+\infty$ in $\mathbb{R}\setminus C$, prox_{ι_C} reduces to the projector Π_C onto C. Explicit forms of this operator are known for numerous functions $\varphi \in \Gamma_0(\mathbb{R})$ [43][37][44]. Of particular interest in this work is the proximity operator of the ℓ_1-norm function. More precisely, let $\alpha > 0$, and set $\varphi\colon \mathbb{R} \to \mathbb{R}\colon \xi \mapsto \alpha|\xi|$ (where $|\cdot|$ means the absolute value). Then, for every $\xi \in \mathbb{R}$,

$$\mathrm{prox}_{\varphi}\,\xi = \mathrm{sign}(\xi)\max\{|\xi|-\alpha, 0\} \quad \text{with} \quad \mathrm{sign}(\xi) = \begin{cases} +1 & \text{if } \xi > 0 \\ -1 & \text{if } \xi < 0 \\ 0 & \text{if } \xi = 0 \end{cases} \tag{5}$$

where $\max\{\cdot,\cdot\}$ means the maximum of the two values given in argument. The proximity operator of the ℓ_1-norm function reduces to a soft thresholding operation. Another example of interest is the proximity operator of the (squared) ℓ_2-norm function. More precisely, let $\alpha > 0$, and set $\varphi\colon \mathbb{R} \to \mathbb{R}\colon \xi \mapsto \alpha|\xi|^2$. Then, for every $\xi \in \mathbb{R}$,

$$\mathrm{prox}_{\varphi}\,\xi = \frac{\xi}{2\alpha+1}. \tag{6}$$

More recently, some authors [45] introduced the notion of proximity operators associated with a Symmetric Positive Definite (SPD) matrix $\mathbf{P}$ that is considering the metric induced by $\mathbf{P}$ in the definition of the proximity operator. First, we recall that $\forall \mathbf{x} \in \mathbb{R}^L$, $\|\mathbf{x}\|_{\mathbf{P}}^2 = \langle \mathbf{x}, \mathbf{P}\mathbf{x} \rangle$, where $\langle \cdot, \cdot \rangle$ is the inner product. The proximity operator of a function $\varphi \in \Gamma_0(\mathbb{R}^L)$ associated with a SPD matrix $\mathbf{P}$ is defined as [45]

$$\text{prox}_{\mathbf{P},\varphi} \colon \mathbb{R}^L \to \mathbb{R}^L \colon \mathbf{v} \mapsto \arg\min_{\mathbf{u} \in \mathbb{R}^L} \frac{1}{2} \|\mathbf{u} - \mathbf{v}\|_{\mathbf{P}}^2 + \varphi(\mathbf{u}). \tag{7}$$

Note that if $\mathbf{P}$ reduces to the identity matrix, then the definition (7) of the proximity operator associated with a SPD matrix reduces to the classical definition of the proximity operator given in (4).

C. Block-Coordinate Variable Metric Forward-Backward (BC-VMFB) algorithm

A proximal algorithm based on proximity operators associated with a SPD matrix $\mathbf{P}$ [31] is presented here. The considered minimization problem reads

$$\underset{\mathbf{x} \in \mathbb{R}^L}{\text{minimize}} \;\; \mathcal{F}(\mathbf{x}) + \sum_{j=1}^{J} \mathcal{R}_j(\mathbf{x}^{(j)}). \tag{8}$$

where, as aforementioned, $\mathcal{F}$ is smooth, satisfies $\mathbf{H}_1$, as well as each function $\mathcal{R}_j : \mathbb{R}^{L_j} \to]-\infty, +\infty]$ for all $j = 1, \ldots, J$ which additionally satisfy $\mathbf{H}_3$.

The optimization problem defined in (8) can be solved using a Block-Coordinate Variable Metric Forward-Backward (BC-VMFB) algorithm whose principle is summarized below in Algorithm 1.

Algorithm 1 Block-Coordinate Variable Metric Forward-Backward (BC-VMFB) algorithm.

Let $\mathbf{x}_0 \in \text{dom}\mathcal{R}$, $k \in \mathbb{N}$ and $\gamma_k \in]0, +\infty[$ *// Initialization step*

for $k = 0, 1, \ldots$ ***do*** *// k-th iteration of the algorithm*

  Let $j_k \in \{1, \ldots, J\}$ *// Processing of block number j_k (chosen, here, according to a quasi cyclic rule)*

  Let $\mathbf{P}_{j_k}(\mathbf{x}_k)$ be a SPD matrix *// Construction of the preconditioner $\mathbf{P}_{j_k}(\mathbf{x}_k)$*

  Let $\nabla_{j_k}\mathcal{F}(\mathbf{x}_k)$ be the Gradient *// Calculation of Gradient*

  $\tilde{\mathbf{x}}_k^{(j_k)} = \mathbf{x}_k^{(j_k)} - \gamma_k \mathbf{P}_{j_k}(\mathbf{x}_k)^{-1} \nabla_{j_k}\mathcal{F}(\mathbf{x}_k)$ *// Updating of block j_k according to a Gradient step*

  $\mathbf{x}_{k+1}^{(j_k)} \in \text{prox}_{\gamma_k^{-1}\mathbf{P}_{j_k}(\mathbf{x}_k), \mathcal{R}_{j_k}} \left(\tilde{\mathbf{x}}_k^{(j_k)} \right)$ *// Updating of block j_k according to a Proximal step*

  $\mathbf{x}_{k+1}^{\bar{j}_k} = \mathbf{x}_k^{\bar{j}_k}$ where $\bar{j} = \{1, \ldots, J\} \setminus \{j\}$ *// Other blocks are kept unchanged*

end for

We can see that it consists mainly of two steps:

❶ the computation of the gradient of $\mathcal{F}$ (more precisely a partial gradient w.r.t. the chosen block j (or j_k if it is considered at the $k-th$ iteration of the algorithm (see Algorithm 1)),

❷ the computation of the proximity operator of each $\mathcal{R}_j$ ($j \in \{1, \ldots, J\}$) associated to the metric $\mathbf{P}$.

It is important to notice that if the proximal stage ❷ (ensuring regularization and/or hard constraints) is removed, the algorithm goes back to an alternating preconditioned gradient method. The preconditioning matrix enables to accelerate the algorithm.

Finally, it has been proven that the convergence of this algorithm is guaranteed under assumptions [31] that are recalled in Appendix.

III. Application to the Penalized Nonnegative Tensor Factorization problem

Due to our application in 3D fluorescence spectroscopy, we now focus on the problem of the penalized nonnegative CP decomposition. We explain how this problem can be naturally expressed in the form of (8), justifying the use of the BC-VMFB approach to tackle the CPD problem.

A. Nonnegative Canonical Polyadic Decomposition of third order tensors

The (Canonical) polyadic decomposition consists of decomposing an original tensor into a (minimal) sum of rank-1[6] terms. In the case of third order tensor, it reads:

$$\overline{\mathcal{T}} = \sum_{r=1}^{\overline{R}} \bar{\mathbf{a}}_r^{(1)} \circ \bar{\mathbf{a}}_r^{(2)} \circ \bar{\mathbf{a}}_r^{(3)} = [\![\bar{\mathbf{A}}^{(1)}, \bar{\mathbf{A}}^{(2)}, \bar{\mathbf{A}}^{(3)}]\!], \tag{9}$$

where $\overline{\mathcal{T}} \in \mathbb{R}_+^{I_1 \times I_2 \times I_3}$ is a nonnegative third order tensor, $\circ$ is the outer product of vectors $\bar{\mathbf{a}}_r^{(n)}$ $(n = 1, \ldots, 3)$ which are called the loading factors, and $\bar{\mathbf{a}}_r^{(n)} = (\overline{a}_{i_n,r}^{(n)})_{i_n} \in \mathbb{R}^{I_n}$ with $i_n = 1, \ldots, I_n$ for all $n \in \{1, 2, 3\}$ and for all $r \in \{1, \ldots, \overline{R}\}$. The matrix $\bar{\mathbf{A}}^{(n)} = [\bar{\mathbf{a}}_1^{(n)}, \bar{\mathbf{a}}_2^{(n)}, \ldots, \bar{\mathbf{a}}_{\overline{R}}^{(n)}] \in \mathbb{R}_+^{I_n \times \overline{R}}$ is called n-th loading matrix. Denote by $\overline{\mathbf{T}}_{I_n, I_{-n}}^{(n)} \in \mathbb{R}_+^{I_n \times I_{-n}}$ the matrix obtained by unfolding the tensor $\overline{\mathcal{T}}$ in the n-th mode where the size I_{-n} is equal to $I_1 I_2 I_3 / I_n$. The previous model can be written in the matrix form as follows

$$\overline{\mathbf{T}}_{I_n, I_{-n}}^{(n)} = \bar{\mathbf{A}}^{(n)} (\overline{\mathbf{Z}}^{(-n)})^\top, \quad n \in \{1, 2, 3\} \tag{10}$$

where

$$\begin{aligned}
\overline{\mathbf{Z}}^{(-1)} &= \bar{\mathbf{A}}^{(3)} \odot \bar{\mathbf{A}}^{(2)} \in \mathbb{R}_+^{I_{-1} \times \overline{R}}, \\
\overline{\mathbf{Z}}^{(-2)} &= \bar{\mathbf{A}}^{(3)} \odot \bar{\mathbf{A}}^{(1)} \in \mathbb{R}_+^{I_{-2} \times \overline{R}}, \\
\overline{\mathbf{Z}}^{(-3)} &= \bar{\mathbf{A}}^{(2)} \odot \bar{\mathbf{A}}^{(1)} \in \mathbb{R}_+^{I_{-3} \times \overline{R}},
\end{aligned}$$

with $\odot$ standing for the Khatri-Rao product [46] and $(.)^\top$ denoting the transpose of a matrix. Solely from an observation $\mathcal{T}$ of $\overline{\mathcal{T}}$ (e.g. $\mathcal{T}$ can be a noisy version of $\overline{\mathcal{T}}$), our aim is to estimate the hidden variables *i.e.* the loading matrices $\bar{\mathbf{A}}^{(n)}$ for all $n = 1, \ldots 3$. To reach that goal, we are going to express this problem under a variational approach. It will lead us to a minimization problem of the form given by (8), whose solution will constitute an estimate $\widehat{\mathcal{T}}$ of $\overline{\mathcal{T}}$.

[6]A rank-1 tensor of order N (resp. 3) is defined as the outer product of N (resp. 3) vectors.

B. Proposed approach

The tensor structure naturally leads, in the case of third order (resp. N-th order) tensor, to consider 3 (resp. N) blocks corresponding to the loading matrices $\mathbf{A}^{(1)}$, $\mathbf{A}^{(2)}$ and $\mathbf{A}^{(3)}$ (resp. $\mathbf{A}^{(1)}$, ..., $\mathbf{A}^{(N)}$). Thus, we propose to solve the following optimization problem with regularization for $\mathbf{A}^{(1)}$, $\mathbf{A}^{(2)}$ and $\mathbf{A}^{(3)}$

$$\underset{\mathbf{A}^{(n)}\in\mathbb{R}^{I_n\times R},n\in\{1,2,3\}}{\text{minimize}} \quad \mathcal{F}(\mathbf{A}^{(1)},\mathbf{A}^{(2)},\mathbf{A}^{(3)})+\mathcal{R}_1(\mathbf{A}^{(1)})+R_2(\mathbf{A}^{(2)})+\mathcal{R}_3(\mathbf{A}^{(3)}). \tag{11}$$

We opt for a quadratic data fidelity term

$$\mathcal{F}(\mathbf{A}^{(1)},\mathbf{A}^{(2)},\mathbf{A}^{(3)})=\frac{1}{2}\|\mathcal{T}-[\![\mathbf{A}^{(1)},\mathbf{A}^{(2)},\mathbf{A}^{(3)}]\!]\|_F^2=\frac{1}{2}\|\mathbf{T}^{(n)}_{I_n,I_{-n}}-\mathbf{A}^{(n)}\mathbf{Z}^{(-n)^\top}\|_F^2 \tag{12}$$

and the penalization terms $\left(\mathcal{R}_n(\mathbf{A}^{(n)})\right)_{n\in\{1,2,3\}}$ are defined by

$$\mathcal{R}_n(\mathbf{A}^{(n)})=\sum_{i_n=1}^{I_n}\sum_{r=1}^{R}\rho_n(a^{(n)}_{i_nr}) \qquad \forall n\in\{1,2,3\} \tag{13}$$

where the loading matrices are defined element wise as $\mathbf{A}^{(n)}=(a^{(n)}_{i_nr})_{(i_n,r)\in\{1,\ldots,I_n\}\times\{1,\ldots,R\}}$ and where

$$\rho_n(\omega)=\begin{cases}\alpha^{(n)}|\omega|^{\pi^{(n)}} & \text{if } \eta^{(n)}_{\min}\leq\omega\leq\eta^{(n)}_{\max}\\ +\infty & \text{otherwise}\end{cases} \tag{14}$$

and $\alpha^{(n)}\in]0,+\infty[$, $\pi^{(n)}\in\mathbb{N}^*$, $\eta^{(n)}_{\min}\in[-\infty,+\infty[$ and $\eta^{(n)}_{\max}\in[\eta^{(n)}_{\min},+\infty]$. This shows that, in the example developed in this article, the regularization parameters are chosen block dependent but are constant within a block (in our case a block will correspond to a loading matrix but other splittings could have been chosen).

Then, the minimization can be performed using Algorithm 1 with $J=3$ (*i.e.* considering 3 blocks namely $\mathbf{A}^{(1)},\mathbf{A}^{(2)}$ and $\mathbf{A}^{(3)}$). At the convergence (i.e. after $k_{\max}$ iterations), the algorithm will provide estimations $\widehat{\mathbf{A}}^{(n)}$ of loading matrices $\bar{\mathbf{A}}^{(n)}$ for all $n\in\{1,2,3\}$.

As mentioned in the previous Section, algorithm 1 requires to compute the partial gradient matrices of $\mathcal{F}$ with respect to $\mathbf{A}^{(n)}$ for all $n=1,\ldots,3$ but also the proximity operators of each regularization term $\left(\mathcal{R}_n(\mathbf{A}^{(n)})\right)_{n\in\{1,2,3\}}$ associated to the metric $\mathbf{P}^{(n)}$.

We recall that the gradient matrices of $\mathcal{F}$ with respect to $\mathbf{A}^{(n)}$ for all $n=1,\ldots,3$, are defined as

$$\nabla_n\mathcal{F}(\mathbf{A}^{(1)},\mathbf{A}^{(2)},\mathbf{A}^{(3)})=-(\mathbf{T}^{(n)}_{I_n,I_{-n}}-\mathbf{A}^{(n)}\mathbf{Z}^{(-n)^\top})\mathbf{Z}^{(-n)}. \tag{15}$$

In Algorithm 1, the gradient matrix $\nabla_n\mathcal{F}(\mathbf{A}^{(1)}[k],\mathbf{A}^{(2)}[k],\mathbf{A}^{(3)}[k])$ is replaced by a more compact notation $\nabla_n[k]$ where k stands for the iteration step.

Another key point is to derive good preconditioners. Inspired by [47] and [33], we generalized the preconditioners proposed in [33] (dedicated to NMF) to the case of NTF problems. To this end, we adopted a similar

approach using the n-th mode unfolding of the tensor given in (10) (*i.e.* a matrix form of the NTF problem). The majorant function $\mathcal{Q}$ involved in the algorithm's convergence proof (see (23) of Appendix) writes

$$\mathcal{Q}_n(\mathbf{A}^{(n)}|\mathbf{A}^{(1)}[k],\mathbf{A}^{(2)}[k],\mathbf{A}^{(3)}[k]) = \mathcal{F}(\mathbf{A}^{(1)}[k],\mathbf{A}^{(2)}[k],\mathbf{A}^{(3)}[k]) + \mathsf{trace}((\mathbf{A}^{(n)}-\mathbf{A}^{(n)}[k])^\top \nabla_n[k]) + \frac{1}{2}\mathsf{trace}((\mathbf{A}^{(n)}-\mathbf{A}^{(n)}[k])^\top(\mathbf{P}^{(n)}[k]\boxdot(\mathbf{A}^{(n)}-\mathbf{A}^{(n)}[k]))),\ \forall n\in\{1,2,3\}, \quad (16)$$

where $\boxdot$ denotes the Hadamard product between two matrices and $\mathbf{P}^{(n)}[k]$ is a compact notation standing for $\mathbf{P}^{(n)}(\mathbf{A}^{(1)}[k],\ \mathbf{A}^{(2)}[k],\mathbf{A}^{(3)}[k])$.

The matrix $\mathbf{P}$ for the n-th block arising in (16) can be defined as follows

$$\mathbf{P}^{(n)}(\mathbf{A}^{(1)},\mathbf{A}^{(2)},\mathbf{A}^{(3)}) = \mathbf{A}^{(n)}(\mathbf{Z}^{(-n)^\top}\mathbf{Z}^{(-n)}) \oslash \mathbf{A}^{(n)}, \qquad \forall n\in\{1,2,3\}, \quad (17)$$

where $\oslash$ denotes the Hadamard division between two matrices.

Finally, the proximity operator associated with each $(\mathcal{R}_n)_{n\in\{1,2,3\}}$ and corresponding to the computation of $\mathrm{prox}_{\gamma[k]^{-1}\mathbf{P}^{(n)}[k],\mathcal{R}_n}(\widetilde{\mathbf{A}}^{(n)}[k])$ still has to be delineated. To rigorously define it, we need to vectorize the data[7], that is

- Let define each element of the preconditioner matrix $\mathbf{P}^{(n)}[k]$ in (17) as $(p_i^{(n)}[k])_{i\in\{1,\ldots,RI_n\}}$ and let define the diagonal matrix $\widetilde{\mathbf{P}}^{(n)}[k]$ as $\widetilde{\mathbf{P}}^{(n)}[k] = \mathsf{Diag}(p_1^{(n)}[k],\ldots,p_{RI_n}^{(n)}[k])$,
- Let $\widetilde{\mathbf{A}}^{(n)}[k]$ be stored in a vector $\widetilde{\mathbf{a}}^{(n)}[k]\in\mathbb{R}^{RI_n}$.

Then, the computation of $\mathrm{prox}_{\gamma[k]^{-1}\mathbf{P}^{(n)}[k],\mathcal{R}_n}(\widetilde{\mathbf{A}}^{(n)}[k])$ consists in computing $\mathrm{prox}_{\gamma[k]^{-1}\widetilde{\mathbf{P}}^{(n)}[k],\mathcal{R}_n}(\widetilde{\mathbf{a}}^{(n)}[k])$ whose definition is given in (7).

The regularization being separable and given the shape of the preconditioning matrices $\widetilde{\mathbf{P}}^{(n)}[k]$, proximity operator can be expressed as

$$(\forall y=(y^{(i)})_{i\in\{1,\ldots,RI_n\}}\in\mathbb{R}^{RI_n}) \qquad \mathrm{prox}_{\gamma[k]^{-1}\mathbf{P}^{(n)}[k],\mathcal{R}_n}(y) = \left(\mathrm{prox}_{\gamma[k]^{-1}p_i^{(n)}[k],\rho_n}(y^{(i)})\right)_{i\in\{1,\ldots,RI_n\}}. \quad (18)$$

For all $i\in\{1,\ldots,RI_n\}$, we have [48] $(\forall \upsilon\in\mathbb{R})$

$$\mathrm{prox}_{\gamma[k]^{-1}p_i^{(n)},\rho_n}(\upsilon) = \min\left\{\eta_{\max}^{(n)},\max\left\{\eta_{\min}^{(n)},\mathrm{prox}_{\gamma[k]\alpha^{(n)}(p_i^{(n)}[k])^{-1}|\,.\,|^{\pi^{(n)}}}(\upsilon)\right\}\right\} \quad (19)$$

[Fig. 1 about here.]

The proximity operator of ρ_n is illustrated in Fig. 1 for fixed parameter values $\alpha^{(n)}=2$ and $[\eta_{\min}^{(n)},\eta_{\max}^{(n)}]=[0,4]$. We see that the positivity constraint is applied (negative values are projected to 0) but also that the maximum value $\eta_{\max}^{(n)}$ is respected. Furthermore, when the ℓ_1-norm is considered, we recognize the thresholding rule (up

[7]Note that this data vectorization is not applied in practice: for the implementation elementwise operations are performed thus avoiding memory issues.

to 2 which is the value of the regularization parameter $\alpha^{(n)}$ chosen here) which enforces data sparsity. Note that when the ℓ_2-norm is considered, this is no longer the case.

For all the simulations that will be performed in the next section, we have considered $\pi^{(n)} \in \{1, 2\}$ (the proximity operator when $\pi^{(n)} = 1$ (resp. when $\pi^{(n)} = 2$) is recalled in Eq. (5) (resp. Eq. (6)). Regularization parameter $\alpha^{(n)}$ and range constraint parameters being simulation dependent, they will be defined in the numerical simulation section.

C. Derived algorithm

At this stage, all the elements involved in the BC-VMFB algorithm suggested, here, to solve the penalized nonnegative CPD problem have been defined. The general scheme of the resulting penalized nonnegative CPD algorithm is depicted in Fig. 2.

[Fig. 2 about here.]

Finally, we notice that we have chosen to update the blocks according to a quasi cyclic rule. It means that the block that is updated is chosen randomly according to a uniform law (see [17] for the importance of randomization in the choice of blocks). Moreover, we make sure that each block is chosen at least once every K iterations (with $K = 100$).

IV. Numerical simulations

In this section dedicated to computer simulations, we are going to show the interest of the BC-VMFB algorithm in the context of environmental data analysis. The method is applied to both synthetic (*i.e.* numerically simulated) 3D fluorescence spectroscopy data and raw data coming from an experiment of water quality monitoring. In order to better assess the performance of the suggested algorithm and to be able to compare it with other algorithms of the literature, two error indices are first introduced.

A. Error measures

When simulated data are used, the true tensor rank $\overline{R}$ is known, while the tensor rank that will be used for the model and thus for the decomposition is denoted by $\widehat{R}$. In the case of simulated data, we can consider the two error indices that have already been used in [49] instead of the reconstruction error[8] $\|\mathcal{T} - \widehat{\mathcal{T}}\|_F^2$ which is classically used with real experimental data. The first error index, denoted by $\mathbf{E}_1$, measures the error of estimation but discarding the over-factoring part. The second one, denoted by $\mathbf{E}_2$ measures the error induced by the over-factoring part only. The purpose is to evaluate more accurately the quality of the results. In this case,

[8]or the normalized reconstruction error $\frac{\|\mathcal{T}-\widehat{\mathcal{T}}\|_F^2}{\|\mathcal{T}\|_F^2}$, where $\widehat{\mathcal{T}}$ stands for the reconstructed (estimated) tensor.

the inherent indetermination of the CPD problem (*i.e.* permutation and scaling ambiguities) also have to be taken into account: if we apply to a given solution a permutation of their loading factors or an appropriate scaling of their loading matrices, the result is also a solution and defines the same decomposition. Therefore, permutation and scaling-independent measurements of the errors are necessary. For 3-way tensors, the loading matrices $\mathbf{A}^{(n)}$, for all $n \in \{1, \ldots, 3\}$, are first normalized in such a way that each column of $\mathbf{A}^{(n)}$ for $n \in \{1, ..., 2\}$ is normalized in l_1 and each column of $\mathbf{A}^{(3)}$ carries the weight.

The estimated normalized solutions of the CPD algorithm are denoted by $\widehat{\mathbf{A}}^{(n)}$. Then, the $\widehat{R}$ column vectors of $\widehat{\mathbf{A}}^{(n)}$ are permuted such that its Euclidean distance to $\mathbf{A}^{(n)}$ is minimized. The permuted normalized estimates are denoted by $\widehat{\mathbf{A}}_{\sigma}^{(n)}$ while $\widehat{\mathbf{A}}_{\sigma}^{(n)}(1:R)$ means that only its R first columns are considered and σ is the considered permutation. Thus, the estimation error $\mathbf{E}_1$ is defined by

$$\mathbf{E}_1(\sigma) = \frac{\sum_{n=1}^{3} \|\widehat{\mathbf{A}}_{\sigma}^{(n)}(1:R) - \mathbf{A}^{(n)}\|_1}{\sum_{n=1}^{3} \|\mathbf{A}^{(n)}\|_1} \tag{20}$$

$$\Rightarrow \begin{cases} \mathbf{E}_1 = \min_{\sigma} \mathbf{E}_1(\sigma) \quad \text{or} \quad \mathbf{E}_{1\mathsf{dB}} = 10\log_{10}(\mathbf{E}_1) \\ \sigma_{opt} = \arg\min_{\sigma} \mathbf{E}_1(\sigma) \end{cases} \tag{21}$$

The over-factoring error $\mathbf{E}_2$ concerns the remaining components $R+1, \ldots, \widehat{R}$ in (9). It is computed as follows

$$\mathbf{E}_2 = \| \sum_{r=R+1}^{\widehat{R}} \mathbf{a}_{\sigma_{opt},r}^{(1)} \circ \mathbf{a}_{\sigma_{opt},r}^{(2)} \circ \mathbf{a}_{\sigma_{opt},r}^{(3)} \|_1 \quad \text{or} \quad \mathbf{E}_{2\mathsf{dB}} = 10\log_{10}(\mathbf{E}_2). \tag{22}$$

B. Synthetic case

In this section, we illustrate the behavior of the BC-VMFB algorithm on two synthetic data sets: the first one, is noiseless whereas the second one, is corrupted by a Gaussian noise (with a $\mathsf{SNR} = 17.6$ dB). Here, the Signal to Noise Ratio (SNR) is defined as $\mathsf{SNR} = 20\log_{10} \frac{\|\overline{\mathcal{T}}\|_F}{\|\overline{\mathcal{T}} - \mathcal{T}\|_F}$ and the noisy data are truncated: all negative values are set to 0. In both cases, the tensor rank, $\overline{R}$, is equal to 5, yet, it will be overestimated and the tensor decomposition will be performed considering that it is equal to 6 (*i.e.* $\widehat{R} = 6$). The purpose is to evaluate the robustness versus noise of the penalized nonnegative CPD algorithm suggested here but also its ability to overcome model errors. The performance of the proposed algorithm will also be compared with other algorithms of the literature namely the Bro's N-way algorithm [50] and the fast HALS algorithm described in [51].

The studied data sets are built as follows. The tensors $\mathcal{T}$ and $\overline{\mathcal{T}}$ are $100 \times 100 \times 100$. The excitation and emission spectra are created using sums of (shifted) density function drawn from a generalized normal distribution and can be either monomodal or bimodal. They are truncated to ensure the non-negativity of the spectra. The concentrations are generated according to a uniform law between 0 and 10.

For our simulations, regarding the BC-VMFB algorithm, we have used the following parameters: $\eta_{\min}^{(n)} \equiv \eta_{\min} = 2e^{-16}$ and $\eta_{\max}^{(n)} \equiv \eta_{\max} = 1000$. The value of the exponent $\pi^{(n)}$ is chosen based on the maximum-likelihood method [52, p. 225] for each block and is equal to 1 for $n = 1, 2$ and to 2 for $n = 3$ in these tests. The step-size $\gamma[k] \equiv \gamma$ has been fixed to 0.99. Each time, we consider the two following cases: (i) with regularization parameter $\alpha^{(n)} \equiv \alpha = 0.05$ for all blocks (constant within blocks) and (ii) without regularization *i.e.* with $\alpha = 0$ for all blocks.

The estimated Fluorescence Emission Excitation Matrices (FEEM) are displayed in Fig. 3 (resp. Fig. 5) in the noiseless case (resp. in the noisy case for a $\mathsf{SNR} = 17.6$ dB). The estimated emission and excitation spectra as well as the estimated concentrations are provided in Fig. 4 (resp. Fig. 6) in the noise-free case (resp. noisy case for a $\mathsf{SNR} = 17.6$ dB).

[Fig. 3 about here.]

[Fig. 4 about here.]

[Fig. 5 about here.]

[Fig. 6 about here.]

The estimation error $\mathbf{E}_1$ is found to be equal to -15dB (resp. -12.4dB, -11.2dB, -12.5dB) and the over-factoring error $\mathbf{E}_2$ is found to be equal to -409dB (resp. 25.6dB, -409dB, 30.6dB) in the penalized noiseless case (resp. non penalized noiseless case, penalized noisy case and non penalized noisy case). In the noisy case, the final SNR is 32.3dB (resp. 31.3dB) in the penalized case (resp. non penalized case). We clearly observe that the penalized version of the BC-VMFB algorithm accurately estimates the loading matrices (even in the presence of noise) but is also able to overcome the problem of overfactoring by detecting the absence of a sixth component. In both noise-free and noisy cases, the regularized version of the BC-VMFB algorithm outperforms its non regularized version.

Finally, in the noisy, overestimated case, we compare the BC-VMFB algorithm ((i) with regularization $\alpha = 0.05$ and (ii) without regularization) with two other classical NTF algorithms of the literature ((iii) the Bro's N-way algorithm under non-negative constraints and (iv) the fast HALS algorithm under non-negative constraints) both in terms of performance and computational cost per iteration. To that aim, we draw 100 initial values randomly from a uniform distribution $\mathcal{U}(0, 1)$. The following stopping conditions were used: (a) the number of iterations is either equal to $k_{\max} = 10^5$ or (b) the relative diminishing rate of the quadratic criterion $\frac{\|\mathcal{F}[\cdot+l]-\mathcal{F}[\cdot]\|}{\mathcal{F}[\cdot]}$ is smaller than a given tolerance $tol = 10^{-8}$. Here l is equal to 1 for (iii) the N-way algorithm and (iv) the fast HALS algorithm, while for versions (i) and (ii) of the BC-VMFB algorithm, l corresponds to a block of 500 iterations. The obtained performance versus the different initializations are given in Fig. 7 (they are sorted

in ascending order). The error index $\mathbf{E}_1$ is displayed at the top of this figure while the (over-factoring) error index $\mathbf{E}_2$ is displayed at bottom of this figure. The computation time per iteration for each algorithm is given in Table I.

[Fig. 7 about here.]

[TABLE 1 about here.]

We observe that in most cases (98% of the initializations that were tested here), the penalized version of the BC-VMFB algorithm outperforms all the other algorithms (no fake compound is estimated as it is clearly emphasized by the over-factoring error index $\mathbf{E}_2$), yet the counterpart is that, in the noisy case, the performance measured thanks to the error index $\mathbf{E}_1$ is slightly below to that obtained with the other algorithms among which is the non penalized version of the BC-VMFB algorithm (the observed small difference is due to the fact that the noise is only distributed on existing compounds since non existing compounds are enforced to be equal to 0). In the remaining cases (2% of the tested initializations), the algorithm exhibits the same behavior as the other algorithms: the chosen initializations seem to be too bad to be able to recover the true solution. We also notice in Table I that the BC-VMFB algorithm remains competitive regarding the computation time (since the fast-HALS and this implementation of the Bro's N-way algorithm can be regarded as the fastest algorithms of the literature).

C. Experimental case: a water monitoring campaign

In this section, we now test our penalized BC-VMFB method on a real experimental data set [53]. In this experiment, the data were acquired automatically every 3 minutes, during a 10 days monitoring campaign performed on water extracted from an urban river. The size of the baseline data set is: $36 \times 111 \times 2594$. The excitation wavelengths range from 225nm to 400nm with a 5nm bandwidth, whereas the emission wavelengths range from 280nm to 500nm with a 2nm bandwidth. The FEEM have been pre-processed using the Zepp's method [54] implying that all remaining negative values were set to 0 (but it should have negligible impact on the overall analysis, since on random chosen FEEMs, less than 10 negative points out of 3996 were found (i.e. $\simeq 0.25\%$). One example of the FEEM before and after pre-processing is shown in Fig. 8.

[Fig. 8 about here.]

Additional pre-processing were applied: the data corresponding to the first 6 emission slits were removed. The first 1200 FEEMs (data acquired during the first six days) were discarded too and 2 other FEEMs (corresponding to the 1737th and 1738th acquisition time) were suppressed due to noticeable acquisition problems. Finally, the size of data that are processed in this example is $36 \times 105 \times 1392$. Different tensor ranks (*i.e.* different

estimates of the number of chemical compounds) were tested: $\widehat{R} = 4/5/6$ but the obtained results are presented in two cases only ($\widehat{R}$ = 4 and 6). Concerning the penalized BC-VMFB method, a regularization $\pi^{(n)} \equiv \pi = 1$ was applied on all loading matrices. When searching for $\widehat{R} = 4$ components (resp. $\widehat{R} = 6$ components), the following value $\alpha^{(n)} \equiv \alpha = 2 \times 10^4$ (resp. $\alpha = 8 \times 10^3$) were chosen for the regularization parameters for each loading factors. The reconstructed Fluorescence Excitation-Emission Matrices are displayed on the top of Fig. 9 for $\widehat{R} = 4$ compounds and at its bottom for $\widehat{R} = 6$ compounds. Results obtained with our method are given on the left part of this figure whereas there are given on its right part for the Bro's N-way algorithm. The (scaled) reconstructed concentrations are given on Fig. 10 (on its left with our method and on its right with the Bro's N-way algorithm). The (scaled) reconstructed concentrations for $\widehat{R} = 4$ compounds are given on the top of Fig. 10 and on its bottom for $\widehat{R} = 6$ compounds.

[Fig. 9 about here.]

[Fig. 10 about here.]

Finally on Fig. 11, we compare the (scaled) estimated excitation and emission spectra obtained when we assume that $\widehat{R} = 4$ compounds are present, using either our BC-VMFB algorithm with different values of the regularization parameters ((b) $\alpha = 3 \times 10^4$, (c) $\alpha = 2 \times 10^4$ and (d) $\alpha = 10^4$) or the Bro's N-way algorithm with nonnegativity constraints ((a)). Whereas, on Fig. 12, the same study is performed considering that $\widehat{R} = 6$, (for the BC-VMFB algorithm: (b) $\alpha = 8 \times 10^3$, (c) $\alpha = 7 \times 10^3$ and (d) $\alpha = 6 \times 10^3$).

[Fig. 11 about here.]

[Fig. 12 about here.]

Thanks to the obtained results, we are now able to conclude that only four fluorescent chemical compounds were present in the studied data set (the FEEM of the first and fourth estimated compounds in the case $\widehat{R} = 6$ are nearly null as well as their concentrations, see the left bottom part of the Fig. 9 and the left bottom part of the Fig. 10). During this experiment, a contamination with diesel oil [55] appeared 7 days after the beginning of the monitoring campaign: it can be clearly observed on the Fig. 9. In fact, before the 1480th sample, one single fluorescent chemical compound was mainly present (compound labeled (1) in the case $\widehat{R} = 4$, or compound labeled (6) in the case $\widehat{R} = 6$) whereas two others occur in trace amounts (compounds labeled (3) and (4) in the case $\widehat{R} = 4$, or compounds labeled (2) and (3) in the case $\widehat{R} = 6$). After this time, a fourth compound occurs (compound labeled (2) in the case $\widehat{R} = 4$, or compound labeled (5) in the case $\widehat{R} = 6$) and we also observe an important increase of the concentrations of the two aforementioned compounds. Moreover, the concentration curves of those three compounds seem to exhibit a similar behavior. Even if our estimated

spectra, concentrations and FEEM are sometimes close to those estimated with the Bro's N-way algorithm, they remain indeed different (it is particularly true for the concentrations and the FEEM). The main advantage of the BC-VMFB algorithm is that thanks to penalization the estimated spectra and concentrations are stable with respect to the tested ranks. It is not the case with the N-way algorithm. With our method, we were also able to decide that only four components were effectively present. We also observe the influence of the regularization parameters on the obtained results: the smallest regularization does not seem to lead to satisfying results in the case of 6 sought compounds, whereas the highest one is not adapted in the case of 4 sought compounds.

V. Conclusion

In this article, we addressed the problem of tensor factorizations subject to certain constraints (nonnegativity, sparsity, regularity, etc.). We tackled this problem within the broader framework of Block Coordinate Variable Metric Forward-Backward (BC-VMFB) approaches. The main interest of BC-VMFB approaches is to offer a clear theoretical and mathematical framework, since the conditions under which the sequence generated by this family of algorithms converges to a critical point of the objective function have been established in previous works of other authors. Through this general framework, we were able to derive a new penalized nonnegative third order CPD algorithm. In our case, the forward stage consists of a gradient step and the backward stage consists of a proximal step. Moreover, a preconditioning is also introduced in order to increase the convergence rate. Attention must still be drawn to the fact that some regularization parameters have to be set and that they may have an impact on the obtained results: that is why different strategies regarding the choice of the regularization terms that are added have been investigated. Computer simulations have been provided in order to enlighten the effectiveness and the robustness of the proposed approach in the applicative context of 3D fluorescence spectroscopy. Both simulated and real experimental data have been considered. Even if we only took advantage of a very small part of the enormous potential of the BC-VMFB approaches on those examples, we were able to illustrate some of their very interesting properties: reliability, robustness versus noise, good performance despite model errors and relative quickness. On real experimental data, identifying relevant components with traditional CPD algorithms is not always so straightforward, leaving the end-user to decide which components have a chemical meaning. The family of algorithms presented here can help to automate decisions. We focused on third order tensors but we have already extended these approaches to tensors of higher orders. The problem of possible missing data under the BC-VMFB framework will be addressed in future works.

Appendix

The BC-VMFB algorithm's convergence theorem reads

Theorem A.1. *[31, Theorem 3.1] Assume that Assumptions A.1 hold and that $\mathcal{F}+\mathcal{R}$ satisfies the Kurdyka-Łojasiewicz inequality*[9] *[57], [58]. Then the sequence $(\mathbf{x}_k)_{k\in\mathbb{N}}$ converges to a critical point $\hat{\mathbf{x}}$ of $\mathcal{F}+\mathcal{R}$. Moreover, $(\mathcal{F}(\mathbf{x}_k)+\mathcal{R}(\mathbf{x}_k))_{k\in\mathbb{N}}$ is a non increasing sequence converging to $\mathcal{F}(\hat{\mathbf{x}})+\mathcal{R}(\hat{\mathbf{x}})$.*

It is guaranteed under the following assumptions [31]

Assumption A.1. 1) *Let $k\in\mathbb{N}$ and let $j_k\in\{1,...,J\}$. The quadratic function defined as:*

For every $\mathbf{x}^{(j_k)}\in\mathbb{R}^{j_k}$,

$$Q_{j_k}(\mathbf{x}^{(j_k)}|\mathbf{x}_k)=\mathcal{F}(\mathbf{x}_k)+\left\langle \mathbf{x}^{(j_k)}-\mathbf{x}_k^{(j_k)},\nabla_{j_k}\mathcal{F}(\mathbf{x}_k)\right\rangle+\frac{1}{2}\|\mathbf{x}^{(j_k)}-\mathbf{x}_k^{(j_k)}\|^2_{P_{j_k}(\mathbf{x}_k)} \tag{23}$$

is a majorant function of the restriction of $\mathcal{F}$ to its j_k-th block on $\mathsf{dom}R_{j_k}$, i.e., for every $\mathbf{x}^{(j_k)}\in\mathsf{dom}R_{j_k}$,

$$\mathcal{F}(\mathbf{x}_k^{(1)},\ldots,\mathbf{x}_k^{(j_k-1)},\mathbf{x}_k^{(j_k)},\mathbf{x}_k^{(j_k+1)},\ldots,\mathbf{x}_k^{(J)})\leq Q_{j_k}(\mathbf{x}^{(j_k)}|\mathbf{x}_k) \tag{24}$$

Moreover, the eigenvalues of $P_{j_k}(\mathbf{x}_k)$ are lower and upper bounded by positive values.

2) *Blocks $(j_k)_{k\in\mathbb{N}}$ are updated according to an essentially cyclic rule, i.e., there exists $K\geq J$ such that, for every $k\in\mathbb{N}$, $\{1,\ldots,J\}\subset\{j_k,\ldots,j_k+K-1\}$.*

3) *One of the following statements holds:*

   a) *There exists $(\tilde{\gamma}_1,\tilde{\gamma}_2)\in]0,+\infty[^2$ such that, for every $k\in\mathbb{N},\tilde{\gamma}_1\leq\gamma_k\leq 1-\tilde{\gamma}_2$.*

   b) *For every $j\in\{1,\ldots,J\}$, R_j is a convex function and there exists $(\tilde{\gamma}_1,\tilde{\gamma}_2)\in]0,+\infty[^2$ such that, for every $k\in\mathbb{N},\tilde{\gamma}_1\leq\gamma_k\leq 2-\tilde{\gamma}_2$.*

REFERENCES

[1] J. R. Lacowicz. *Principle of fluorescence spectroscopy*. Kluwer Academic Plenum Publishers, 1999.

[2] P. Carroll and J. J. Chang. Analysis of individual differences in multi-dimensional scaling via n-way generalization of eckart-young decomposition. *Psychometrika*, 35:283–319, 1970.

[3] R. A. Harshman. Foundation of the Parafac procedure: models and conditions for an explanatory multimodal factor analysis. *UCLA Working papers in phonetics*, 16:1–84, 1970.

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

[5] R. Bro. *Multi-way analysis in the food industry: models, algorithms and applications*. PhD thesis, University of Amsterdam, The Netherlands, 1998.

[6] X. Meng, A. J. Morris, and E. B. Martin. On-line monitoring of batch processes using a parafac representation. *Journal of Chemometrics*, 17(1):65–81, 2003.

[7] C. A. Stedmon, S. Markager, and R. Bro. Tracing dissolved organic matter in aquatic environments using a new approach to fluorescence spectroscopy. *Marine Chemistry*, 82:239–254, 2003.

[8] C. A. Stedmon and S. Markager. Behaviour of the optical properties of coloured dissolved organic matter under conservative mixing. *Estuarine, Coastal and Shelf Science*, 57(5-6):973–979, 2003.

[9]Note that the Kurdyka-Łojasiewicz inequality holds for a wide class of functions [56].

[9] C. A. Stedmon and S. Markager. Resolving the variability in dissolved organic matter fluorescence in a temperate estuary and its catchment using parafac analysis. *Limnology and Oceanography*, 50(2):686–697, 2005.

[10] R. Bro and H. A. L. Kiers. A new efficient method for determining the number of components in parafac models. *J. Chemom.*, 17(5):274–286, May 2003.

[11] J. Da Costa, M. Haardt, and F. Roemer. Robust methods based on the hosvd for estimating the model order in parafac models. In *2008 5th IEEE Sensor Array and Multichannel Signal Processing Workshop*, pages 510–514, July 2008.

[12] L. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, pages 279–311, 1966.

[13] J. B. Kruskal. Three way arrays: Rank and uniqueness of trilinear decompositions with apllication to arithmetic complexity and statistics. *Linear Algebra Appl.*, 18:95–138, 1977.

[14] J. B. Kruskal. Rank, decomposition and uniqueness for 3-way and n-way arrays. *Multiway Data Analysis*, pages 7–18, April 1989.

[15] N. Sidiropoulos and R. Bro. On the uniqueness of multilinear decomposition of n-way arrays. *Journal of chemometrics*, 14:229–239, 2000.

[16] P. Paatero. Construction and analysis of degenerate Parafac models. *J. Chemometrics*, 14(3):285–299, 2000.

[17] N. Vervliet and L. De Lathauwer. A randomized block sampling approach to canonical polyadic decomposition of large-scale tensors. *IEEE Journal on Selected Topics in Signal Processing*, 10(2):284–295, 2016.

[18] N. Sidiropoulos, E. E. Papalexakis, and C. Faloutsos. Parallel randomly compressed cubes. *IEEE Signal Process. Mag. Special Issue on Big Data*, 31(5):57–70, 2014.

[19] P. G. Coble. Characterization of marine and terrestrial dom in seawater using excitation-emission matrix spectroscopy. *Marine Chemistry*, 52:325–346, 1996.

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

[21] J.-P. Royer, N. Thirion-Moreau, P. Comon, R. Redon, and S. Mounier. A regularized nonnegative canonical polyadic decomposition algorithm with preprocessing for 3D fluorescence spectroscopy. *Journal of Chemometrics*, 29:253–265, March 2015.

[22] A. Karami, M. Yazdi, and A. Asli. Noise reduction of hyperspectral images using kernel non-negative tucker decomposition. *Selected Topics in Signal Processing*, pages 487–493, 2011.

[23] 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.

[24] H. Wang and N. Ahuja. Compact representation of multidimensional data using tensor rank-one decomposition. In *Int'l Conf. Pattern Recognition (ICPR'04)*, volume 1, pages 44–47, 2004.

[25] H. Wang and N. Ahuja. A tensor approximation approach to dimensionality reduction. *Int'l Journal Computer Vision*, 73(3):217–229, 2008.

[26] I. Kopriva and A. Cichocki. Blind decomposition of low-dimensional multi-spectral image by sparse component analysis. *Journal of Chemometrics*, 23:590–597, 2009.

[27] I. Kopriva and A. Cichocki. Blind multispectral image decomposition by 3d nonnegative tensor factorization. *Optics Letters*, 34(14):2010–2012, 2009.

[28] A. H. Andersen and W. S. Rayens. Structure-seeking multilinear methods for the analysis of fmri data. *NeuroImage*, 22(2):728–739, 2004.

[29] A. S. Field and D. Graupe. Topographic component (parallel factor) analysis of multichannel evoked potentials: practical issues in trilinear spatiotemporal decomposition. *Brain topography*, 3(4):407–423, 1991.

[30] A. Cichocki, R. Zdunek, A. H. Phan, and S. I. Amari. *Non negative matrix and tensor factorizations: Application to exploratory multi-way data analysis and blind separation*. Wiley, 2009.

[31] E. Chouzenoux, J.-C. Pesquet, and A. Repetti. A block coordinate variable metric forward-backward algorithm. *J. Global Optim.*, pages 1–29, 2016.

[32] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.

[33] A. Repetti, E. Chouzenoux, and J.-C. Pesquet. A preconditioned forward-backward approach with application to large-scale nonconvex spectral unmixing problems. In *39th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)*, Florence, Italie, May, 4-9 2014.

[34] Y. Xu and W. Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM J. Imaging Sci.*, 6(3):1758–1789, 2013.

[35] Y. Zhang, G. Zhou, A. Cichocki, and X. Wang. Fast nonnegative tensor factorization based on accelerated proximal gradient and low-rank approxiamation. *Neurocomputing*, 198:148–154, Jul. 2016.

[36] D. R. Hunter and K. Lange. A tutorial on MM algorithms. *Amer. Statist*, pages 30–37, 2004.

[37] C. Chaux, P. L. Combettes, J.-C. Pesquet, and V. R. Wajs. A variational formulation for frame based inverse problems. *Inverse Probl.*, 23(4):1495–1518, Aug. 2007.

[38] A. Repetti, M. Q. Pham, L. Duval, E. Chouzenoux, and J.-C. Pesquet. Euclid in a taxicab: Sparse blind deconvolution with smoothed l1/l2 regularization. *IEEE Signal Processing Letters*, 22(5):539–543, May 2015.

[39] M. W. Mahoney. Randomized algorithm for matrices and data. *Foundations and trends in Machine Learning*, 3(2):123–224, November 2011.

[40] V. Cevher, S. Becker, and M. Schmidt. Convex optimization for big data: scalable, randomized and parallel algorithms for big data analytics. *IEEE Signal Processing Magazine*, 31(5):32–43, September 2014.

[41] E. Chouzenoux, J.-C. Pesquet, and A. Repetti. Variable metric forward-backward algorithm for minimizing the sum of a differentiable function and a convex function. *Journal of Optimization Theory and Applications*, 162(1):107–132, Jul. 2014.

[42] J.-J. Moreau. Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. France*, 93:273–299, 1965.

[43] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4):1168–1200, Nov. 2005.

[44] P. L. Combettes and J.-C. Pesquet. A proximal decomposition method for solving convex variational inverse problems. *Inverse Probl.*, 24(6):27, Dec. 2008.

[45] J.-B. Hiriart-Urruty and C. Lemarchal. *Convex Analysis and Minimization Algorithms*. Springer-Verlag, New York, 1993.

[46] C. G. Khatri and C. R. Rao. Solutions to some functional equations and their applications to characterization of probability distributions. *Sankhya: Indian J. Statistics*, Series A, 30:167–180, 1968.

[47] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556–562. MIT Press, 2001.

[48] P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, editors, *Fixed-point algorithms for inverse problems in science and engineering*, pages 185–212. Springer Verlag, 2010.

[49] X. T. Vu, C. Chaux, S. Maire, and N. Thirion-Moreau. Study of different strategies for the canonical polyadic decomposition of nonnegative third order tensors with application to the separation of spectra in 3D fluorescence spectroscopy. In *IEEE Machine Learning for Signal Processing (MLSP'14)*, Reims, France, Sept. 2014.

[50] R. Bro. Parafac: tutorial and applications. *Chemom. Intell. Lab. Syst.*, 38(2):149–171, October 1997.

[51] A. H. Phan, P. Tichavskỳ, and A. Cichocki. Fast alternating ls algorithms for high order candecomp/parafac tensor factorizations. *IEEE Trans. on Sig. Proc.*, 61(19):4834–4846, June 2013.

[52] G. H. Paul. *Introduction to Mathematical Statistics*. Wiley, 5th edition, 1984.

[53] E. M. Carstea, S. Mounier, R. Redon, X. Luciani, C. Gadio, and A. Baker. On-line parafac analysis of fluorescence spectra. In *International Workshop on Organic Matter Spectroscopy (WOMS'13)*, Toulon, France, July 2013.

[54] R. G. Zepp, W. M. Sheldon, and M. A. Moran. Dissolved organic fluorophores in southeastern us coastal waters: correction method for eliminating rayleigh and raman scattering peaks in excitation-emission matrices. *Marine Chemistry*, 89(1-4):15–36, 2004.

[55] Z. Fan, O. Schroder, and J. Krahl. Analysis of diesel fuels/biodiesel blends and identification of biodiesel using time-resolved laser-induced fluorescence spectroscopy (trlfs). *Appl Agric Forestry Res*, 65:1–14, 2015.

[56] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming*, 116(1):5–16, 2009.

[57] S. Łojasiewicz. Une propriété topologique des sous-ensembles analytiques réels. *Editions du Centre National de le Recherche Scientifique*, pages 87–89, 1963.

[58] K. Kurdyka and A. Parusinski. w_f-stratification of subanalytic functions and the Łojasiewicz inequality. *Comptes rendus de l'acadmie des Sciences*, 318(2):129–133, 1994.

LIST OF FIGURES

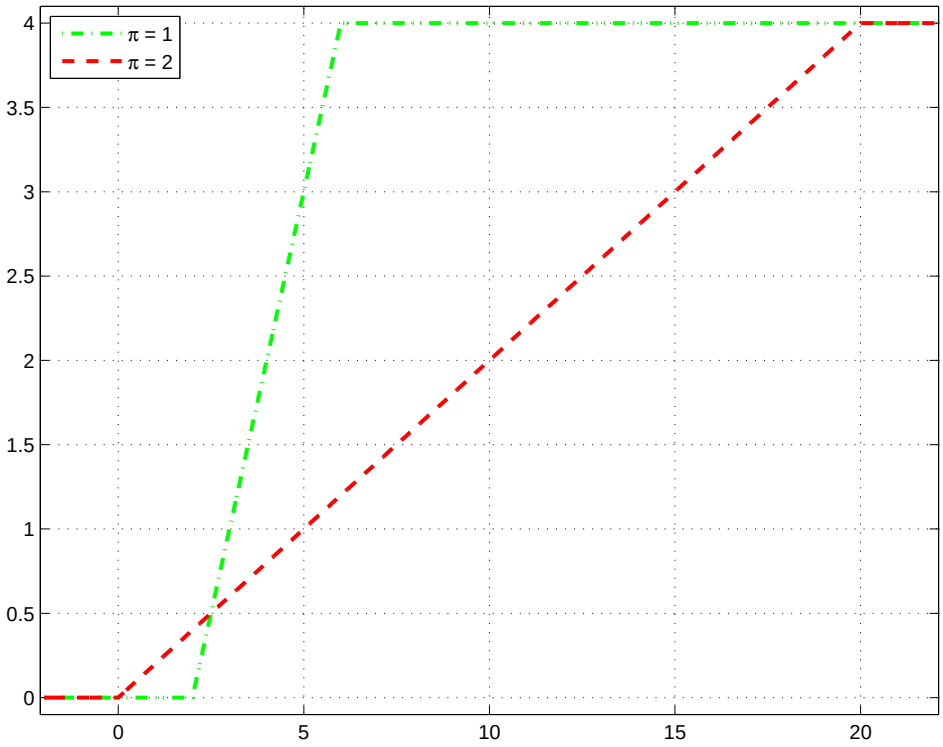

Fig. 1. Plots of $\mathrm{prox}_{\rho_n}(\upsilon)$ in function of $\upsilon \in [-2, 22]$ when $\alpha^{(n)} = 2$ and $[\eta_{\min}^{(n)}, \eta_{\max}^{(n)}] = [0, 4]$.

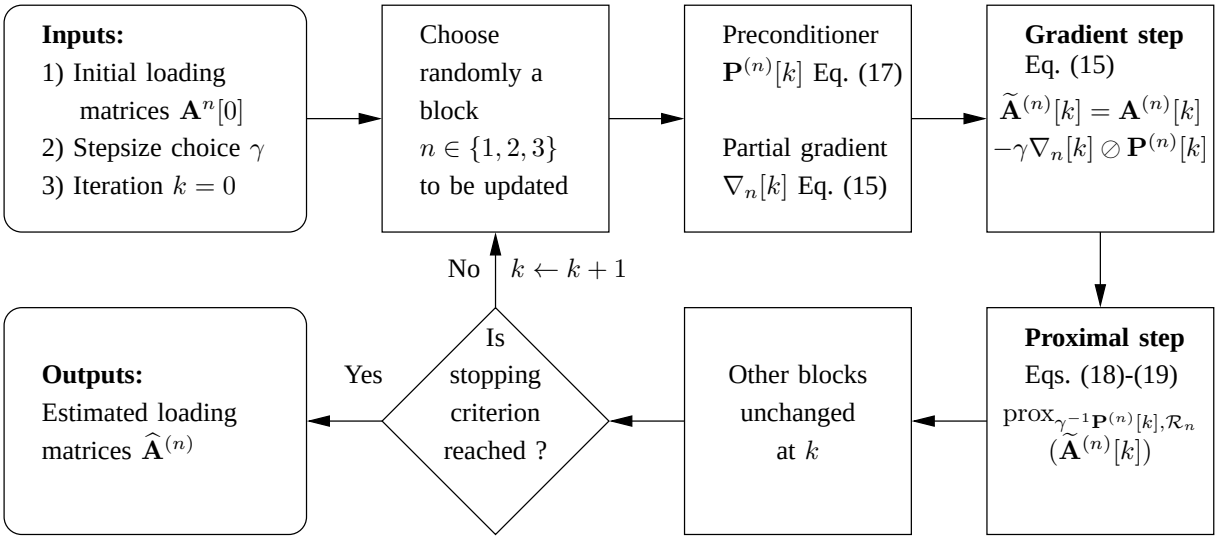

Fig. 2. BC-VMFB algorithm to minimize (11).

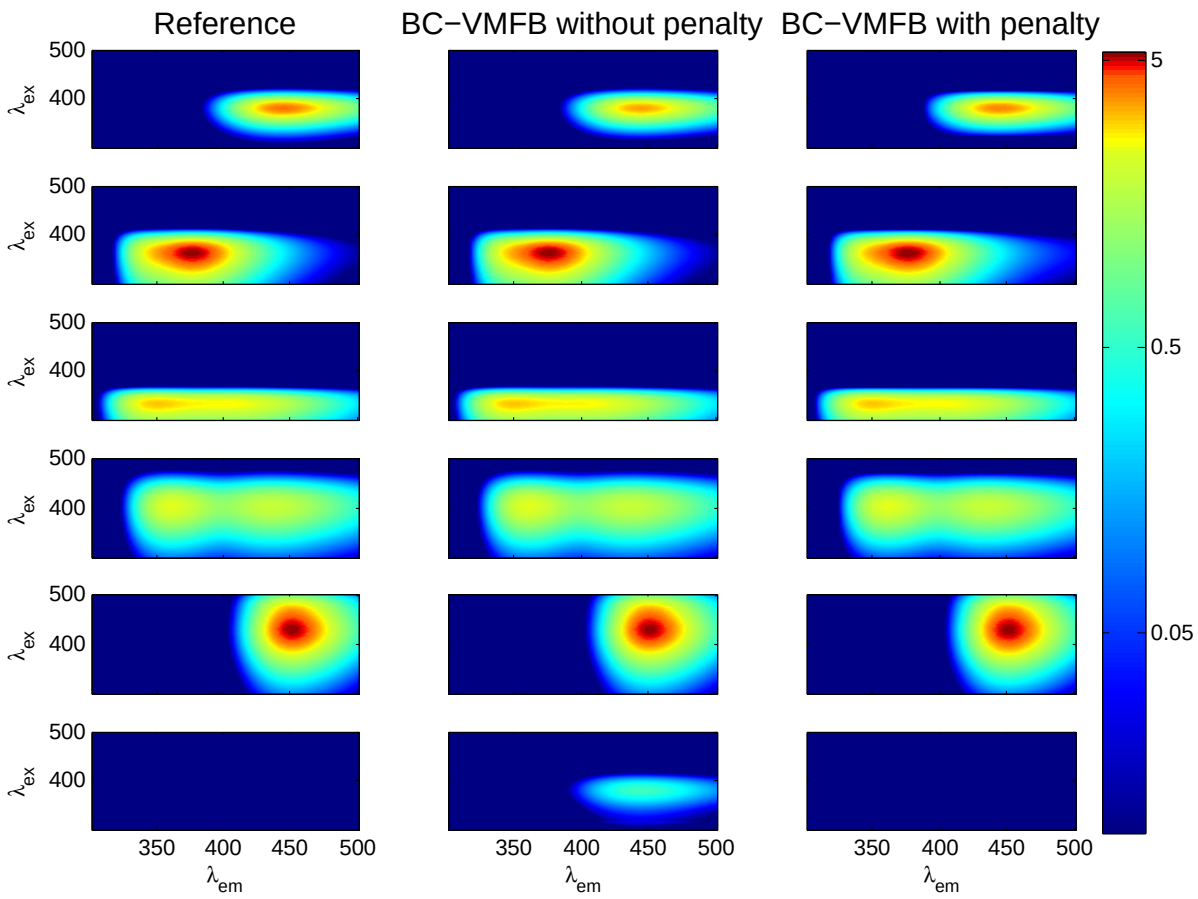

Fig. 3. The FEEM of reference (left column) and the FEEM reconstructed by the BC-VMFB algorithm in two cases: without regularization (middle column) and with regularization $\alpha = 0.05$ (right column) in the case of noiseless data

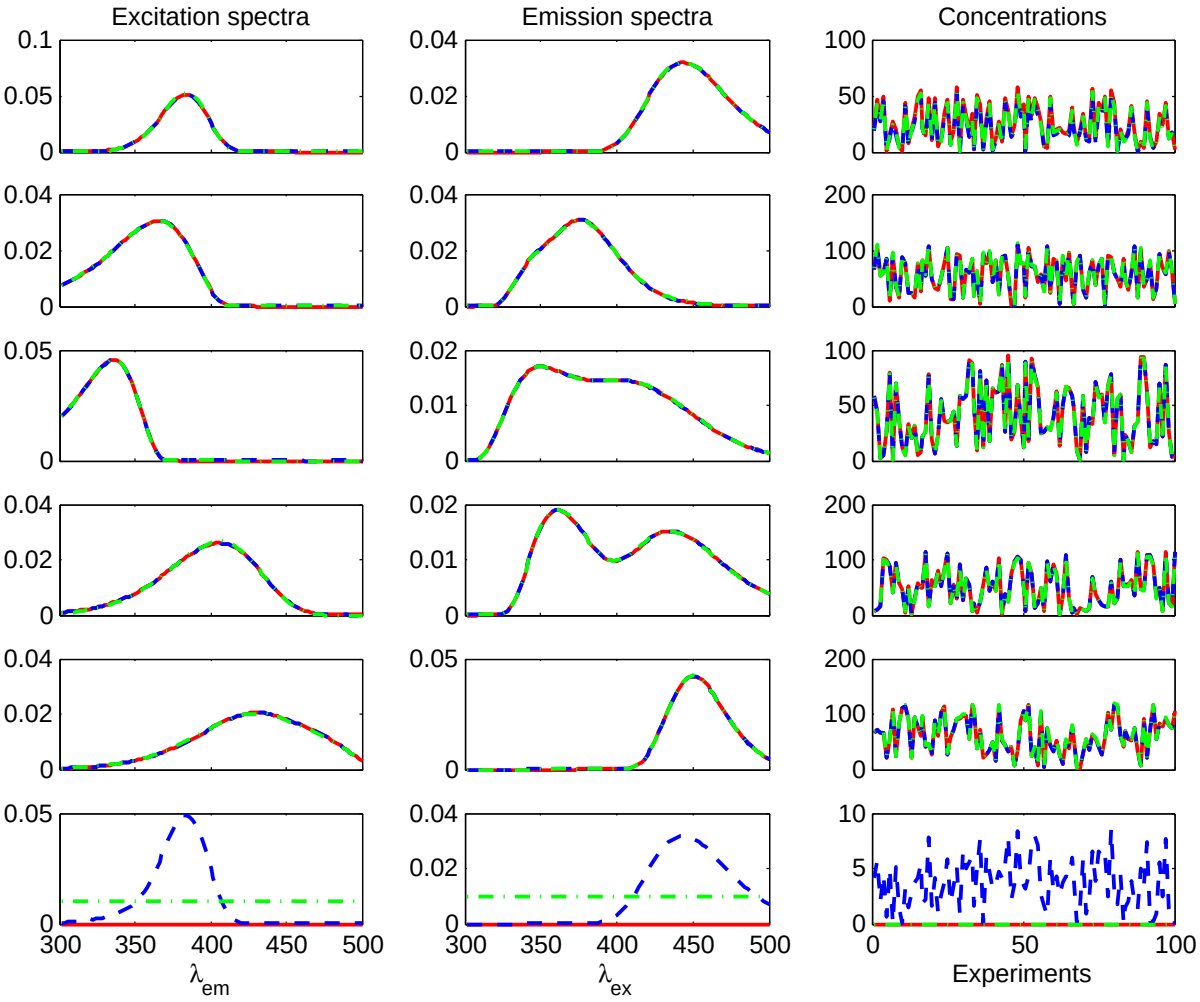

Fig. 4. The $\widehat{R} = 6$ emission spectra, excitation spectra and concentrations estimated with the BC-VMFB algorithm in two cases: without regularization and with regularization $\alpha = 0.05$ in the case of noiseless data. In solid red line: the reference spectra, in dashed blue line: BC-VMFB without penalty and in dash-dot green line: BC-VMFB with penalty.

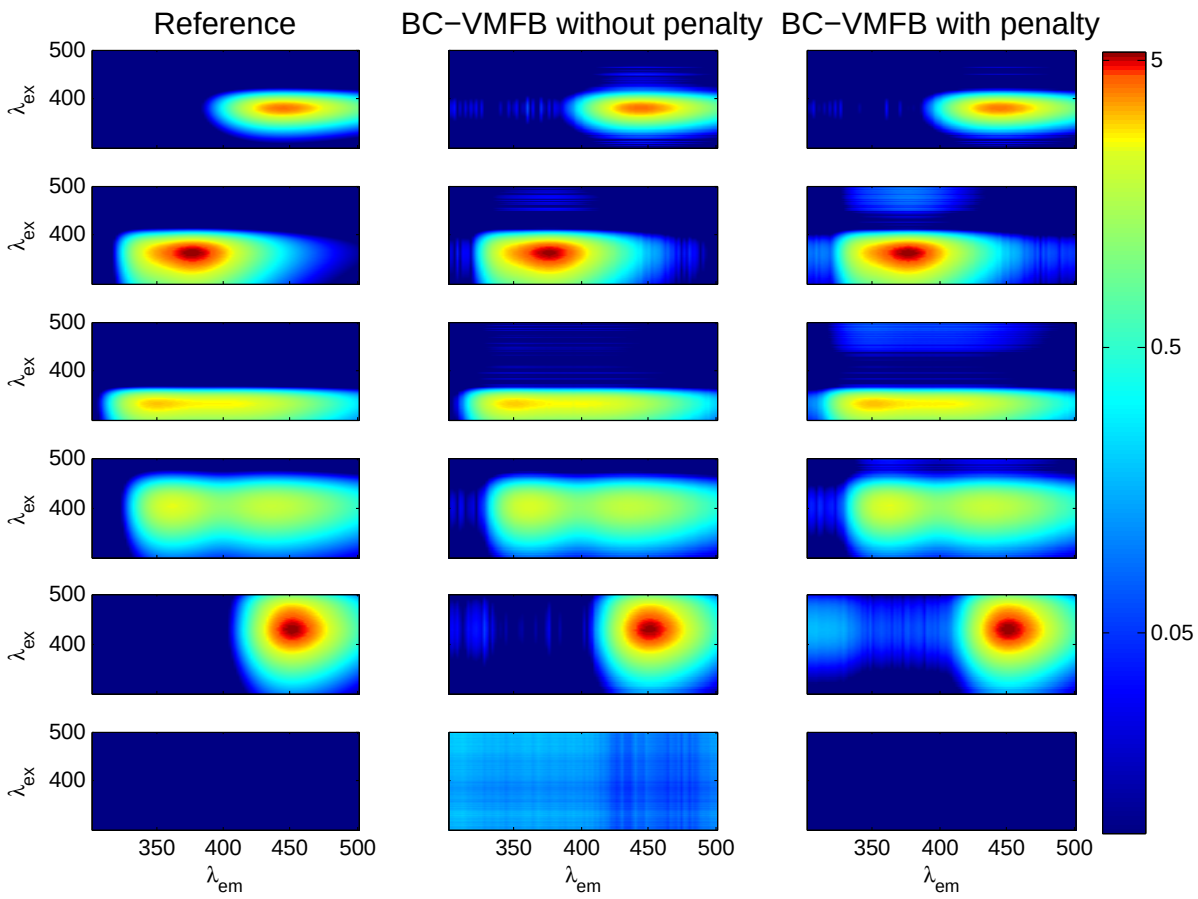

Fig. 5. The FEEM of reference (left column) and the estimated FEEM using the BC-VMFB algorithm, in two cases: without regularization (middle column) and with regularization $\alpha = 0.05$ (right column) in the noisy case (for a $\mathsf{SNR} = 17.6$ dB)

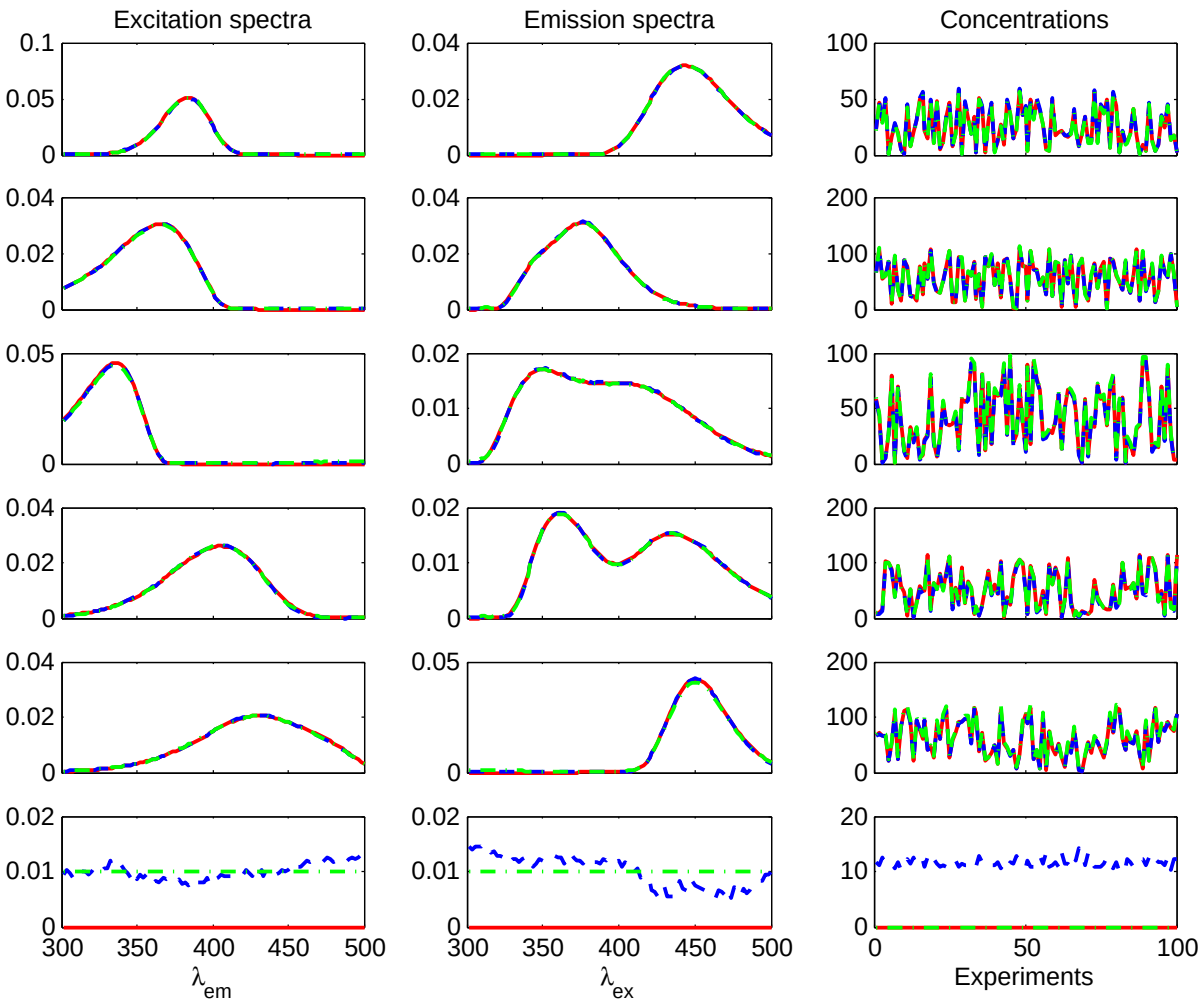

Fig. 6. The $\widehat{R} = 6$ estimated spectra by BC-VMFB in two cases: without regularization and with regularization $\alpha = 0.05$ in the noisy case (for a $\mathsf{SNR} = 17.6$ dB). In solid red line: the reference spectra, in dashed blue line: the BC-VMFB without penalty and in dash-dot green line: BC-VMFB with penalty.

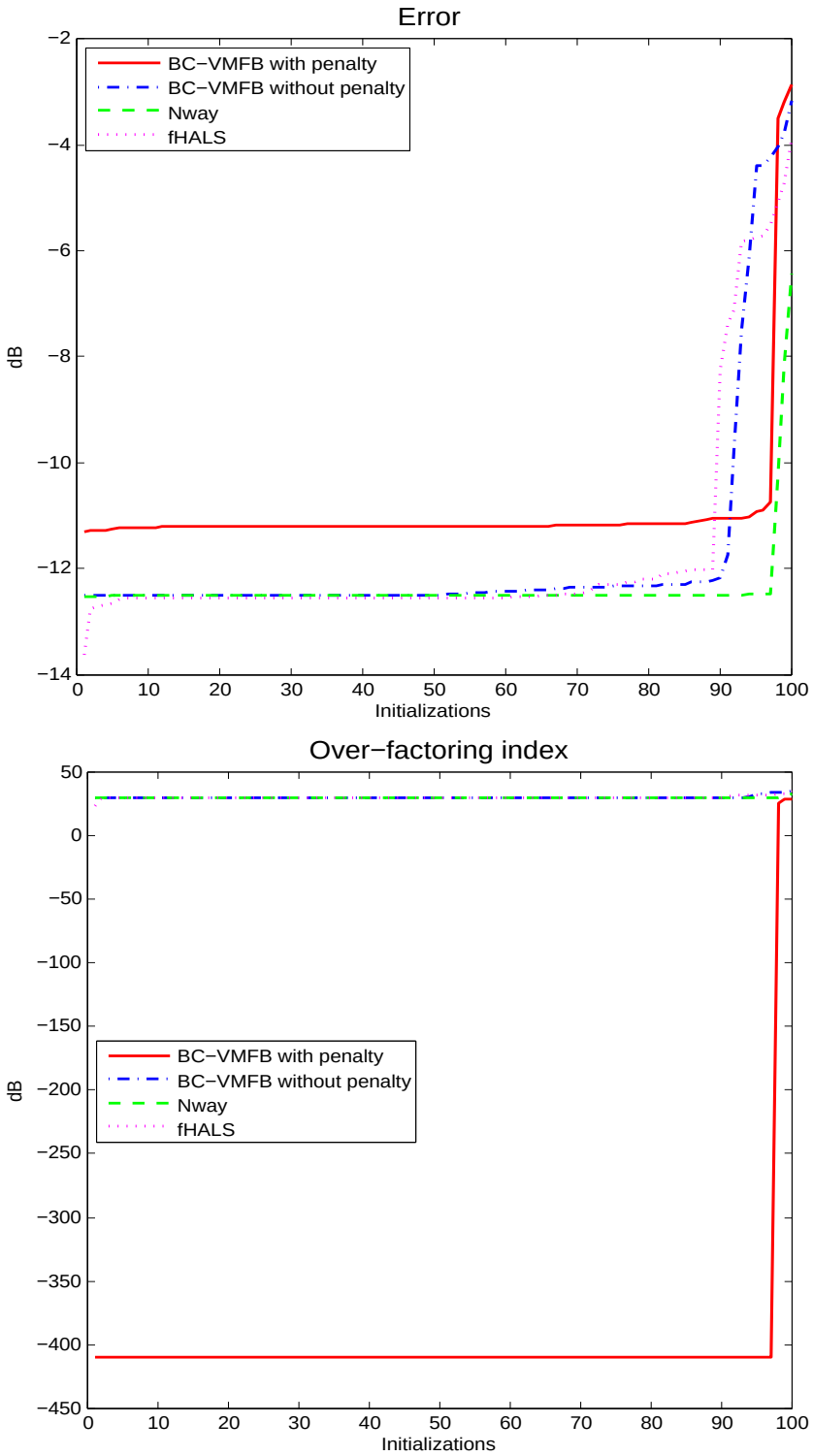

Fig. 7. Performance versus the different initializations sorted in ascending order in the noisy, overestimated case: error index $\mathbf{E}_1$ (top), overfactoring error index $\mathbf{E}_2$ (bottom).

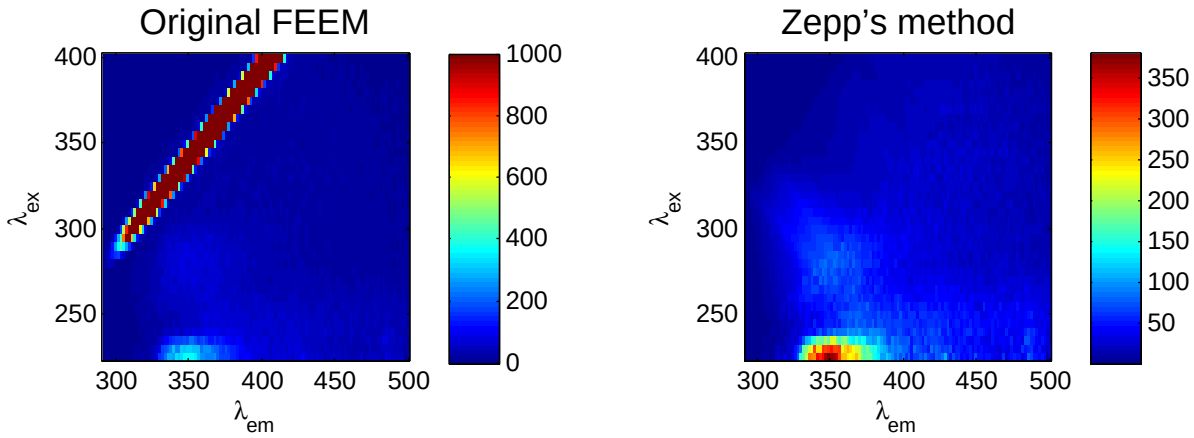

Fig. 8. One of the 2594 original FEEM acquired during the monitoring campaign and the same FEEM after pre-processing thanks to the Zepp's method

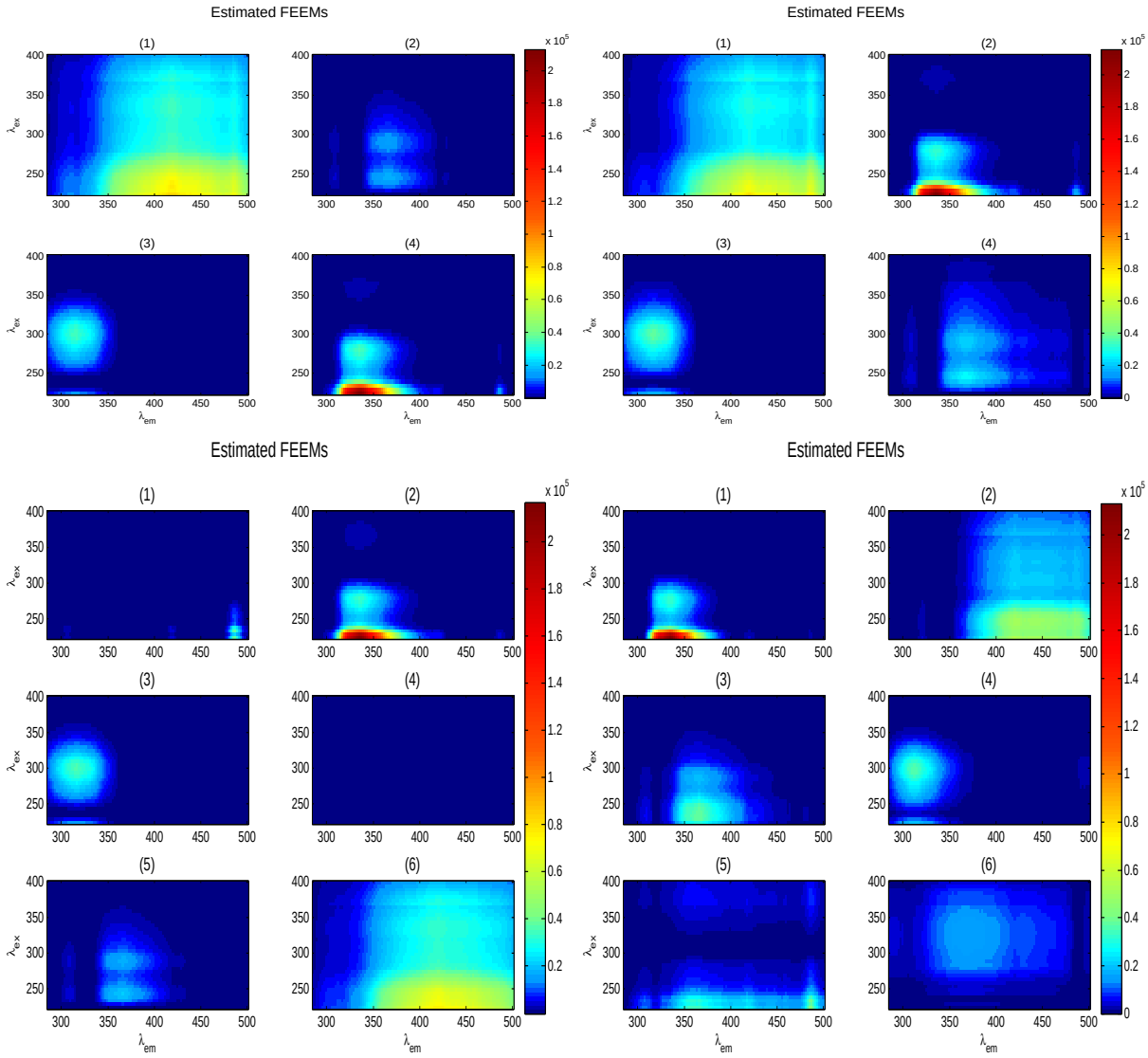

Fig. 9. Estimated FEEM using the penalized BC-VMFB algorithm (left), and Bro's N-way algorithm (right). Case $\widehat{R} = 4$ (top), case $\widehat{R} = 6$ (below)

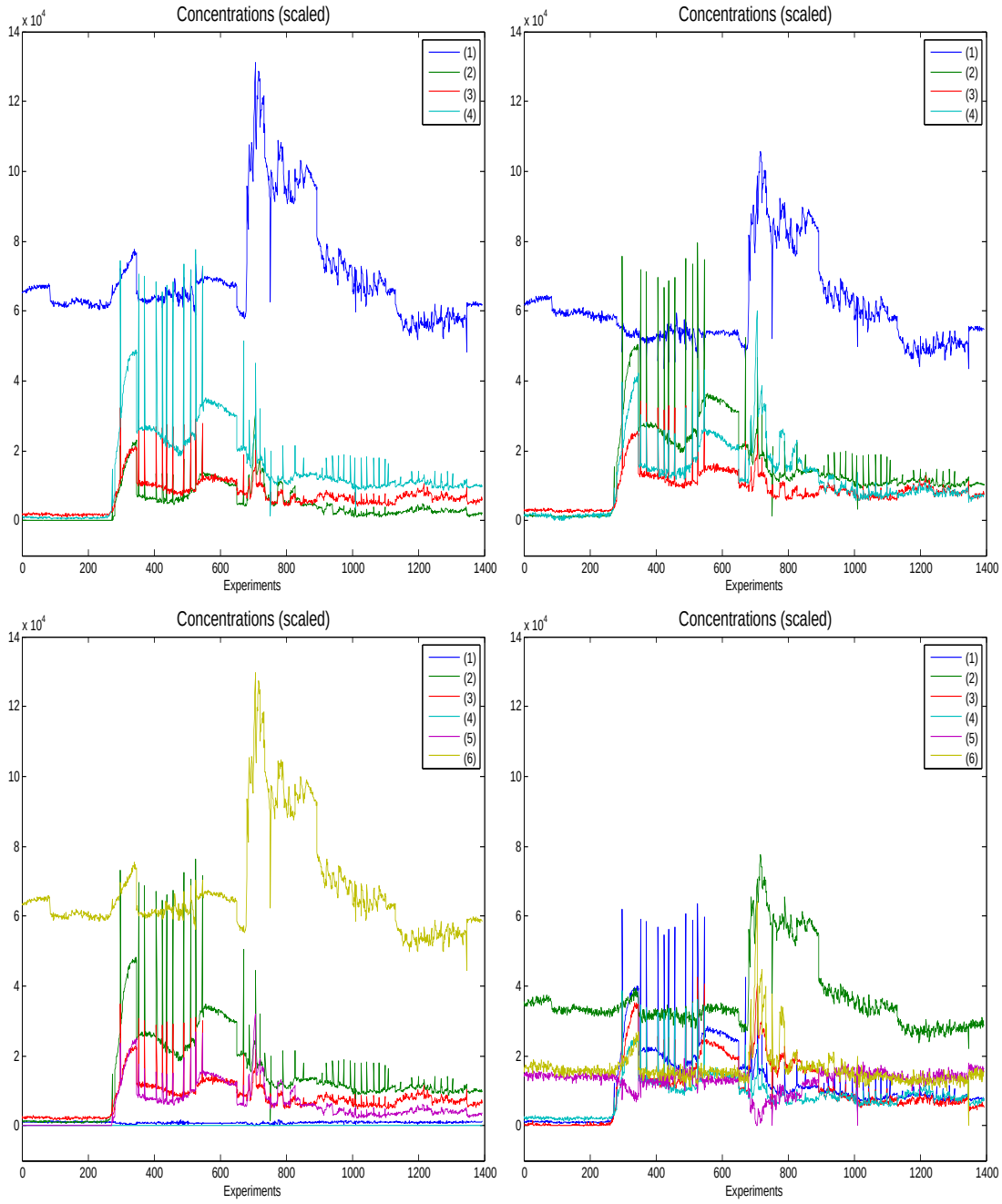

Fig. 10. Estimated concentrations, using the penalized BC-VMFB algorithm (left), using Bro's N-way algorithm (right). Case $\widehat{R} = 4$ (top), case $\widehat{R} = 6$ (bottom)

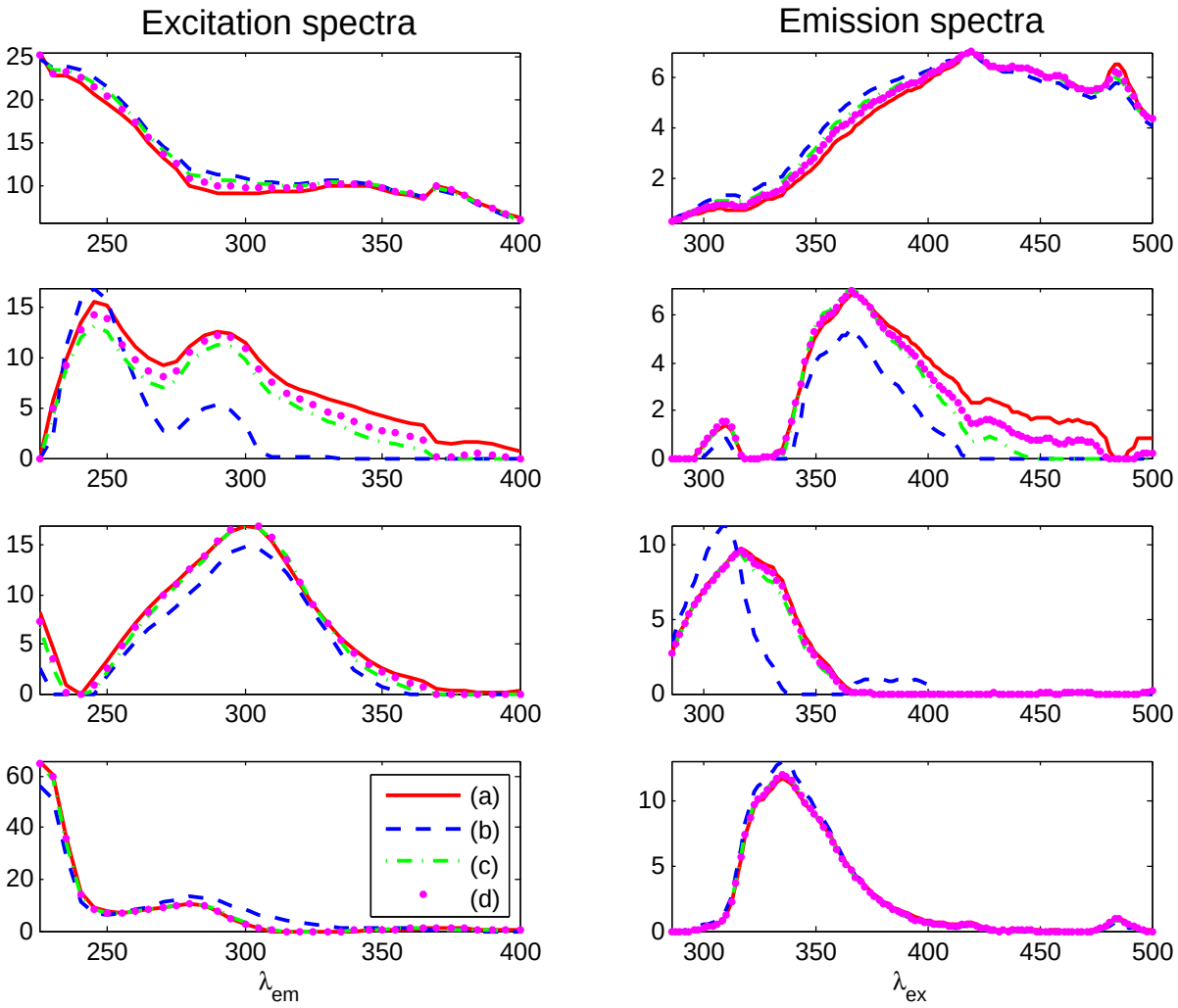

Fig. 11. Estimated emission and excitation spectra using Bro's N-way with nonnegativity constraints (a) and BC-VMFB with different regularization parameters (b, c, d), case $\widehat{R} = 4$

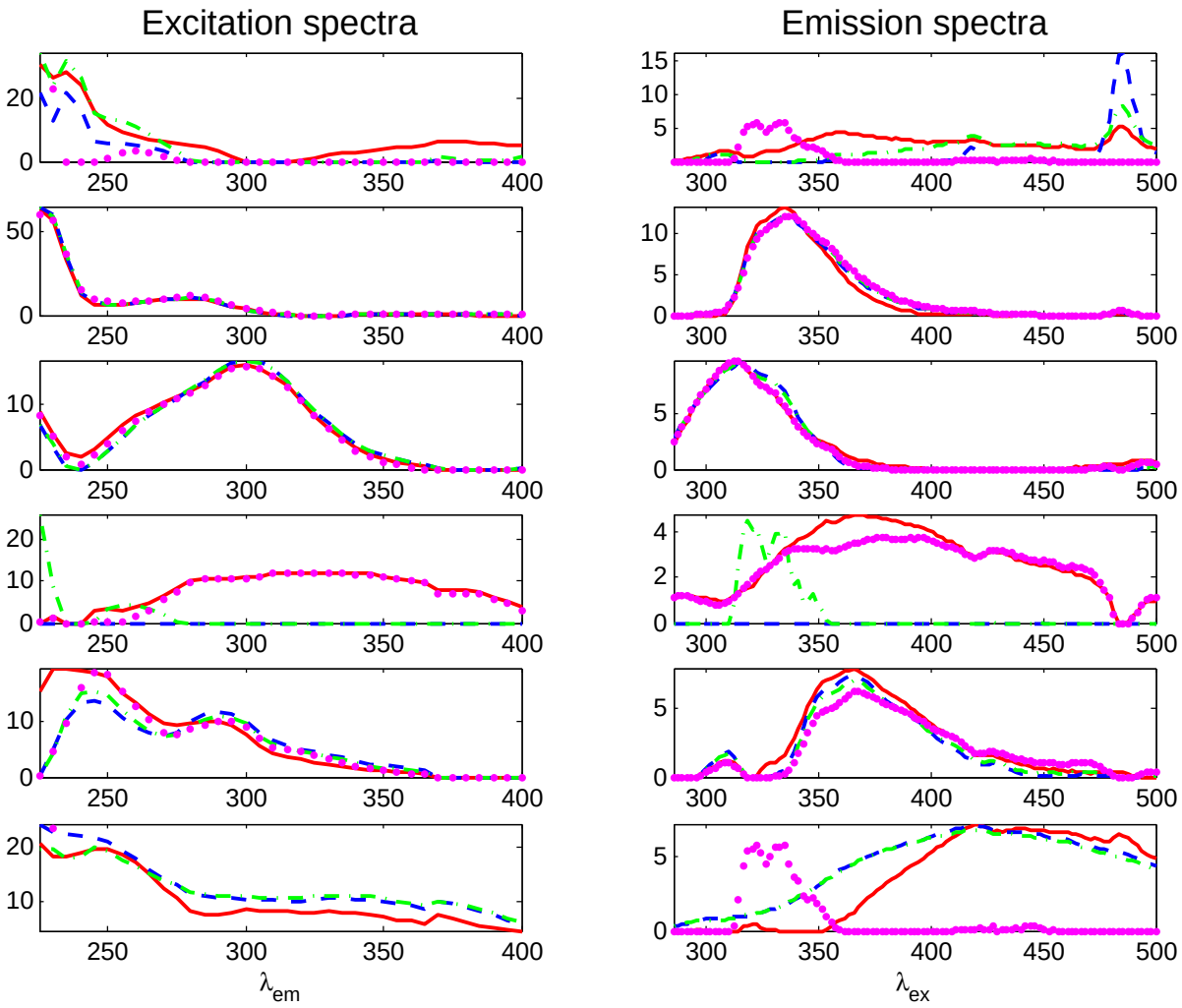

Fig. 12. Estimated emission and excitation spectra using Bro's N-way with nonnegativity constraints ((a) solid red line) and BC-VMFB with different regularization parameters ((b) dashed blue line, (c) dash-dot green line, (d) dotted magenta line), case $\widehat{R} = 6$

LIST OF TABLES

	Elapsed time (s)	BC-VMFB without penalty	BC-VMFB with penalty	N-way	fast HALS
Noisy case	For 50 iterations	0.2	0.2	11	0.5
	To reach stopping conditions (actual number of iterations) (SNR,E_1, E_2) dB	102 (48500) (31.3, -12.5, 30.6)	75 (36500) (32.7, -11.2, -409)	8 (43) (31.3, -12.5, 30.6)	8 (1856) (31.3, -12.5, 30.6)
Noiseless case	To reach stopping conditions (actual number of iterations) (RRE,E_1, E_2) dB	202 (100000) (-75.1,-12.4,25.6)	74 (36500) (-44.7, -15, -409)	80 (838) (-127.9,-8.7, 31.7)	3.7 (308) (-63.9, -6.1, 31.7)

TABLE I

COMPUTATION TIME COMPARISON OF BC-VMFB IN TWO CASES: WITH OR WITHOUT PENALTY, WITH N-WAY AND FAST HALS USING THE SAME INITIAL VALUE IN NOISY DATA AND IN NOISELESS DATA.